Manuel de l'Utilisateur calibre

Version 7.9.0

Kovid Goyal

avril 19, 2024

Table des matières

1 L'Interface Graphique Utilisateur 3

2 Ajouter votre site web favori d'actualités 33

3 La visionneuse de livre numérique 53

4 Conversion de livre numérique 61

5 Éditer des livres numériques 81

6 Le Serveur de contenu calibre 117

7 Comparer des livres numériques 125

8 Éditer les métadonnées d'un livre numérique 129

9 Questions Fréquemment Posées 133

10 Tutoriels 161

11 Le système d'URL calibre :// 263

12 Personnaliser calibre 267

13 Interface en ligne de commande 315

14 Mise en place d'un environnement de développement calibre 357

15 Gestion des Droits Numériques (GDN - DRM) 389

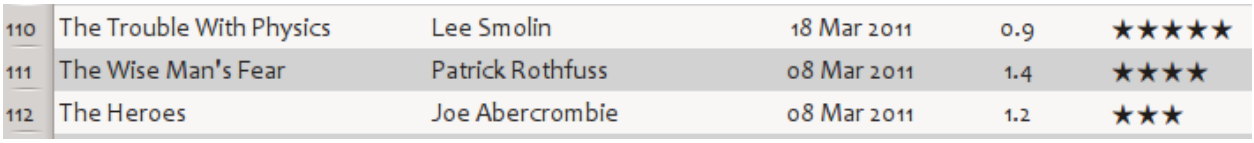
16 Glossaire 393

Index des modules Python 395

Index 397

calibre est un gestionnaire de bibliothèque de livres numériques. Il peut visualiser, convertir et cataloguer des livres numériques dans la plupart des principaux formats. Il peut aussi communiquer avec beaucoup de périphériques de lecture de livre numérique. Il peut aller sur Internet et chercher des métadonnées pour vos livres. Il peut télécharger des journaux et les convertir en livres numériques pour une lecture confortable. Il est multi-plateforme, fonctionnant sur Linux, Windows et Mac macOS .

Vous venez de commencer avec calibre. Qu'allez-vous faire maintenant ? Avant que calibre puisse faire quoique ce soit avec vos livres numériques, il faut d'abord qu'il les connaisse. Glissez et déposez quelques fichiers de livre numérique dans calibre, ou cliquez sur le bouton « Ajouter des livres » et sélectionnez les livres numériques avec lesquels vous voulez travailler. Une fois que vous aurez ajouté ces livres, ils apparaîtront dans l'écran principal ressemblant à quelque chose comme ceci :



110	The Trouble With Physics	Lee Smolin	18 Mar 2011	0.9	★★★★★
111	The Wise Man's Fear	Patrick Rothfuss	08 Mar 2011	1.4	★★★★
112	The Heroes	Joe Abercrombie	08 Mar 2011	1.2	★★★

Une fois que vous aurez admiré avec satisfaction la liste des livres que vous venez d'ajouter, vous aurez probablement envie de les lire. Pour ce faire, vous devrez convertir le livre dans un format que votre liseuse comprenne. Lors de la première ouverture de calibre, *l'Assistant de bienvenue* a démarré et a paramétré calibre pour votre périphérique de lecture. La conversion est un jeu d'enfant. Il suffit de sélectionner le livre que vous voulez convertir, puis de cliquer sur le bouton « Convertir des livres ». Ignorer toutes les options pour l'instant et cliquez sur « OK ». La petite icône dans le coin en bas à droite va commencer à tourner. Une fois que c'est fini, votre livre converti est prêt. Cliquez sur le bouton « Visualiser » pour lire le livre.

Si vous voulez lire le livre sur votre liseuse, connecter la liseuse à l'ordinateur, attendez que calibre la détecte (10-20 secondes), puis cliquez sur le bouton « Envoyer au périphérique ». Une fois que l'icône s'arrête de tourner à nouveau, déconnectez votre liseuse et lisez tout de suite ! Si vous n'aviez pas converti le livre dans l'étape précédente, calibre l'aura automatiquement fait pour que le livre soit dans un format que votre périphérique de lecture peut comprendre.

Pour vous lancer dans un usage plus avancé, vous devriez lire *L'Interface Graphique Utilisateur* (page 3). Pour encore plus de puissance et de polyvalence, apprenez *Interface en ligne de commande* (page 315). Vous trouverez la liste des *Questions Fréquemment Posées* (page 133) bien utile elle aussi.

Si vous avez d'autres questions ou si vous voulez discuter de calibre avec d'autres utilisateurs, demander de l'aide pour des choses spécifiques, il existe des forums et d'autres ressources d'aide disponibles[1].

Sections

[1] https://calibre-ebook.com/fr/help

CHAPITRE 1

L'Interface Graphique Utilisateur

L'Interface Graphique Utilisateur *(GUI)* permet d'accéder à toute la gestion de la bibliothèque et aux caractéristiques de conversion des formats de livre numérique. La méthode de travail basique pour l'utilisation de calibre est d'abord d'ajouter des livres à la bibliothèque depuis votre disque dur. calibre essaiera de lire les métadonnées des livres et les ajoutera à sa base de donnée interne. Une fois qu'ils sont dans la base de données, vous pouvez effectuer différentes *Actions* (page 4) sur ceux-ci incluant la conversion d'un format à un autre, transférer vers le périphérique de lecture, visualiser sur votre ordinateur, et éditer les métadonnées. Cette dernière inclut la modification de la couverture, de la description et les étiquettes parmi d'autres détails. Notez que calibre crée des copies des fichiers que vous lui ajoutez. Vos fichiers originaux sont laissés intacts.

L'interface est divisée en différentes sections :

- *Actions* (page 4)
- *Préférences* (page 11)
- *Catalogues* (page 11)
- *Rechercher et trier* (page 12)
- *L'interface de recherche* (page 13)
- *Recherches sauvegardées* (page 19)
- *Recherche dans le texte intégral de tous les livres* (page 19)
- *Bibliothèques virtuelles* (page 19)
- *Marquage temporaire des livres* (page 20)
- *Deviner les métadonnées à partir des noms de fichier* (page 20)
- *Détails du livre* (page 21)
- *Navigateur d'étiquettes* (page 23)
- *Grille de couvertures* (page 25)
- *Navigateur de couvertures* (page 26)

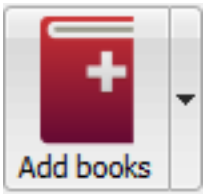
- *Ajouter des notes pour les auteurs, les séries, etc.* (page 27)
- *Affichage rapide* (page 28)
- *Tâches* (page 29)
- *Raccourcis clavier* (page 29)

1.1 Actions

La barre d'outils contient des raccourcis pratiques pour les actions couramment utilisées. Si vous faites un clic droit sur les icônes, vous pouvez effectuer des variations de l'action par défaut. Notez que la barre d'outils a une apparence légèrement différente selon que vous avez un lecteur de livre numérique connecté à votre ordinateur ou pas.

- *Ajouter des livres* (page 4)
- *Modifier les métadonnées* (page 5)
- *Convertir des livres* (page 6)
- *Visualiser* (page 6)
- *Envoyer vers le périphérique* (page 7)
- *Récupérer des informations* (page 7)
- *Bibliothèque* (page 8)
- *Périphérique connecté* (page 9)
- *Enregistrer sous…* (page 9)
- *Connecter/partager* (page 10)
- *Supprimer des livres* (page 10)

1.1.1 Ajouter des livres



L'action *Ajouter des livres* a sept variations accessibles en faisant un clic droit sur le bouton.

1. **Ajouter des livres à partir d'un dossier unique** : ouvre une boîte de dialogue de sélection de fichiers et vous permet de spécifier quels livres doivent être ajoutés à partir d'un dossier. Cette action est *sensible au contexte*, c'est à dire qu'elle dépend de quel *catalogue* (page 11) vous avez sélectionné. Si vous avez sélectionné la *Bibliothèque* , les livres seront ajoutés à la bibliothèque. Si vous avez sélectionné le périphérique de lecture de livre numérique, les livres seront téléchargés sur le périphérique, et ainsi de suite.

2. **Ajouter les livres à partir des dossiers et sous-dossiers** : Permet de choisir un dossier. Le dossier et tous ses sous-dossiers sont scannés récursivement, et les livres numériques trouvés sont ajoutés à la bibliothèque. Vous pouvez choisir de faire ajouter par calibre tous les fichiers présents dans un seul dossier à une seule fiche de livre ou à plusieurs fiches de livre. calibre suppose que chaque dossier contient un seul livre. Tous les fichiers de livres numériques d'un dossier sont supposés être le même livre dans différents formats. Cette action est l'inverse de l'action Enregistrer sous…`<save_to_disk_multiple>, c'est-à-dire que vous pouvez :guilabel :`Enregistrer sous…, supprimer les livres et les rajouter en mode livre unique par dossier, sans aucune information perdue à l'exception de la date (cela suppose que vous n'avez pas modifié les paramètres de l'action Enregistrer sous…).
3. **Ajouter de multiples livres à partir d'un dossier compressé (ZIP/RAR)** : vous permet d'ajouter de multiples livres numériques qui sont stockés à l'intérieur des fichiers ZIP ou RAR sélectionnés. C'est un raccourci pratique qui évite de devoir d'abord décompresser des archives et puis d'ajouter les livres par l'intermédiaire d'une des deux options ci-dessus.
4. **Ajouter un livre vide (une entrée de livre sans formats)** : vous permet de créer un fichier livre vide. Cela peut servir à remplir manuellement les informations sur un livre que vous n'avez pas encore dans votre collection.
5. **Ajouter à partir de l'ISBN** : Vous permet d'ajouter un ou plusieurs livres en entrant leurs numéros ISBN.
6. **Ajouter les fichiers aux enregistrements de livre sélectionnés** : Vous permet d'ajouter ou de mettre les fichiers à jour liés à un livre existant dans votre bibliothèque.
7. **Ajouter des fichiers de données aux enregistrements de livres sélectionnés** : Permet d'ajouter un nombre quelconque de fichiers supplémentaires qui seront stockés dans un sous-répertoire `data` dans le répertoire du livre. Voir *Ajouter des fichiers de données supplémentaires à un livre* (page 132) pour plus de détails.
8. **Ajouter un fichier vide aux fiches de livres sélectionnées** : Vous permet d'ajouter un fichier vide du format spécifié aux fiches de livres sélectionnées.

L'action *Ajouter des livres* peut lire les métadonnées à partir d'une grande variété de formats de livres numériques. En outre, calibre essaie de deviner les métadonnées du fichier. Voir la section *Deviner les métadonnées à partir des noms de fichier* (page 20) pour apprendre comment configurer cela.

Pour ajouter un format additionnel à un livre existant vous pouvez faire n'importe laquelle de ces trois actions :

1. Glisser et déposer le fichier sur le panneau des détails du livre sur le côté droit de la fenêtre principale.
2. Faire un clic droit sur le bouton *Ajouter des livres* et choisir *Ajouter des fichiers aux livres sélectionnés*.
3. Cliquer le bouton *Ajouter des livres* dans la partie supérieure droite de la boîte de dialogue *Modifier les métadonnées*, accédé par l'action *Modifier les métadonnées* (page 5).

1.1.2 Modifier les métadonnées

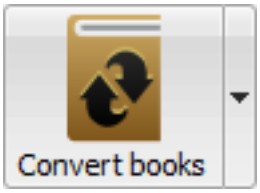
L'action *Modifier les métadonnées* a quatre variations accessibles en faisant un clic droit sur le bouton.

1. **Modifier les métadonnées individuellement** : permet de modifier les métadonnées des livres un par un, avec la possibilité de récupérer des métadonnées, y compris les couvertures, à partir de l'Internet. Il vous permet également d'ajouter ou de supprimer des formats de livres numériques particuliers pour ce livre.
2. **Modifier les métadonnées par lot** : Permet de modifier les champs de métadonnées communes à plusieurs livres, de manière simultanée. Cela fonctionne pour tous les livres que vous avez sélectionnés dans la *Vue bibliothèque* (page 12).

3. **Télécharger les métadonnées et les couvertures** : Télécharge les métadonnées et les couvertures (si disponible) pour les livres qui sont sélectionnés dans la liste de livres.
4. **Fusionner les enregistrements de livres** : Vous donne la possibilité de fusionner les métadonnées et les formats de deux ou plusieurs enregistrements de livres. Vous pouvez choisir de supprimer ou de conserver les enregistrements qui n'ont pas été cliqué en premier.
5. **Gérer les fichiers de données** : Gérer les fichiers de données supplémentaires associés aux livres sélectionnés.

Pour plus de détails, voir *Éditer les métadonnées d'un livre numérique* (page 129).

1.1.3 Convertir des livres



Les livres numériques peuvent être convertis à partir d'un certain nombre de formats dans le format que votre liseuse préfère. De nombreux livres numériques disponibles à l'achat seront protégés par la technologie *Digital Rights Management* (page 389) *(DRM)*. calibre ne sait pas convertir ces livres numériques. Il est facile de supprimer les DRM de nombreux formats, mais c'est illégal, vous devrez trouver vous-même des outils pour déverrouiller vos livres et utiliser ensuite calibre pour les convertir.

Pour la plupart des gens, la conversion devrait être l'affaire d'un simple clic. Si vous voulez en savoir plus sur le processus de conversion, voir *Conversion de livre numérique* (page 61).

L'action *Convertir des livres* a trois variations accessibles en faisant un clic droit sur le bouton.

1. **Convertir individuellement** : Vous permet de spécifier les options de conversion pour personnaliser la conversion de chaque livre numérique sélectionné.
2. **Convertir par lot** : Vous permet de spécifier des options qu'une seule fois pour convertir un certain nombre de livres numériques en même temps.
3. **Créer un catalogue des livres de votre bibliothèque calibre** : Vous permet de générer une liste complète des livres dans votre bibliothèque, en y précisant les métadonnées, et ce dans plusieurs formats tels que XML, CSV, BiBTeX, EPUB et MOBI. Le catalogue contiendra tous les livres se trouvant visibles à ce moment dans la vue bibliothèque. Cela vous permet d'utiliser les fonctions de recherche pour limiter les livres à cataloguer. En outre, si vous sélectionnez plusieurs livres avec la souris, seuls les livres sélectionnés seront compris dans le catalogue. Si vous générez le catalogue dans un format de livre numérique comme EPUB, MOBI ou AZW3, la prochaine fois que vous connecterez votre lecteur de livre numérique, le catalogue sera envoyé automatiquement au périphérique. Pour avoir plus d'informations sur comment fonctionne les catalogues lisez *Création de catalogues AZW3 • EPUB • MOBI* (page 255).

1.1.4 Visualiser

L'action *Visualiser* affiche le livre dans un programme de visualisation de livre numérique. calibre a une visionneuse intégrée qui fonctionne pour de nombreux formats de livre numérique. Pour d'autres formats, il utilise l'application du système d'exploitation par défaut. Vous pouvez configurer les formats qui doivent s'ouvrir avec

la visionneuse interne via *Préférences → Intervace → Comportement*. Si un livre a plus d'un format, vous pouvez voir un format particulier en faisant un clic droit sur le bouton.

1.1.5 Envoyer vers le périphérique

L'action *Envoyer vers le périphérique* dispose de huit variations, accessibles en faisant un clic droit sur le bouton.

1. **Envoyer vers la mémoire principale** : Les livres sélectionnés sont transférés vers la mémoire principale du lecteur de livre numérique.
2. **Envoyer vers la carte mémoire (A)** : Les livres sélectionnés sont transférés vers la carte de stockage (A) du lecteur de livre numérique.
3. **Envoyer vers la carte mémoire (B)** : Les livres sélectionnés sont transférés vers la carte de stockage (B) du lecteur de livre numérique.
4. **Envoyer un format spécifique vers** : Les livres sélectionnés sont transférés vers l'emplacement de stockage sélectionné du périphérique, dans le format que vous spécifiez.
5. **Éjecter le périphérique** : Détache le périphérique de calibre.
6. **Indiquer l'action par défaut pour envoyer au dispoitif ** : Vous permet de spécifier laquelle des options, 1 à 5 ci-dessus ou 7 ci-dessous, sera l'action par défaut lorsque vous cliquez sur le bouton principal.
7. **Envoyer et effacer de la bibliothèque** : Les livres sélectionnés sont transférés vers l'emplacement de stockage sélectionné sur le périphérique, et puis **supprimées** de la bibliothèque.
8. **Récupérer les annotations (expérimental)** : Transfère des annotations que vous avez faites sur un livre numérique depuis votre périphérique, vers la métadonnée commentaires du livre dans la bibliothèque calibre.

Vous pouvez contrôler le nom du fichier et la structure du dossier des fichiers envoyés au périphérique en mettant en place un modèle dans *Préférences → Importer/exporter → Envoyer des livres aux périphériques*. Voir également *Le langage de modèle calibre* (page 169).

1.1.6 Récupérer des informations

L'action *Récupérer des Actualités* permet de télécharger des actualités de divers sites web et de les convertir en un livre numérique qui peut être lu sur votre liseuse. Normalement, le livre numérique nouvellement créé est ajouté à votre bibliothèque, mais si un lecteur de livre numérique est connecté au moment où se termine le téléchargement, les actualités sont également automatiquement téléchargées sur le lecteur.

L'action *Récupérer des Actualités* utilise une simple recette (10-15 lignes de code) pour chaque site d'actualité. Pour apprendre à créer des recettes pour vos propres sources d'informations, voir *Ajouter votre site web favori d'actualités* (page 33).

L'action *Récupérer des informations* a trois variations accessibles en faisant un clic droit sur le bouton.

1. **Planifier le téléchargement des informations** : Permet de programmer le téléchargement de vos sources d'actualités sélectionnées à partir d'une liste de centaines disponibles. La planification peut être réglée de manière particulière pour chaque source d'actualités que vous sélectionnez et la planification est flexible vous permettant de sélectionner certains jours de la semaine ou une fréquence de jours entre les téléchargements.
2. **Ajouter une source personnalisée d'actualités** : Vous permet de créer une recette simple pour télécharger des nouvelles d'un site d'information personnalisé auquel vous souhaitez accéder. La création de la recette peut être aussi simple que de spécifier un flux d'actualités RSS, ou vous pouvez être plus normatif en créant un code sur base Python pour la tâche. Pour plus d'information voir *Ajouter votre site web favori d'actualités* (page 33).
3. **Télécharger toutes les sources d'actualités planifiées** : entraîne que calibre se met immédiatement à télécharger toutes les sources d'information que vous avez programmées.

1.1.7 Bibliothèque

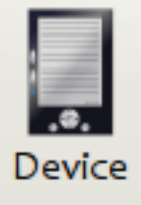
L'action *Bibliothèque* vous permet de créer, de basculer entre, de renommer ou de supprimer une bibliothèque. calibre vous permet de créer autant de bibliothèques que vous le souhaitez. Vous pourriez par exemple, créer une bibliothèque fiction, une bibliothèque non-fiction, une bibliothèque de langue étrangère, une bibliothèque projet, ou toute structure qui conviendrait à vos besoins. Les bibliothèques sont les plus hautes structures organisationnelles dans calibre. Chaque bibliothèque a son propre ensemble de livres, étiquettes, catégories et emplacement de stockage de base.

1. **Basculer/créer une bibliothèque…**. Vous permet de ; a) vous connecter à une bibliothèque calibre préexistante à un autre emplacement, b) créer une bibliothèque vide à un nouvel emplacement ou, c) déplacer la bibliothèque existante à un emplacement nouvellement spécifié.
2. **Bascule rapide** : Vous permet de basculer entre des bibliothèques qui ont été enregistrée ou créée dans calibre.
3. **Renommer un bibliothèque** : Vous permet de renommer une Bibliothèque.
4. **Lire un livre au hasard** : Choisit un livre au hasard dans la bibliothèque pour vous
5. **Supprimer une bibliothèque** : Vous permet de dés-enregistrer une bibliothèque de calibre.
6. **Exporter/importer toutes les données de calibre** : Vous permet soit d'exporter des données de calibre pour la migration vers un nouvel ordinateur, soit d'importer des données exportées précédemment.
7. **<library name>** : Les actions 7, 8, etc…. vous permettent de basculer immédiatement entre plusieurs bibliothèques que vous avez créées ou auxquelles vous vous êtes connecté. Cette liste ne contient que les 5 bibliothèques les plus utilisées. Pour la liste complète, utilisez le menu Changement rapide.
8. **Maintenance de la bibliothèque** : Vous permet de vérifier les problèmes de consistance de la bibliothèque courante et restaure la base de donnée de la bibliothèque depuis des sauvegardes.

Note : Les métadonnées concernant vos livres numériques, par ex. titre, auteur, et étiquettes sont stockées dans un seul fichier dans votre répertoire de bibliothèque calibre appelé metadata.db. Si ce fichier se corrompt (un événement très rare), vous perdez les métadonnées. Heureusement, calibre sauvegarde automatiquement les métadonnées pour chaque livre individuel dans le répertoire du livre en tant que fichier OPF. En utilisant l'action Restaurer la base de données sous Maintenance de la bibliothèque décrite ci-dessus, calibre peut reconstruire le fichier metadata.db depuis les fichiers individuels OPF pour vous.

Vous pouvez copier ou déplacer des livres entre les différentes bibliothèques (une fois que vous avez installé plus d'une bibliothèque) en faisant un clic-droit sur le livre et en sélectionnant l'action *Copier dans la bibliothèque*.

1.1.8 Périphérique connecté



L'action *Périphérique* vous permet de consulter les livres de la mémoire principale ou des cartes de stockage de votre périphérique de lecture, ou d'éjecter le périphérique (le détacher de calibre). Cette icône apparaît automatiquement dans la barre d'outils principale de calibre lorsque vous connectez un périphérique pris en charge. Vous pouvez cliquer sur l'icône pour voir les livres qui sont sur votre périphérique. Vous pouvez également faire glisser et déposer des livres de votre bibliothèque calibre sur l'icône Périphérique pour les transférer sur votre liseuse. Inversement, vous pouvez glisser et déposer des livres à partir de votre périphérique sur l'icône de la bibliothèque de la barre d'outils pour transférer des livres à partir de votre périphérique vers la bibliothèque calibre.

1.1.9 Enregistrer sous...

L'action *Enregistrer sous…* a cinq variations, accessibles en faisant un clic droit sur le bouton.

1. **Enregistrer sous…** : sauve les livres sélectionnés vers le disque organisés en dossiers. La structure du dossier ressemble à

```
Author_(sort)
    Title
        Book Files
```

Vous pouvez contrôler le nom du fichier et la structure des répertoires des fichiers lors de la copie sur le disque en mettant en place un modèle dans *Préférences → Importer/exporter → Sauvegarde de livres sur le disque*. Voir aussi *Le langage de modèle calibre* (page 169).

2. **Enregistrer sur le disque dans un seul dossier** : Enregistre les livres sélectionnés vers le disque dans un dossier unique.

Pour 1. et 2., tous les formats disponibles, ainsi que les métadonnées, sont stockés sur le disque pour chaque livre sélectionné. Les métadonnées sont stockées dans un fichier OPF. Livres enregistrés peuvent être réimportés dans la bibliothèque sans aucune perte d'information en utilisant l'action *Ajouter des livres* (page 4).

3. **Enregistrer seulement le format *<votre format préféré>* sur le disque** : Enregistre les livres sélectionnés sur le disque dans la structure de dossier comme montrée en (1), mais seulement pour votre format de livre numérique préféré. Vous pouvez définir votre format préféré dans *Préférences → Interface → Comportement → Format de sortie préféré*.

4. **Enregistrer seulement le format *<votre format préféré>* sur le disque dans un dossier unique** : Enregistre les livres sélectionnés sur le disque dans un dossier unique, mais seulement dans votre format de livre numérique préféré. Vous pouvez définir votre format préféré dans *Préférences → Interface → Comportement → Format de sortie préféré*.

5. **Enregistrer un seul format vers le disque…** : Enregistre les livres sélectionnés sur le disque dans la structure de dossier comme montrée en (1), mais seulement dans le format que vous sélectionnez dans

la liste contextuelle.

1.1.10 Connecter/partager

L'action *Connecter/partager* vous permet de vous connecter manuellement à un périphérique ou un dossier sur votre ordinateur. Il vous permet également de mettre en place l'accès à votre bibliothèque calibre via un navigateur web ou par courriel.

L'action *Connecter/partager* a quatre variations, accessibles en faisant un clic-droit sur le bouton.

1. **Connectez au dossier** : Vous permet de vous connecter à n'importe quel dossier de votre ordinateur comme s'il s'agissait d'un périphérique et d'utiliser pour ce dossier toutes les fonctionnalités que calibre utilise avec les périphériques. Utile si votre périphérique ne peut pas être pris en charge par calibre, mais est disponible en tant que disque USB.
2. **Démarrer le Serveur de contenu** : Démarre le serveur web intégré de calibre. Au démarrage, votre bibliothèque calibre sera accessible via un navigateur Web à partir d'Internet (si vous le voulez). Vous pouvez configurer la façon dont le serveur Web est accessible en définissant des préférences dans *Préférences → Partager → Partager via Internet*
3. **Configurer le partage des livres par courriel** : Permet de partager des livres et des fils d'information par courriel. Après la mise en place des adresses courriel pour cette option, calibre enverra les mises à jour des actualités et les mises à jour de livres au carnet d'adresses de messagerie saisies. Vous pouvez configurer la façon dont calibre envoie un courriel en définissant des préférences dans *Préférences → Partager → Partager des livres par courriel*. Une fois que vous avez mis en place une ou plusieurs adresses courriel, cette entrée de menu sera remplacée par des entrées de menu pour envoyer des livres aux adresses de messagerie configurées.

1.1.11 Supprimer des livres

L'action *Supprimer des livres* **supprime les livres de manière permanente** aussi utilisez-la avec précaution. Il est *sensible au contexte*, c'est à dire qu'il dépend de quel *catalogue* (page 11) vous avez sélectionné. Si vous avez sélectionné *Bibliothèque*, les livres seront retirés de la bibliothèque. Si vous êtes sur la liste des livres de votre liseuse, les livres seront retirés du périphérique. Pour supprimer uniquement un format particulier pour un livre donné, utiliser l'action *Modifier les métadonnées* (page 5). Supprimer des livres a également cinq variations qui peuvent être accessibles en faisant un clic-droit sur le bouton.

1. **Supprimer les livres sélectionnés** : Vous permet de supprimer **définitivement** tous les livres qui sont sélectionnés dans la liste de livres.
2. **Supprimer les fichiers d'un format spécifique pour les livres sélectionnés …** : Permet de supprimer **définitivement** un fichier de livre numérique d'un format spécifié pour ceux qui sont sélectionnés dans la liste de livres.

3. **Supprimer tous les formats pour les livres sélectionnés, excepté…** : Vous permet de supprimer **définitivement** les fichiers de livres numériques de n'importe quel format sauf celui spécifié pour ceux qui sont sélectionnés dans la liste de livres.
4. **Supprimer tous les formats des livres sélectionnés** : Vous permet de supprimer **définitivement** tous les formats pour les livres qui sont sélectionnés dans la liste de livres. Seules les métadonnées seront conservées.
5. **Supprimer les couvertures des livres sélectionnés** : Vous permet de supprimer **définitivement** des fichiers de couverture des livres qui sont sélectionnés dans la liste de livres.
6. **Supprimer les livres correspondants du périphérique** : Vous permet de supprimer les livres numériques d'un périphérique connecté qui correspondent à ceux sélectionnés dans la liste de livres,
7. **Restaurer des livres récemment supprimés** : Permet d'annuler la suppression de livres ou de formats.

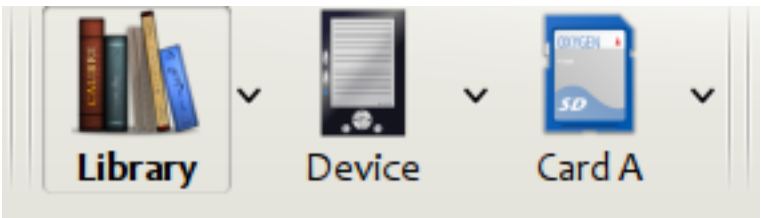
Note : Notez que lorsque vous utilisez *Supprimer des livres* pour supprimer des livres de votre bibliothèque calibre, l'enregistrement du livre est supprimé, mais les livres sont temporairement stockés, pendant quelques jours, dans un dossier poubelle. Vous pouvez annuler la suppression en faisant un clic droit sur le bouton *Supprimer des livres* et en choisissant de *Restaurer des livres récemment supprimés*.

1.2 Préférences

L'action *Préférences* vous permet de modifier de diverses façons la manière dont calibre fonctionne. Elle dispose de quatre variations, accessibles en faisant un clic droit sur le bouton.

1. **Préférences** : vous permet de modifier de diverses façons la manière dont calibre fonctionne. Cliquer sur le bouton effectue aussi cette action.
2. **Démarrer l'assistant de bienvenue** : Vous permet de démarrer l'Assistant de bienvenue qui apparaît la première fois que vous avez démarré calibre.
3. **Obtenir des extensions pour améliorer calibre** : Ouvre une nouvelle fenêtre qui affiche les extensions pour calibre. Ces extensions sont développées par des parties tierces pour étendre les fonctionnalités de calibre.
4. **Redémarrer en mode dépannage** : Vous permet d'activer mode de dépannage qui peut assister les développeurs de calibre à résoudre des problèmes que vous rencontrez avec le programme. Pour la plupart des utilisateurs ceci devraient rester désactivé à moins d'une instruction d'un développeur pour l'activer.

1.3 Catalogues



Un *catalogue* est une collection de livres. calibre peut gérer deux types de catalogues différents :

1. **Bibliothèque** : C'est une collection de livres stockés dans votre bibliothèque calibre sur votre ordinateur.
2. **Périphérique** : C'est une collection de livres stockés dans votre lecteur de livre numérique. Elle sera disponible lorsque vous connecterez le lecteur à votre ordinateur.

De nombreuses opérations, telles que l'ajout des livres, la suppression, la visualisation, etc., sont sensibles au contexte. Ainsi, par exemple, si vous cliquez sur le bouton *Visualiser* quand vous avez le catalogue du **Périphérique** sélectionné, calibre va ouvrir le fichier sur le périphérique. Si vous avez sélectionné le catalogue de la **Bibliothèque** sélectionnée, le fichier de votre bibliothèque calibre sera ouvert à la place.

1.4 Rechercher et trier

L'option rechercher et trier vous permet d'effectuer un certain nombre d'actions puissantes sur votre collection de livres.

- Vous pouvez les trier par titre, auteur, date, note, etc. en cliquant sur les titres des colonnes. Vous pouvez également faire des tri sur plusieurs clés, c'est à dire un tri sur plusieurs colonnes. Par exemple, si vous cliquez sur la colonne titre, puis la colonne auteur, le livre sera trié par auteur, puis toutes les entrées pour un même auteur seront triées par titre.
- Vous pouvez rechercher un livre ou un ensemble de livres en utilisant la Barre de recherche. Plus de détails ci-dessous à ce propos.
- Vous pouvez rapidement et facilement modifier les métadonnées en sélectionnant l'entrée que vous souhaitez modifier dans la liste et en appuyant sur la touche `E`.
- Vous pouvez effectuer des *Actions* (page 4) sur des ensembles de livres. Pour sélectionner de multiples livres vous pouvez soit :
  - Maintenir la touche `Ctrl` enfoncée et cliquer sur tous les livres que vous voulez sélectionner
  - Maintenir la touche `Shift` enfoncée et cliquer sur le premier puis sur le dernier livre d'un éventail de livres que vous voulez sélectionner.
- Vous pouvez configurer quels champs (colonnes) vous souhaitez afficher en utilisant le dialogue *Préférences* (page 11).

- Pour effectuer un tri complexe basé sur plusieurs colonnes, ajoutez l'outil *Classement par* à une barre d'outils via *Préférences → Barres d'outils et menus*.

1.5 L'interface de recherche

Vous pouvez rechercher toutes les métadonnées des livres en saisissant des termes de recherche dans la barre de recherche. Par exemple

```
Asimov Foundation format:lrf
```

Cela correspondra à tous les livres dans votre bibliothèque qui ont les mots Asimov et Fondation dans leurs métadonnées et qui sont disponibles dans le format LRF. Quelques exemples de plus :

```
author:Asimov and not series:Foundation
title:"The Ring" or "This book is about a ring"
format:epub publisher:feedbooks.com
```

Les types de recherche

Il existe quatre types de recherche : *contains*, *equality*, *regular expression* (voir regular expressions[2]), et *character variant*. Vous choisissez le type de recherche avec un caractère préfixe.

*Les recherches "contains" *

Les recherches sans caractère de préfixe sont *contains* et sont par défaut insensibles à la casse. Un élément correspond si la chaîne de recherche apparaît n'importe où dans les métadonnées indiquées. Vous pouvez rendre les recherches *contains* sensibles à la casse en cochant l'option *Recherche sensible à la casse* dans *Préférences / Recherche*. Si l'option de recherche *Les caractères non accentués correspondent aux caractères accentués et la ponctuation est ignorée* est cochée, alors un caractère correspondra à toutes ses variantes (par exemple, *e* correspond à *é*, *è*, *ê*, et *ë*) et toute la ponctuation et les espaces seront ignorés. Par exemple, si l'option *Les caractères accentués correspondent à …* est cochée, les deux titre s de livres suivants seront pris en compte :

1. Big, Bothéred, and Bad
2. Big Bummer

dès lors ces recherches trouvent :

- `title:"er"` correspond aux deux ("e" correspond à la fois à "é" et à "e").
- `Title:"g "` correspond aux deux car les espaces sont ignorés.
- `title:"g,"` correspond aux deux car la virgule est ignorée.
- `title:"gb"` correspond aux deux car ", " est ignoré dans le livre 1 et les espaces sont ignorés dans le livre 2.
- `Title:"g b"` correspond aux deux car la virgule et l'espace sont ignorés.
- `title :"db"` correspond à #1 parce que l'espace dans "and Bad" est ignoré.
- `Title:","` correspond aux deux (il correspond en fait à tous les livres) car les virgules sont ignorées.

Si l'option *Les caractères non accentués correspondent …* n'est pas cochée, les variantes de caractères, la ponctuation et l'espacement sont tous significatifs.

Vous ne pouvez définir qu'une seule des options de recherche « Recherche sensible à la casse » et « Les caractères non accentués correspondent aux caractères accentués et la ponctuation est ignorée ».

Les recherches "equality"

[2] https://en.wikipedia.org/wiki/Regular_expression

Les recherches par égalité sont indiquées en faisant précéder la chaîne de recherche d'un signe égal (=). Par exemple, la requête `tag :"=science"` correspondra à *science*, mais pas à *science fiction* ou *hard science*`. Les variantes de caractères sont importantes : *é* ne correspond pas à *e*.

Deux variantes de recherches d'égalité sont utilisées pour les éléments hiérarchiques (par exemple, A.B.C) : les recherches de préfixes hiérarchiques et les recherches de composants hiérarchiques. La première, indiquée par un seul point après le signe égal (`=.`) correspond aux parties initiales d'un élément hiérarchique. La seconde, indiquée par deux points après le signe égal (`=..`), correspond à un nom interne de l'élément hiérarchique. Exemples, en utilisant l'étiquette « Histoire.Militaire.WWII » comme valeur :

- `tags:"=.Histoire"` : True. `Histoire` est un préfixe de l'étiquette.
- `tags:"=.Histoire.militaire"` : True. `Histoire.militaire` est un préfixe de l'étiquette.
- `tags:"=.Histoire.Militaire.WWII"` : True. `Histoire.Militaire.WWII` est un préfixe de l'étiquette, bien qu'impropre.
- `tags:"=.Militaire"` : True. `Militaire` est un préfixe de l'étiquette.
- `tags :"=.WWII"` : False. `WWII` n'est pas un préfixe d'étiquette.
- `tags:"=..Histoire"` : True. La hiérarchie contient la valeur `Histoire`.
- `tags:"=..Militaire"` : True. La hiérarchie contient la valeur `Militaire`.
- `tags:"=..WWII"` : True. La hiérarchie contient la valeur `WWII`.
- `tags:"=..Militaire.WWII"` : False. La recherche `..` recherche des valeurs uniques.

Recherches "Expressions régulières"

Les recherches par expression régulière sont indiquées en préfixant la chaîne de recherche par un tilde (~). Toute expression régulière compatible avec Python[3] peut être utilisée. Les barres obliques inverses utilisées pour échapper aux caractères spéciaux dans les expressions régulières doivent être doublées car les barres obliques inverses simples seront supprimées lors de l'analyse des requêtes. Par exemple, pour faire correspondre une parenthèse littérale, vous devez saisir `\\(` ou bien utiliser les *super-guillemets* ; (voir ci-dessous). . Les recherches par expression régulière sont des recherches de type « contains », sauf si l'expression est ancrée. Les variantes de caractères sont significatives : `~e` ne correspond pas à `é`.

Recherches de "variantes de caractères"

Les recherches de variantes de caractères sont indiquées en faisant précéder la chaîne de recherche d'un signe d'insertion (^). Cette recherche est similaire à la recherche *contains* (ci-dessus) sauf que :

- la casse des lettres est toujours ignorée.
- les variantes de caractères correspondent toujours les unes aux autres.
- la ponctuation et les espaces sont toujours significatifs.

Les options de recherche *Les caractères non accentués correspondent aux caractères accentués et la ponctuation est ignorée* et *Recherche sensible à la casse* sont ignorées. Elles n'ont aucun effet sur le comportement de cette recherche.

L'exemple suivant compare cette recherche à une recherche avec contenu, en supposant que l'option *Les caractères accentués correspondent…* est cochée (voir ci-dessus), pour les deux mêmes titres de livres :

1. Big, Bothéred, and Bad
2. Big Bummer

alors ces recherches de variantes de caractères trouvent :

- `title:"^er"` correspond aux deux ("e" correspond à la fois à "é" et à "e")
- `title:"^g"` correspond aux deux

[3] https://docs.python.org/library/re.html

- `title:"^g "` correspond au numéro 2 car l'espace est significatif
- `title:"^g,"` correspond au numéro 2 car la virgule est significative
- `title:"^gb"` ne correspond à rien car l'espace et la virgule sont significatifs
- `title:"^g b"` correspond au numéro 2 car la virgule est significative
- `title:"^db"` ne correspond à rien
- `title:"^,"` correspond au n°1 (au lieu de tous les livres) car la virgule est significative

Syntaxe des expressions de recherche

Une *expression de recherche* est une séquence de *termes de recherche* éventuellement séparés par les opérateurs `AND` et `OR`. Si deux termes de recherche sont présents sans opérateur de séparation, le terme `AND` est supposé. L'opérateur `AND` a priorité sur l'opérateur `OR`; par exemple, l'expression `a OR b AND c` est identique à `a OR (b AND c)`. Vous pouvez utiliser des parenthèses pour modifier la priorité; par exemple `(a OR b) AND c` pour que l'opérateur `OR` soit évalué avant l'opérateur `AND`. Vous pouvez utiliser l'opérateur `NOT` pour annuler (inverser) le résultat de l'évaluation d'une expression de recherche. Exemples :

- `not tag:foo` trouve tous les livres qui ne contiennent pas la balise `foo`
- `NOT (author:Asimov OR author:Weber)` trouve tous les livres qui ne sont pas écrits par Asimov ou Weber.

Les exemples ci-dessus montrent des exemples de *termes de recherche*. Un *terme de recherche* de base est une séquence de caractères ne comprenant pas d'espaces, de guillemets (`"`), de barres obliques inversées (`\`), ou de parenthèses (`( )`). Il peut éventuellement être précédé d'un spécificateur de nom de colonne : le *nom de recherche* d'une colonne suivi de deux points (`:`), par exemple `author:Asimov`. Si un terme de recherche doit contenir un espace, alors le terme entier doit être mis entre guillemets, comme dans `title : "The Ring"`. Si le terme recherché doit contenir des guillemets, ils doivent être *échappés* par des barres obliques inverses. Par exemple, pour rechercher une série nommée *La « Boule » et La « Chaîne »*, utiliser

```
series:"The \"Ball\" and The \"Chain\"
```

Si vous avez besoin d'une barre oblique inversée, ce qui arrive fréquemment dans les recherches par *expressions régulières*, utilisez deux barres obliques inversées (`\\`).

Il est parfois difficile d'obtenir tous les échappements corrects pour que le résultat corresponde à ce que vous voulez, notamment dans les recherches par *expression régulière* et *modèle*. Dans ces cas, utilisez la *super-quote* : `"""séquence de caractères"""`. Les caractères surquotés sont utilisés tels quels : aucun traitement d'échappement n'est effectué.

Plus d'information

Pour rechercher une chaîne de caractères qui commence par un signe égal, un tilde ou un caret, préfixez la chaîne de caractères par une barre oblique inverse.

Mettez les chaînes de recherche entre guillemets (») si la chaîne contient des parenthèses ou des espaces. Par exemple, pour trouver des livres portant l'étiquette `science-fiction`, vous devez rechercher `tag:"=science fiction"`. Si vous recherchez `tag:=science fiction`, vous trouverez tous les livres avec le tag `science` et le mot `fiction` dans toutes les métadonnées.

Vous pouvez facilement construire des requêtes de recherche avancées en utilisant le *Boîte de dialogue de recherche avancée* accessible en cliquant sur le bouton à gauche de la boîte de recherche.

Les champs disponibles pour la recherche sont : `tag`, `title`, `author`, `publisher`, `series`, `series_index`, `rating`, `cover`, `comments`, `format`, `identifiers`, `date`, `pubdate`, `search`, `size`, `vl` et les colonnes personnalisées. Si un périphérique est connecté, le champ `ondevice` devient disponible, en recherchant la vue bibliothèque de calibre. Pour trouver le nom de recherche (appelée actuellement la

consultation de nom) pour une colonne personnalisée, placez votre souris au dessus de l'entête de colonne dans la vue bibliothèque.

Dates

La syntaxe pour rechercher des dates est :

```
pubdate:>2000-1 Will find all books published after Jan, 2000
date:<=2000-1-3 Will find all books added to calibre before 3 Jan, 2000
pubdate:=2009 Will find all books published in 2009
```

Si la date est ambiguë alors la date actuelle en local est utilisée pour la comparaison de date. Par exemple, dans une date locale mm/dd/yyyy 2/1/2009 est interprétée comme 1 Fév 2009. Dans une date locale dd/mm/yyyy elle est interprétée comme 2 Jan 2009. Certaines chaines spéciales de date sont disponibles. La chaîne `today` traduite en date du jour, quelle qu'elle soit. Les chaînes `yesterday` et `thismonth` (ou leur équivalent traduit dans la langue courante) fonctionne aussi. En complément, la chaîne `daysago` (également traduite) peut être utilisée pour comparer une date à un certain nombre de jours avant. Par exemple

```
date:>10daysago
date:<=45daysago
```

Pour éviter des problèmes potentiels avec les chaines traduites lors de l'utilisation d'une version non Anglaise de calibre, les chaînes `_today`, `_yesterday`, `_thismonth`, and `_daysago` seront toujours disponibles. Elles ne sont pas traduites.

Recherche de dates et de valeurs numériques avec des comparaisons relationnelles

Les champs date et numérique supportent les opérateurs relationnels = (égal), > (plus grand que), >= (plus grand que ou égal à), < (plus petit que), <= (plus petit que ou égal à), et `!=` (différent de). Les champs de notation sont considérés être numérique. Par exemple, la recherche `rating:>=3` trouvera les livres notés 3 ou supérieur.

Vous pouvez rechercher des livres qui ont un format d'une certaine taille comme ceci :

- `size:>1.1M` trouvera les livres dont le format est supérieur à 1.1MB
- `size:<=1K` trouvera les livres dont le format est inférieur ou égal à 1KB

Vous pouvez rechercher le nombre d'éléments dans des champs à valeurs multiples tels que les étiquettes en utilisant le caractère « #» puis en utilisant la même syntaxe que les champs numériques. Par exemple, pour trouver tous les livres ayant plus de 4 étiquettes, utilisez `tags:#>4`. Pour trouver tous les livres ayant exactement 10 étiquettes, utilisez `tags:#=10`.

Indices des séries

Les indices des séries peuvent faire l'objet d'une recherche. Pour les séries standard, le nom de recherche est `series_index`. Pour les colonnes personnalisées de séries, utilisez le nom de recherche de la colonne suivit par _index. Par exemple, pour rechercher les indices pour une colonne de séries personnalisée `#my_series`, vous devrez utiliser le nom de recherche `#my_series_index`. Les indices de séries sont des nombres, donc vous pouvez utiliser les opérateurs relationnels décrits plus haut.

Recherches sauvegardées

Le champ spécial `search` est utilisé pour *Recherches sauvegardées* (page 19). Si vous enregistrez une recherche avec le nom « Livres de mon conjoint », vous pouvez entrer `search:"Livres de mon conjoint"` dans la Barre de recherche pour réutiliser la recherche enregistrée. Plus d'informations sur la sauvegarde des recherches ci-dessous.

Bibliothèques virtuelles

Le champ spécial `vl` est utilisé pour chercher des livres dans une Bibliothèque Virtuelle. Par exemple, `vl:Lu` trouvera tous les livres dans la Bibliothèque Virtuelle *Lu*. La recherche `vl:Lu and vl:"Science Fiction"` trouvera tous les livres qui sont tant dans les Bibliothèques Virtuelles *Lu* que *Science Fiction*. La valeur suivant `vl:` doit être le

nom d'une Bibliothèque Virtuelle. Si le nom de la Bibliothèque Virtuelle contient des espaces entourez-le alors avec des guillemets.

Si un champ a une valeur

Vous pouvez rechercher l'absence ou la présence d'une valeur pour un champ en utilisant « true » et « false ». Par exemple :

- `cover:false` trouve tous les livres sans couverture
- `series:true` trouve tous les livres qui font partie d'une série
- `series:false` trouve tous les livres qui ne sont pas dans une série
- `comments:false` trouve tous les livres avec un commentaire vide
- `formats:false` trouve tous les livres sans fichier de livre (enregistrements vides).

Colonnes personnalisées oui/non

Recherche de colonnes personnalisées Oui/non pour `false`, `empty`, ou `blank` trouvera tous les livres avec des valeurs indéfinies dans la colonne. Faire une recherche `true` trouvera les livres qui n'ont pas de valeurs indéfinies dans la colonne. Faire une recherche `yes` ou `checked` trouvera tous les livres avec `Yes` dans la colonne. Faire une recherche `no` ou `unchecked` trouvera tous les livres avec `No` dans la colonne. Notez que les mots `yes`, `no`, `blank`, `empty`, `checked` et `unchecked` sont traduits ; vous pouvez utiliser soit utiliser le mot équivalent dans la langue courante ou le mot anglais. Les mots `true` et `false` et les valeurs spéciales `_yes`, `_no`, et `_empty` ne sont pas traduits.

Identifiants

Les identifiants (par exemple, ISBN, DOI, LCCN, etc.) utilisent une syntaxe étendue. Un identificateur a la forme `type:valeur`, comme dans `isbn:123456789`. La syntaxe étendue vous permet de spécifier indépendamment le type et la valeur à rechercher. Les deux parties de la requête, type et valeur, peuvent utiliser n'importe quels types de recherche. Exemples :

- `identifiers:true` trouvera les livres avec n'importe quel identifiant.
- `identifiers:false` trouvera les livres sans identifiant.
- `identifiers:123` recherchera des livres de n'importe quel type qui ont une valeur contenant *123*.
- `identifiers:=123456789` recherchera des livres de n'importe quel type qui ont une valeur égale à *123456789*.
- `identifiers:=isbn:` et `identifiers:isbn:true` trouvera les livres avec un type égal à ISBN et ayant n'importe quelle valeur
- `indentifiers:=isbn:false` trouvera les livres avec un type n'équivalant pas à ISBN.
- `identifiers:=isbn:123` trouvera les livres avec un type égal à ISBN ayant une valeur contenant *123*.
- `identifiers:=isbn:=123456789` trouvera les livres avec un type égal à ISBN et ayant une valeur égale à *123456789*.
- `identifiers:i:1` trouvera les livres avec un type contenant un *i* ayant une valeur contenant un *1*.

Catégories visibles dans le Navigateur d'étiquettes

La recherche `in_tag_browser:true` trouve tous les livres qui sont dans les catégories (tags, auteurs, etc.) actuellement affichées dans le *Navigateur d'étiquettes*. Ceci est utile si vous avez défini les deux préférences *Préférences → Apparence et présentation → Navigateur d'étiquettes → Cacher les catégories vides* et *Trouver et afficher tous les éléments qui correspondent*. Une fois ces deux préférences définies, l'exécution de la commande *Trouver* dans le *Navigateur d'étiquettes* n'affiche que les catégories contenant des éléments correspondant à la commande *Trouver*. Ensuite, la recherche `in_tag_browser:true` trouve en plus les livres avec ces catégories/élements. Vous pouvez facilement lancer cette recherche en appuyant sur la touche `Ctrl+Alt+Shift+F` ou en cliquant sur le bouton de configuration dans le *Navigateur d'étiquettes* et en choisissant l'entrée *Afficher uniquement les livres dont les catégories sont visibles*.

Recherche à l'aide de modèles

Vous pouvez effectuer une recherche en utilisant un modèle dans *Le langage de modèle calibre* (page 169) au lieu d'un champ de métadonnées. Pour ce faire, vous saisissez un modèle, un type de recherche et la valeur à rechercher. La syntaxe est la suivante

```
template: (the template) #@#: (search type) : (the value)
```

Le `template` est n'importe quel modèle de langage calibre valide. Le `search type` doit être l'un des suivants : `t` (recherche de texte), `d` (recherche de date), `n` (recherche numérique), ou `b` (ensemble/non ensemble (booléen)). La `valeur` est ce que vous voulez, et peut utiliser les *types de recherche* (page 13) décrits ci-dessus pour les différents types de recherche. Vous devez citer la chaîne de recherche entière s'il y a des espaces dans celle-ci.

Exemples :

- `template:"program : connected_device_name('main')#@#:t:kindle"` – est true lorsque le périphérique `kindle` est connecté.
- `template:"program: select(formats_sizes(), 'EPUB')#@#:n:>1000000"` – trouve des livres avec des fichiers EPUB supérieurs à 1 Mo.
- `template:"program: select(formats_modtimes('iso'), 'EPUB')#@#:d:>10daysago"` – trouve des livres avec des fichiers EPUB plus récents qu'il y a 10 jours.
- `Template:"""program: book_count('tags:^"' & $series & '"', 0) != 0#@#:n:1"""` – trouve tous les livres contenant le nom de la série dans les étiquettes. Cet exemple utilise des guillemets superposés car le modèle utilise à la fois des guillemets simples (`'`) et des guillemets doubles (`"`) lors de la construction de l'expression de recherche.

Vous pouvez construire facilement des modèles de recherche en utilisant le *Dialogue de recherche avancée* accessible en cliquant sur le bouton . Vous pouvez tester les modèles sur des livres spécifiques en utilisant *Testeur de modèle* calibre, qui peut être ajouté aux barres d'outils ou aux menus via *Préférences → Barres d'outils & menus*. Il est également possible de lui attribuer un raccourci clavier via *Préférences → Raccourcis*.

Boîte de dialogue recherche avancée

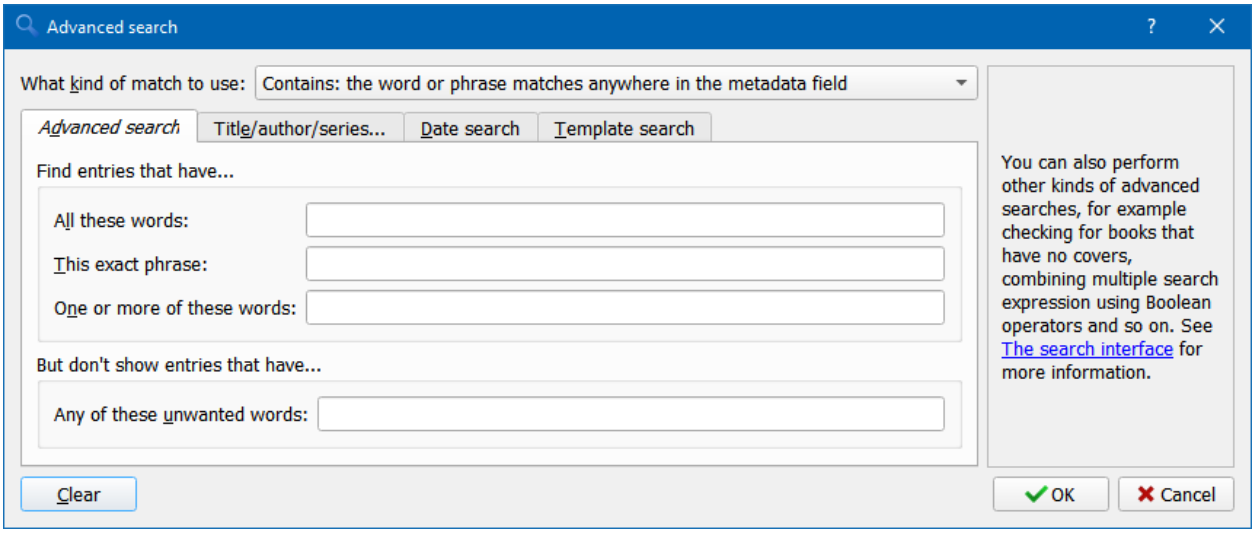


Fig. 1 : *Boîte de dialogue recherche avancée*

1.6 Recherches sauvegardées

calibre vous permet de sauvegarder une recherche fréquemment utilisée sous une dénomination spéciale et ensuite de réutiliser cette recherche avec un simple clic. Pour ce faire, créez votre recherche soit en tapant dans la Barre de recherche ou en utilisant le Navigateur d'étiquettes. Ensuite, tapez le nom que vous souhaitez donner à la recherche dans la zone Recherches Sauvegardées, à droite de la Barre de recherche. Cliquez sur l'icône + à côté de la boîte recherches enregistrées pour sauver la recherche.

Maintenant, vous pouvez accéder à votre recherche sauvegardée dans le Navigateur d'étiquettes sous la rubrique *Recherches sauvegardées*. Un simple clic vous permettra de réutiliser facilement toute recherche complexe, sans avoir à la recréer.

1.7 Recherche dans le texte intégral de tous les livres

calibre peut *optionnellement* indexer le texte intégral des livres de la bibliothèque pour permettre la recherche instantanée de mots dans n'importe quel livre. Pour utiliser cette fonctionnalité, cliquez sur le bouton *FT* situé sur le bord gauche de la barre de recherche.

Ensuite, activez l'indexation pour la bibliothèque actuelle. Une fois l'indexation terminée, vous pouvez rechercher tout le texte de la bibliothèque complète. Lorsque vous ajoutez de nouveaux livres à la bibliothèque, ils sont automatiquement indexés en arrière-plan. Cette recherche vous permet de trouver rapidement tous les livres qui contiennent un mot ou une combinaison de mots. Vous pouvez même rechercher des mots qui apparaissent à proximité d'autres mots, comme le montrent les exemples dans la fenêtre contextuelle de recherche. Notez que cet outil de recherche ne trouvera qu'une seule occurrence de la requête dans un livre particulier, et ne listera pas toutes les occurrences. Pour cela, il est préférable de rechercher à l'intérieur du livre en utilisant le calibre *Visionneuse de livres numériques*.

Vous pouvez réindexer un livre individuel en faisant un clic droit sur le *Panneau de détails du livre* dans calibre et en choisissant *Reindexer ce livre pour une recherche en texte intégral*.

1.8 Bibliothèques virtuelles

Une *Bibliothèque virtuelle* est une manière de prétendre que votre bibliothèque calibre a seulement quelques livres plutôt que sa collection complète. C'est une excellente manière de partitionner votre large collection de livres en des entités plus petites, mieux gérable. Pour apprendre comment créer et utiliser des Bibliothèques virtuelles, voir le tutoriel : *Bibliothèques virtuelles* (page 259).

1.9 Marquage temporaire des livres

Vous pouvez marquer temporairement des ensembles arbitraires de livres. Les livres marqués auront une épingle sur eux et peuvent être trouvés avec la recherche `marked:true`. Pour marquer un livre, appuyez sur `Ctrl+M` ou allez dans *Préférences → Barres d'outils & menus* et ajoutez le bouton *Marquer des livres* à la barre d'outils principale.

Vous pouvez marquer les livres avec une étiquette de texte spécifique en faisant un clic droit sur le bouton *Marquer les livres* et en choisissant *Marquer les livres avec une étiquette de texte*. Les livres marqués avec des étiquettes de texte peuvent être trouvés plus tard en utilisant la recherche `marked :"=le-texte-que-vous-entré"`.

1.10 Deviner les métadonnées à partir des noms de fichier

Normalement, calibre lit les métadonnées à partir du fichier livre, il peut être configuré pour lire les métadonnées à partir du nom de fichier, via *Préférences → Importer/exporter → Ajouter des livres → Lire les métadonnées à partir du contenu du fichier*.

Vous pouvez aussi contrôler comment les métadonnées sont lues à partir du nom de fichier en utilisant les expressions régulières (voir *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225)). Dans la section *Ajouter des livres* de la boîte de configuration, vous pouvez spécifier une expression régulière que calibre utilisera pour essayer et deviner les métadonnées à partir des noms de fichier des livres numériques que vous ajoutez à votre bibliothèque. L'expression régulière par défaut est :

```
title - author
```

c.-à-d., il suppose que tous les caractères jusqu'au premier `` - `` sont le titre du livre et les caractères suivants sont l'auteur du livre. Par exemple, le nom de fichier

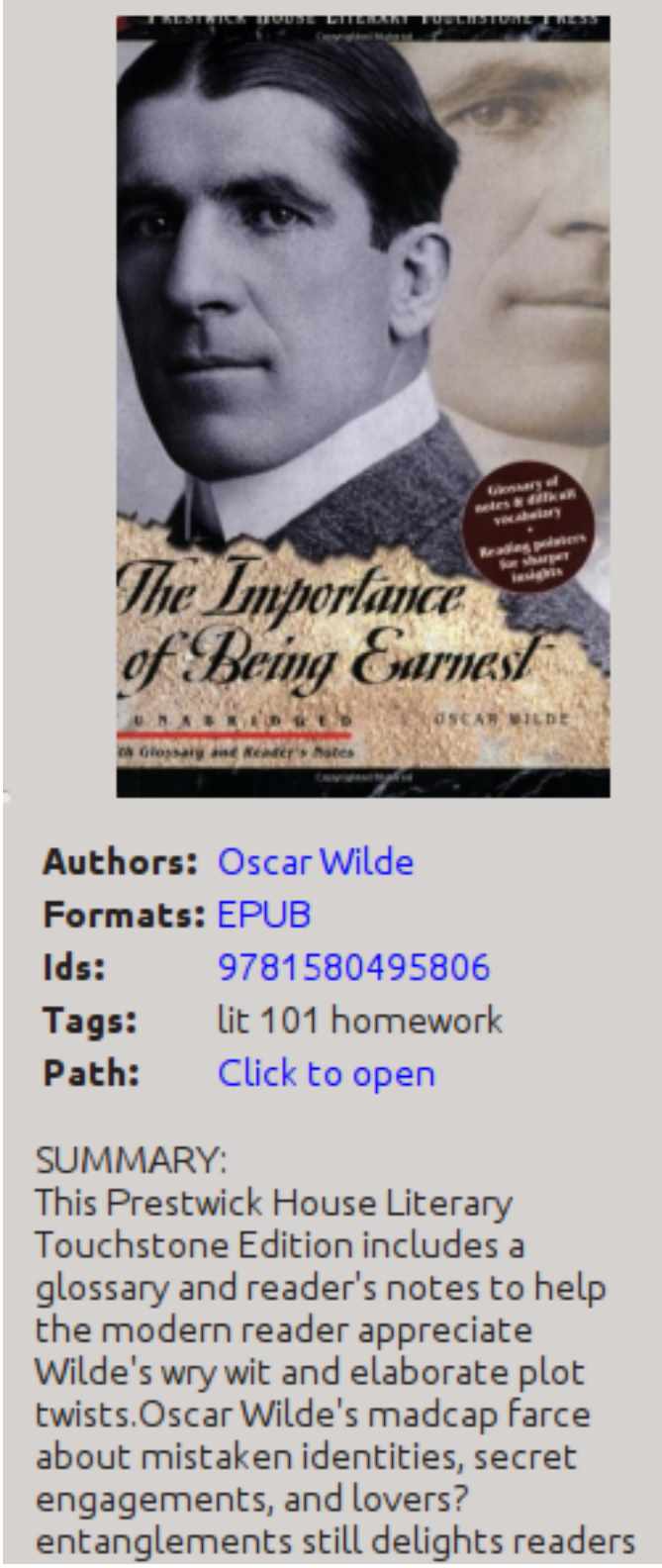
```
Foundation and Earth - Isaac Asimov.txt
```

sera interprété avoir le titre Foundation and Earth et l'auteur : Isaac Asimov

---

Astuce : Si le nom de fichier ne contient pas de trait d'union, l'expression régulière ci-dessus échouera.

1.11 Détails du livre



L'écran Détails du livre affiche la couverture et toutes les métadonnées pour le livre actuellement sélectionné. Il peut être

caché par l'intermédiaire du bouton *Disposition* en bas à droite de la fenêtre principale calibre. Le nom de l'auteur affiché dans le panneau Détail du livre peut être cliqué, il vous mènera vers la page Wikipedia au sujet de l'auteur. Ceci peut être personnalisé en faisant un clic droit sur le nom de l'auteur et en sélectionnant Gérer cet auteur.

Similairement, si vous téléchargez des métadonnées pour le livre, panneau Détail du Livre vous affichera des liens pointant vers les pages web du livre sur Amazon, Worldcat, etc. depuis où les métadonnées ont été téléchargées.

Vous pouvez faire un clic droit sur les formats individuels de livre dans le panneau Détail du livre pour les supprimer, les comparer à leurs versions originales, les sauvegarder sur le disque, les ouvrir avec un programme extérieur etc.

Vous pouvez changer la couverture du livre simplement en glissant et déposant une image sur le panneau Détails du Livre. Si vous voulez utiliser un programme tiers pour éditer l'image de couverture, faites un simple clic droit dessus et choisissez *Ouvrir avec*.

Vous pouvez aussi ajouter des fichiers de livre numérique au livre courant en glissant et déposant les fichiers sur le panneau Détails du livre.

Double cliquer sur le panneau de Détails de livre l'ouvrira dans une fenêtre distincte.

Pour finir, vous pouvez personnaliser exactement quelle information est affichée dans le panneau Détails du livre par l'intermédiaire de *Préférences → Apparence → Détails du livre*.

1.12 Navigateur d'étiquettes

Le Navigateur d'étiquettes vous permet de parcourir facilement votre collection par Auteur/Étiquettes/ Série etc. Si vous cliquez sur n'importe quel élément dans le Navigateur d'étiquettes, par exemple sur le nom d'auteur Isaac Asimov, la liste des livres se limitera à l'affichage des livres de cet auteur. Vous pouvez également cliquer sur les catégories. Par exemple, si vous cliquez sur « Séries » vous affichera tous les livres de n'importe quelle série.

Le premier clic sur un élément permet de limiter la liste des livres à ceux qui contiennent ou correspondent à cet élément. En reprenant l'exemple ci-dessus, en cliquant sur Isaac Asimov vous aurez tous les livres de cet auteur. En cliquant à nouveau sur l'élément, le résultat dépend si l'élément a des sous-éléments ou non (voir les sous-catégories et les éléments hiérarchiques ci-dessous). En continuant l'exemple d'Isaac Asimov en cliquant à nouveau sur Isaac Asimov, vous allez obtenir les livres qui ne sont pas d'Isaac Asimov. Un troisième clic va supprimer la restriction, et vous aurez de nouveau tous les livres. Si vous maintenez enfoncée la touche kbd :*Ctrl* ou kbd :*Maj* et cliquez sur plusieurs éléments, la liste prendra en compte toutes les conditions. Par exemple, vous pourriez maintenir la touche `Ctrl` et cliquez sur les catégories Histoire et Europe pour trouver des livres sur l'histoire européenne. Le Navigateur d'étiquettes fonctionne en construisant des expressions de recherche qui sont entrées automatiquement dans la barre de recherche. En regardant ce que le Navigateur d'étiquettes génère vous pourrez facilement apprendre à construire des expressions de recherche de base.

Les éléments du Navigateur d'étiquettes ont leurs icônes partiellement colorées. La quantité de couleur dépend de la note moyenne des livres dans cette catégorie. Ainsi, par exemple si les livres d'Isaac Asimov ont une moyenne de quatre étoiles, l'icône Isaac Asimov dans le Navigateur d'étiquettes sera colorée au 4/5e. Vous pouvez passer votre souris sur l'icône pour voir la note moyenne.

Les articles des niveaux externes dans le *Navigateur d'étiquettes*, comme Auteurs et Séries, sont appelés catégories. Vous pouvez créer vos propres catégories, appelées *Catégories utilisateur*, qui sont utiles pour organiser les éléments. Par exemple, vous pouvez utiliser l'*Éditeur de catégories utilisateur* (cliquer le bouton *Configurer* en bas à gauche du *Navigateur d'étiquettes* et choisissez *Gérer les auteurs, étiquettes etc → Catégories utilisateur*`) pour créer une Catégorie utilisateur appelée « Auteurs favoris », mettez alors vos éléments favoris dans la catégorie. Les Catégories utilisateur peuvent avoir des sous catégories. Par exemple, la Catégorie utilisateur « Favoris.Auteurs » est une sous catégorie de Favoris. Vous pouvez aussi créer « Favoris.Séries », dans ce cas il y aura deux sous catégories sous Favoris. Les sous catégories peuvent être créées en faisant un clic droit sur une Catégorie utilisateur, choisir *Ajouter une sous catégorie à …*, et entrer le nom de la sous catégorie ; ou en utilisant l'*Éditeur de catégories utilisateur* et en entrant les noms comme l'exemple des Favoris ci-dessus.

Vous pouvez rechercher les Catégories utilisateurs de la même manière que les catégories intégrées, en cliquant dessus. Il y a quatre recherches différentes faits un cycle en cliquant sur :

1. « tout ce qui correspond à l'élément dans la catégorie » indiquée par un seul signe plus vert.
2. « tout ce qui correspond à l'élément de la catégorie ou de ses sous-catégories » indiqué par deux signes plus verts.
3. « tout ce qui ne correspond pas à l'élément dans la catégorie » indiquée par un seul signe moins rouge.
4. « tout ce qui ne correspond pas à l'élément dans la catégorie ou dans ses sous-catégories » indiquées par deux signes moins rouges.

Il est également possible de créer des hiérarchies à l'intérieur de certaines catégories texte comme les étiquettes, les séries et les colonnes personnalisées. Ces hiérarchies apparaissent avec un petit triangle, permettant aux sous éléments d'être masqués. Pour utiliser les hiérarchies d'éléments dans une catégorie, vous devez vous rendre tout d'abord dans *Préférences → Interface → Apparence → Navigateur d'étiquettes* et entrer le(s) nom(s) de catégorie dans le champ « Catégories avec des articles hiérarchiques ». Une fois que c'est fait, les éléments dans cette catégorie qui contiennent des subdivisions seront affichés en utilisant le petit triangle. Par exemple, présumons que vous créez une colonne personnalisée appelée « Genre » et indiquez quelle contient des éléments hiérarchiques. Une fois fait, les éléments tels que Mystère.Thriller et Mystère.Anglais s'afficheront comme Mystère avec le petit triangle à leur suite. Cliquer sur le triangle affichera Thriller et Anglais comme sous éléments. Voir *Gérer des sous groupes de livres, par exemple « genre »* (page 161) pour plus d'information.

Les éléments hiérarchiques (éléments avec des sous-éléments) utilisent les quatre recherches "cliquez dessus" identiques à celles des Catégories utilisateurs. Les éléments qui n'ont pas de sous-éléments utilisent deux des recherches : « tout ce qui correspond » et « tout ce qui ne correspond pas ».

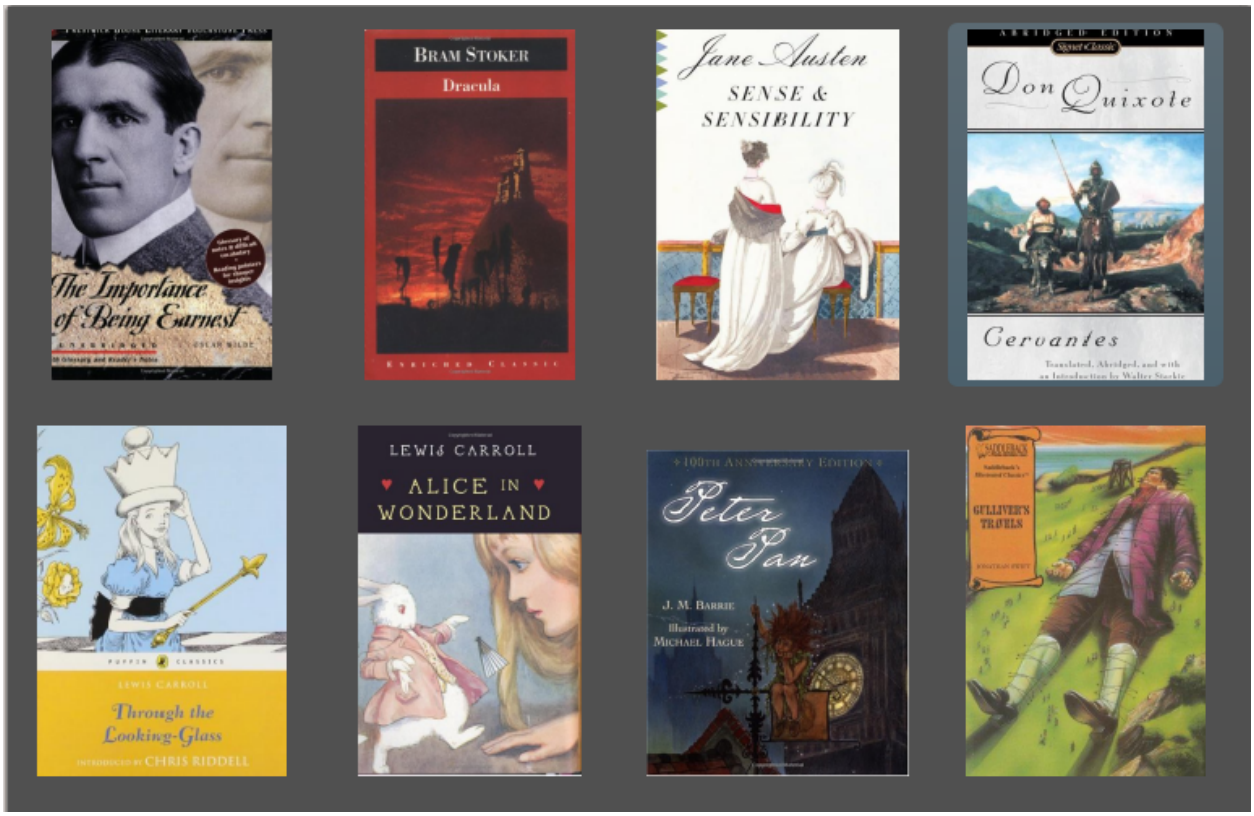
Vous pouvez glisser et déposer des éléments dans le Navigateur d'Étiquettes sur les Catégories d'utilisateurs à ajouter à cette catégorie. Si la source est une Catégorie utilisateur, maintenez la touche `Maj` tout en déplaçant l'article vers la nouvelle catégorie. Vous pouvez aussi glisser et déposer des livres depuis la liste de livres sur les éléments du Navigateur d'étiquettes ; déposer un livre sur des éléments dans le Navigateur d'Étiquettes entraînera que cet élément s'appliquera automatiquement au livre déposé. Par exemple, glisser un livre sur Isaac Asimov établira l'auteur pour ce livre à Isaac Asimov. Déposer le sur l'étiquette Histoire ajoutera l'étiquette Histoire aux étiquettes du livre.

Vous pouvez trouver facilement n'importe quel élément dans le Navigateur d'étiquettes en cliquant le bouton de recherche en bas à droite. En complément, vous pouvez faire un clic droit sur n'importe quel élément et choisir une des diverses opérations. Certains exemples sont de le cacher, de le renommer, ou d'ouvrir une boîte de dialogue « Gérer x » qui vous permet de gérer des éléments de ce type. Par exemple, la boîte de dialogue *Gérer les auteurs* permet de renommer les auteurs et de contrôler comment leurs noms sont triés.

Vous pouvez contrôler comment les éléments sont triés dans le Navigateur d'étiquettes par l'intermédiaire du bouton *Configurer* en bas à gauche du Navigateur d'étiquettes. Vous pouvez choisir de trier par nom, note moyenne ou popularité

(popularité est le nombre de livres avec un élément dans votre bibliothèque ; par exemple, la popularité d'Isaac Asimov est le nombre de livres par Isaac Asimov dans votre bibliothèque).

1.13 Grille de couvertures



calibre peut vous afficher une grille de couvertures à la place de la liste de livres. Si vous préférez plutôt naviguer dans votre collection par couvertures. La *Grille de couvertures* est activée en cliquant sur le bouton *Disposition* grille dans le coin inférieur droit de la fenêtre principale de calibre. Vous pouvez personnaliser la taille des couvertures et l'arrière plan de la *Grille de couvertures* via *Préférences → Interface → Apparence → Grille de couvertures*. calibre peut même afficher n'importe quel champ en dessous des couvertures, comme titre, auteurs ou note ou une colonne personnalisée de votre propre invention.

1.14 Navigateur de couvertures

En complément de la grille de couvertures décrite plus haut, calibre peut aussi afficher les couvertures en une simple ligne. Ceci est activé par l'intermédiaire d'un bouton *Disposition* dans le coin inférieur droit de la fenêtre principale. Dans *Préférences → Apparence → Navigateur de couvertures* vous pouvez changer le nombre de couvertures affichées, et même avoir le navigateur de couverture s'afficher dans une fenêtre distincte.

1.15 Ajouter des notes pour les auteurs, les séries, etc.

Vous pouvez ajouter des notes pour un auteur/une série/une étiquette/un éditeur/etc. à votre bibliothèque calibre. Pour ce faire, faites un clic droit sur le nom de l'auteur dans le *Navigateur d'étiquettes* à gauche ou dans le panneau *Détails du livre* à droite et choisissez *Créer une note* ou *Éditer la note*.

Une simple fenêtre contextuelle vous permet de saisir vos notes en utilisant un formatage de base et en ajoutant des liens et des images. Une fois qu'une note est créée pour un auteur, elle peut être facilement consultée à partir du panneau *Détails du livre* en cliquant sur le petit crayon à côté du nom de l'auteur.

Vous pouvez rechercher toutes les notes de votre bibliothèque en utilisant l'outil *Parcourir les notes* en appuyant sur `Ctrl+Shift+N` ou en l'ajoutant à la barre d'outils via *Préférences → Barres d'outils & menus*.

1.16 Affichage rapide

Parfois vous voulez sélectionner un livre et rapidement obtenir une liste de livres avec la même valeur dans une certaine catégorie (auteurs, étiquettes, éditeur, séries, etc.) comme le livre actuellement sélectionné, mais sans changer la vue actuelle de votre bibliothèque. Vous pouvez faire cela avec Affichage rapide. Affichage rapide ouvre une deuxième fenêtre affichant la liste des livres correspondants à la valeur intéressée. Par exemple, supposons que vous voulez voir une liste de tous les livres avec un ou plus des auteurs du livre actuellement sélectionné. Cliquer dans la cellule de l'auteur qui vous intéresse et appuyez sur la touche "Q" ou cliquez sur l'icône :guilabel :"Affichage rapide" dans la section :guilabel :"Mise en page" de la fenêtre calibre. Une fenêtre ou un panneau s'ouvrira avec tous les auteurs pour ce livre sur la gauche et tous les livres pour l'auteur sélectionné sur la droite.

Quelques exemples des utilisations d'Affichage Rapide : voir rapidement que d'autres livres :

- ont certaine(s) étiquette(s) appliquée(s) au livre actuellement sélectionné,
- sont dans la même série que le livre actuel
- ont les mêmes valeurs dans une colonne personnalisée que le livre actuel
- sont écrits par l'un des mêmes auteurs que le livre actuel
- partager les valeurs dans une colonne personnalisée

Il y deux choix pour où faire apparaître l'information Affichage Rapide :

1. Il peut s'ouvrir « ancré » : au dessus de la fenêtre calibre et il restera ouvert jusqu'à ce que vous le fermiez explicitement.
2. Il peut s'ouvrir « désancré » : comme un panneau dans la section liste de livres de la fenêtre principale de calibre.

Vous pouvez déplacer comme bon vous semble la fenêtre de verrouillé à déverrouillé en utilisant le bouton « Ancrer/Désancrer ».

La panneau Affichage rapide peut être laissé ouvert en permanence, dans ce cas il suivra les mouvements de la liste de livres. Par exemple, si dans la vue bibliothèque calibre vous cliquez sur une colonne de catégorie (étiquettes, séries, éditeur, auteur, etc.) pour un livre, le contenu de la fenêtre Affichage rapide changera pour vous montrer dans le panneau de gauche les éléments dans cette catégorie pour le livre sélectionné (par exemple, les étiquettes pour ce livre). Le premier élément dans cette liste sera sélectionné, et Affichage rapide vous montrera dans le panneau de droite tous les livres dans votre bibliothèque qui utilisent cet valeur. Cliquer sur une valeur différente dans le panneau de gauche pour voir les livres avec cette valeur différente.

Double cliquez sur un livre dans la fenêtre Affichage rapide pour sélectionner ce livre dans la vue bibliothèque. Ceci changera aussi les éléments dans la fenêtre Affichage rapide (panneau de gauche) pour montrer les éléments dans le livre nouvellement sélectionné.

`Shift-` ou `Ctrl-` double clic sur un livre dans la fenêtre Affichage rapide pour ouvrir la boîte de dialogue modifier les métadonnées de ce livre dans la fenêtre calibre. Le livre modifié sera Affiché rapidement quand vous fermerez la boite de dialogue Modifier les métadonnées.

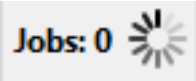
Vous pouvez voir si une colonne peut être Affichée Rapidement en passant votre souris sur l'entête de la colonne et en regardant l'info-bulle pour cet entête. Vous pouvez aussi le savoir en faisant un clic droit sur l'entête de colonne pour voir si l'option « Affichage rapide » apparaît dans le menu, dans un tel cas choisir cette option Affichage rapide est équivalent à appuyer "Q" dans la cellule actuelle.

Options (dans :guilabel :`Préférences->Apparence et présentation->Affichage rapide) :

- Respecte (ou non) la Bibliothèque virtuelle actuelle. Si coché, alors Affichage rapide montre uniquement les livres dans la Bibliothèque virtuelle actuelle. Par défaut : respecte les Bibliothèques virtuelles
- Changer le contenu de la fenêtre Affichage rapide quand la colonne est changée dans la liste de livres en utilisant les touches curseur. Par défaut : ne suit pas les changements effectués avec les touches curseur

- Changer la colonne qui est « Affiché rapidement » quand une cellule dans la fenêtre Affichage rapide est double cliquée. Autrement le livre est changé mais la colonne examinée ne l'est pas. Par défaut : changer la colonne
- Changer la colonne qui est « Affichée rapidement » vers la colonne actuelle quand la touche retour est appuyée dans le panneau Affichage rapide. Autrement le livre est changé mais la colonne qui est examinée ne l'est pas. Par défaut : changer la colonne
- Choisir quelles colonnes sont affichées dans la/le fenêtre/panneau Affichage rapide

1.17 Tâches



Le panneau des Travaux montre le nombre des tâches en cours. Les travaux sont des tâches qui s'exécutent dans un processus séparé. Ils incluent la conversion de livres numériques et les échanges avec votre liseuse. Vous pouvez cliquer sur le panneau des tâches pour accéder à la liste des tâches. Une fois qu'une tâche est terminée vous pouvez voir un journal détaillé de cette tâche en la double cliquant dans cette liste. Ceci est utile pour dépanner des tâches qui ne se sont pas terminées avec succès.

1.18 Raccourcis clavier

calibre a plusieurs raccourcis clavier pour vous faire gagner du temps et du déplacement de souris. Ces raccourcis sont disponibles depuis la liste des livres (quand vous n'êtes pas occupé à éditer les détails d'un livre particulier), et la plupart d'entre eux affectent le titre que vous avez sélectionné. La visionneuse de livre numérique calibre *a ses propres raccourcis* (page 58) qui peuvent être personnalisés en cliquant sur le bouton *Préférences* dans la visionneuse.

Note : Remarque : Les raccourcis clavier calibre ne requièrent pas de touche modificatrice (Command, Option, Control, etc.) sauf si cela est spécialement indiqué. Vous avez seulement besoin d'appuyer la touche de lettre. Par ex. : E pour éditer.

Tableau 1 : Raccourcis clavier pour le programme principal de calibre

Raccourci clavier	Action
F2 (Enter pour macOS)	Editer les métadonnées du champ actuellement sélectionné dans la liste de livres.
A	Ajouter des livres
Shift+A	Ajouter des formats aux livres sélectionnés
C	Convertir les livres sélectionnés
D	Envoyer vers le périphérique
Del	Supprimer les livres sélectionnés
E	Editer les métadonnées des livres sélectionnés
G	Obtenir des livres
I	Afficher Détails du livre
K	Modifier la table des matières

suite sur la page suivante

Tableau 1 – suite de la page précédente

Raccourci clavier	Action
M	Fusionner les enregistrements sélectionnés
Alt+M	Fusionner les enregistrements sélectionnés, garder les originaux
O	Ouvrir le répertoire contenant
P	Polir des livres
S	Enregistrer sous…
T	Editer le livre
V	Visualiser
Shift+	Voir le dernier livre lu
Alt+V/ Cmd+V pour macOS	Visualiser un format spécifique
Alt+Sh	Afficher/Masquer la liste des tâches
Alt+Sh	Afficher/Masquer le Navigateur de Couvertures
Alt+Sh	Afficher/Masquer le panneau des Détails du Livre
Alt+Sh	Afficher/Masquer Le Navigateur d'Étiquette
Alt+Sh	Afficher/Masquer la grille de couvertures
Alt+A	Afficher les livres du même auteur que le livre actuel
Alt+T	Afficher des livres avec les mêmes étiquettes que le livre actuel
Alt+P	Afficher des livres avec le même éditeur que le livre actuel
Alt+Sh	Afficher des livres dans la même série que le livre actuel
/, Ctrl+F	Focalisez la barre de recherche
Shift+	Ouvrir la boîte de dialogue Recherche avancée
Shift+	Basculer la barre de recherche
Esc	Vider la recherche courante
Shift+	Ciblez la liste des livres
Ctrl+E	Nettoyez la Bibliothèque virtuelle
Alt+Es	Nettoyez la restriction additionnelle
Ctrl+*	Créer une Bibliothèque virtuelle temporaire basée sur la recherche actuelle
Ctrl+D	Sélectionne le prochain volet de la Bibliothèque virtuelle
Ctrl+G	Sélectionne le volet précédent de la Bibliothèque virtuelle
N ou F3	Trouver le livre suivant qui correspond à la recherche actuelle (fonctionne uniquement si la la mise en évidence de la recherche est activée dans les préférences de recherche)
Shift+ ou Shift+	Trouver le livre précédent qui correspond à la recherche actuelle (fonctionne uniquement si la la mise en évidence de la recherche est activée dans les préférences de recherche)
Ctrl+D	Télécharger les métadonnées et les couvertures
Ctrl+R	Redémarrer calibre
Ctrl+S	Redémarrer calibre en mode dépannage
Shift+	Ajouter des livres à calibre
Ctrl+M	Basculer entre le statut marqués/démarqués sur les livres sélectionnés
Ctrl+/ ou Ctrl+A	Ouvrez la fenêtre popup pour rechercher le texte intégral de tous les livres de la bibliothèque
Q	Ouvre la fenêtre d'Affichage Rapide pour voir les livres en rapport aux séries/étiquettes/etc.
Shift+	Se concentre sur le panneau d'Affichage Rapide ouvert
Shift+	Effectue une recherche dans le panneau d'Affichage Rapide
F5	Réappliquer le tri actuel

suite sur la page suivante

Tableau 1 – suite de la page précédente

Raccourci clavier	Action
Ctrl+Q	Quitter calibre
X	Activer le défilement automatique de la liste des livres
Ctrl+A	Restreindre les livres affichés aux seuls livres appartenant à une catégorie actuellement affichée dans le *Navigateur d'étiquettes*.
B	Parcourir les annotations (mise en évidence et signets) faites dans la visionneuse calibre pour tous les livres de la bibliothèque
Ctrl+S	Parcourir les notes associées aux auteurs/séries/étiquettes/etc.
Alt+Sh	Basculer la disposition entre les vues larges et étroites

CHAPITRE 2

Ajouter votre site web favori d'actualités

calibre contient une structure de création puissante, flexible et facile à utiliser pour télécharger des actualités sur Internet et les convertir en livre numérique. Ce qui suit vous montrera, au moyen d'exemples, comment obtenir les actualités de divers sites web.

Pour avoir une compréhension de la façon d'employer cette structure de création, suivez les exemples ci-dessous dans l'ordre :

- *Récupération entièrement automatique* (page 34)
  - *Le blog calibre* (page 34)
  - *bbc.co.uk* (page 36)
- *Personnaliser le processus de récupération* (page 36)
  - *Utilisation de la version imprimable de bbc.co.uk* (page 36)
  - *Remplacer les styles d'article* (page 38)
  - *Découper et émincer* (page 38)
  - *Exemple de la vie réelle* (page 39)
- *Astuces pour développer de nouvelles recettes* (page 41)
- *Lectures recommandées* (page 42)
- *Documentation de l'API* (page 42)

2.1 Récupération entièrement automatique

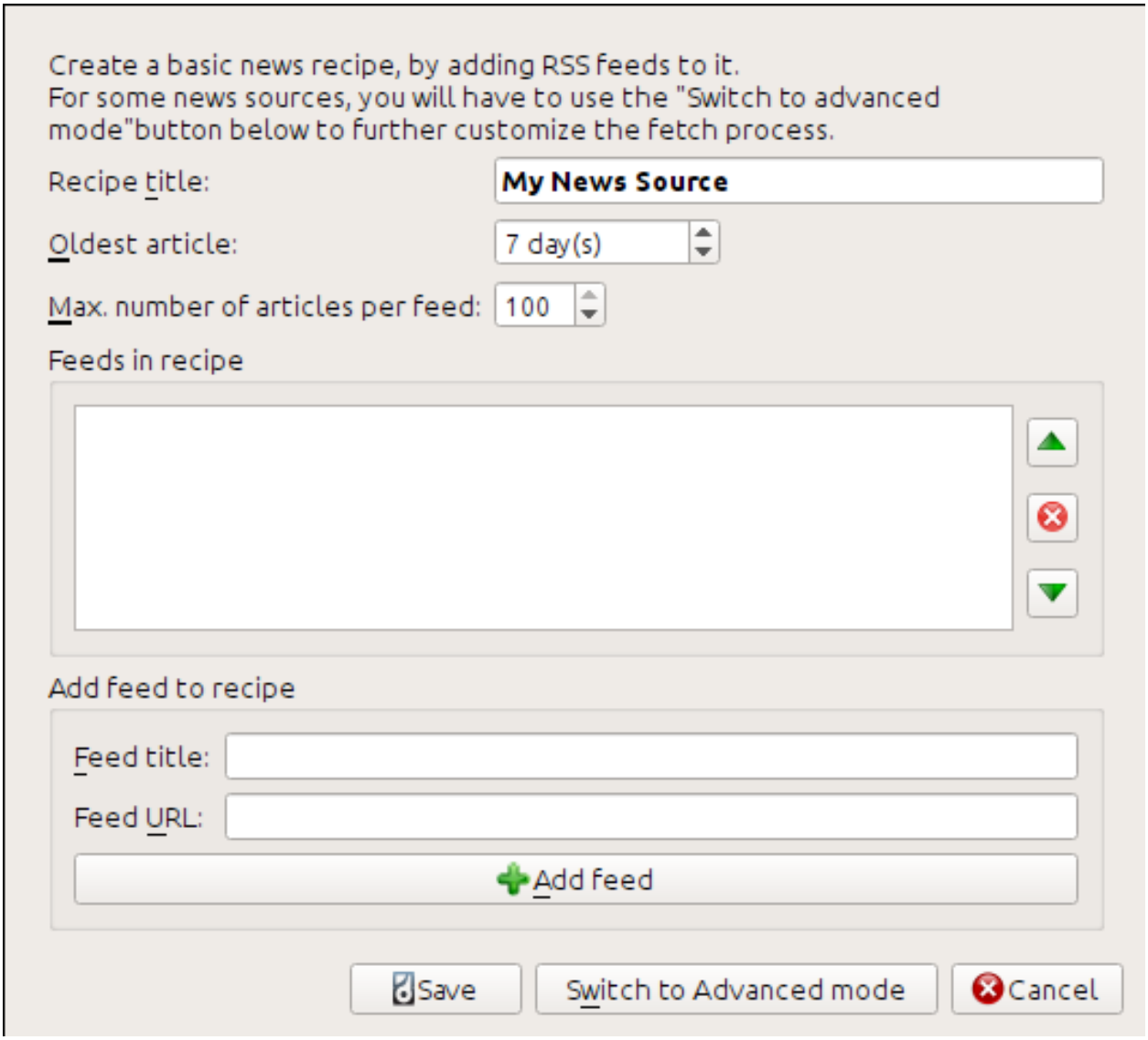
Si votre source d'actualité est assez simple, calibre est capable d'en effectuer la récupération de manière complètement automatique. Tout ce que vous avez à faire est de fournir l'URL. calibre recueille toutes les informations nécessaires pour télécharger une source d'actualités dans une *recette*. Pour intégrer à calibre une nouvelle source d'actualité, vous devez créer une *recette* pour celle-ci. Voyons quelques exemples :

2.1.1 Le blog calibre

Le blog calibre est un blog de postes qui décrivent beaucoup de fonctionnalités utiles de calibre d'une manière simple et accessible pour les nouveaux utilisateurs de calibre. Pour pouvoir télécharger ce blog dans un livre numérique, nous nous référons au flux *RSS* du blog :

```
http://blog.calibre-ebook.com/feeds/posts/default
```

J'ai obtenu l'URL RSS en regardant sous « Subscribe to » en bas de la page du blog et en choisissant *Posts → Atom*. Pour faire en sorte que calibre télécharge les flux et les convertissent en livre numérique, vous devrez faire une clic droit sur le bouton *Récupérer des actualités* et puis le bouton *Ajouter une source personnalisée d'informations*. Une boite de dialogue semblable à celle montrée ci-dessous devrait s'ouvrir.



Tout d'abord entrez `Blog calibre` dans le champ *Titre de la recette*. Ceci sera le titre du livre numérique qui sera créé à partir des articles des flux ci-dessus.

Les deux champs suivants (*Article le plus ancien* et *Max. Nombre d'articles par flux*) donnent la possibilité de contrôler combien d'articles pourront être téléchargés de chaque flux. Ils sont assez explicites.

Pour ajouter les flux à la recette, entrer le titre du flux et l'URL du flux et cliquez sur le bouton *Ajouter un flux*. Une fois que vous avez ajouté le flux, cliquez simplement sur le bouton *Sauvegarder* et vous avez fini ! Fermez la boîte de dialogue.

Pour tester votre nouvelle *recette*, cliquez le bouton *Récupérer des informations* et dans le sous menu *Personnaliser les sources d'actualités* cliquez guilabel :*Blog calibre*. Après quelques minutes, le nouveau livre numérique de postes de blog apparaîtra dans la vue de la bibliothèque principale (si votre lecteur est connecté, il sera déposé sur le lecteur plutôt que dans la bibliothèque). Sélectionnez-le et appuyer le bouton *Visualiser* pour le lire !

La raison pour laquelle cela fonctionne si bien, avec si peu d'efforts, est parce que le blog fournit un flux RSS à *contenu plein*, c'est à dire, dont le contenu de l'article est inclut dans le flux lui-même. Pour la plupart des sites d'actualités qui fournissent des actualités de cette façon, avec des flux à *contenu plein*, vous n'avez pas besoin de faire beaucoup plus d'efforts pour les convertir en livres numériques. Maintenant nous allons regarder à des sources d'actualités qui ne fournissent pas des flux à contenu plein. Dans de tels flux, L'article complet est une page web et le flux contient uniquement un lien vers la page web avec un petit résumé de l'article.

2.1.2 bbc.co.uk

Essayons les deux flux suivants de *The BBC* :

1. News Front Page : https://newsrss.bbc.co.uk/rss/newsonline_world_edition/front_page/rss.xml
2. Science/Nature : https://newsrss.bbc.co.uk/rss/newsonline_world_edition/science/nature/rss.xml

Suivez la procédure comme exposée dans *Le blog calibre* (page 34) pour créer une recette pour *The BBC* (en utilisant les flux ci-dessus). Regardez le livre numérique téléchargé,nous voyons que calibre a réalisé un travail honorable d'extraire seulement le contenu qui vous intéresse depuis la page web de chaque article. Cependant, le processus d'extraction n'est pas parfait. Parfois il laisse du contenu indésirable comme des menus et des aides à la navigation ou enlève du contenu qui aurait dû apparaître seul, comme des titres d'article. Pour obtenir une extraction parfaite, nous aurons besoin d'adapter le processus de récupération, tel que décrit dans la section suivante.

2.2 Personnaliser le processus de récupération

Quand vous voulez perfectionner le processus de téléchargement, ou télécharger du contenu d'un site web particulièrement complexe, vous pouvez vous servir de toutes la puissance et de la flexibilité de la structure *recette*. Afin de faire cela, dans la boîte de dialogue *Ajouter des sources d'actualités personnalisées*, cliquez simplement sur le bouton *Basculer vers le mode Avancé*.

La personnalisation la plus facile et souvent la plus productive est d'employer la version imprimable des articles en ligne. La version imprimable est habituellement moins compliqué et se traduit beaucoup plus souplement en un livre numérique. Essayons d'employer la version imprimable des articles de *The BBC*.

2.2.1 Utilisation de la version imprimable de bbc.co.uk

La première étape est de regarder le livre numérique que nous avons précédemment téléchargé depuis *bbc.co.uk* (page 36). A la fin de chaque article, dans le livre numérique il y a un petit texte de présentation vous indiquant d'où l'article a été téléchargé. Copier et coller cette URL dans un navigateur. Maintenant sur la page Web de l'article recherchez un lien qui pointe vers la « version imprimable ». Cliquez le pour voir la version imprimable de l'article. Il parait beaucoup plus ordonné ! Comparez maintenant les deux URLs. Pour moi elles étaient :

URL d'article
https://news.bbc.co.uk/2/hi/science/nature/7312016.stm

URL de version imprimable
https://newsvote.bbc.co.uk/mpapps/pagetools/print/news.bbc.co.uk/2/hi/science/nature/7312016.stm

Aussi il semble que pour obtenir la version imprimable, nous avons besoin de préfixer que URL d'article avec :

newsvote.bbc.co.uk/mpapps/pagetools/print/

Dorénavant dans le *Mode Avancé* de la boîte de dialogue Personnaliser les sources d'actualités, vous devriez voir quelque chose comme (rappeler vous de sélectionner la recette *The BBC* avant de basculer vers le mode avancé) :

Vous pouvez voir que les champs du *Mode de base* ont été traduits en code de Python d'une façon simple. Nous devons ajouter des instructions à cette recette pour employer la version imprimable des articles. Tout ce qui est nécessaire est d'ajouter les deux lignes suivantes :

```
def print_version(self, url):
    return url.replace('https://', 'https://newsvote.bbc.co.uk/mpapps/pagetools/print/
↪')
```

C'est du Python, donc l'indentation est importante. Après que vous ayez ajouté les lignes, elles devraient ressembler à :

Ci-dessus, `def print_version(self, url)` définit une *méthode* qui est appelée par calibre pour chaque article. `url` est l'URL de l'article original. Ce que `print_version` fait est de prendre cette url et de la remplacer avec l'URL qui pointe vers la version imprimable de l'article. Pour apprendre à propos de Python[4] voir le tutoriel[5].

Maintenant, cliquez sur le bouton *Ajouter/Mettre à jour la recette* et vos changements seront enregistrés. Téléchargez à nouveau le livre numérique. Vous devriez obtenir un livre numérique beaucoup plus amélioré. Un des problèmes avec la nouvelle version est que les polices sur la page Web de la version imprimable sont trop petites. Ceci est automatiquement fixé en convertissant en livre numérique, mais même après le processus de fixation, la taille de la police des menus et la barre de navigation peuvent être trop grandes relativement au texte de l'article. Pour fixer ceci, nous ferons encore plus de personnalisation dans la prochaine section.

[4] https://www.python.org
[5] https://docs.python.org/tutorial/

2.2.2 Remplacer les styles d'article

Dans la section précédente, nous avons vu que la taille de la police pour des articles de la version imprimable de *The BBC* était trop petite. Dans la plupart des sites Web, le *The BBC* inclus, cette taille de la police est placée au moyen de feuilles de style *CSS*. Nous pouvons désactiver la récupération de telles feuilles de style n ajoutant la ligne :

```
no_stylesheets = True
```

La recette ressemble maintenant à :

Recipe source code (python)

```
class AdvancedUserRecipe1206419520(BasicNewsRecipe):
    title          = u'The BBC'
    oldest_article = 7
    max_articles_per_feed = 100
    no_stylesheets = True

    feeds          = [(u'News Front Page', u'http://newsrss.bbc.co.uk/rss/newsonli

    def print_version(self, url):
        return url.replace('http://', 'http://newsvote.bbc.co.uk/mpapps/pagetools/
```

La nouvelle version parait assez bonne. Si vous êtes un perfectionniste, vous voudrez lire la prochaine section, qui traite de modifier réellement le contenu téléchargé.

2.2.3 Découper et émincer

calibre a des capacités très puissantes et flexibles quand il s'agit de manipuler le contenu téléchargé. Pour montrer quelques uns de ces derniers, regardons encore notre vieille amie la recette *The BBC* (page 38). Regardez le code source (*HTML*) de quelques articles (version imprimable), nous voyons qu'il contient un pied de page qui n'apporte aucune information utile

```
<div class="footer">
...
</div>
```

Ceci peut être supprimé en ajoutant :

```
remove_tags    = [dict(name='div', attrs={'class':'footer'})]
```

à la recette. Finalement, remplaçons certains des *CSS* que nous avons désactivés plus tôt, avec notre propre *CSS* qui est plus adapté pour la conversion en livre numérique :

```
extra_css      = '.headline {font-size: x-large;} \n .fact { padding-top: 10pt  }'
```

Avec ces ajouts, notre recette est devenue « qualité de production ».

Cette *recette* explore seulement la partie émergée de l'iceberg quand il s'agit de la puissance de calibre. Pour mieux explorer les capacités de calibre nous examinerons un exemple plus complexe de vie réelle dans la prochaine section.

2.2.4 Exemple de la vie réelle

Un exemple de la vie réelle raisonnablement complexe qui expose plus de l'*API* de `BasicNewsRecipe` est la *recette* pour *The New York Times*

```
import string, re
from calibre import strftime
from calibre.web.feeds.recipes import BasicNewsRecipe
from calibre.ebooks.BeautifulSoup import BeautifulSoup

class NYTimes(BasicNewsRecipe):

    title       = 'The New York Times'
    __author__  = 'Kovid Goyal'
    description = 'Daily news from the New York Times'
    timefmt = ' [%a, %d %b, %Y]'
    needs_subscription = True
    remove_tags_before = dict(id='article')
    remove_tags_after  = dict(id='article')
    remove_tags = [dict(attrs={'class':['articleTools', 'post-tools', 'side_tool',
↪'nextArticleLink clearfix']}),
                dict(id=['footer', 'toolsRight', 'articleInline', 'navigation',
↪'archive', 'side_search', 'blog_sidebar', 'side_tool', 'side_index']),
                dict(name=['script', 'noscript', 'style'])]
    encoding = 'cp1252'
    no_stylesheets = True
    extra_css = 'h1 {font: sans-serif large;}\n.byline {font:monospace;}'

    def get_browser(self):
        br = BasicNewsRecipe.get_browser(self)
        if self.username is not None and self.password is not None:
            br.open('https://www.nytimes.com/auth/login')
            br.select_form(name='login')
            br['USERID']   = self.username
            br['PASSWORD'] = self.password
            br.submit()
        return br

    def parse_index(self):
        soup = self.index_to_soup('https://www.nytimes.com/pages/todayspaper/index.
↪html')

        def feed_title(div):
            return ''.join(div.findAll(text=True, recursive=False)).strip()

        articles = {}
        key = None
        ans = []
        for div in soup.findAll(True,
             attrs={'class':['section-headline', 'story', 'story headline']}):

             if ''.join(div['class']) == 'section-headline':
                 key = string.capwords(feed_title(div))
                 articles[key] = []
                 ans.append(key)

             elif ''.join(div['class']) in ['story', 'story headline']:
                 a = div.find('a', href=True)
```

(suite sur la page suivante)

(suite de la page précédente)

```
                if not a:
                    continue
                url = re.sub(r'\?.*', '', a['href'])
                url += '?pagewanted=all'
                title = self.tag_to_string(a, use_alt=True).strip()
                description = ''
                pubdate = strftime('%a, %d %b')
                summary = div.find(True, attrs={'class':'summary'})
                if summary:
                    description = self.tag_to_string(summary, use_alt=False)

                feed = key if key is not None else 'Uncategorized'
                if feed not in articles:
                    articles[feed] = []
                if not 'podcasts' in url:
                    articles[feed].append(
                              dict(title=title, url=url, date=pubdate,
                                   description=description,
                                   content=''))
        ans = self.sort_index_by(ans, {'The Front Page':-1, 'Dining In, Dining Out':1,
↪ 'Obituaries':2})
        ans = [(key, articles[key]) for key in ans if key in articles]
        return ans

    def preprocess_html(self, soup):
        refresh = soup.find('meta', {'http-equiv':'refresh'})
        if refresh is None:
            return soup
        content = refresh.get('content').partition('=')[2]
        raw = self.browser.open('https://www.nytimes.com'+content).read()
        return BeautifulSoup(raw.decode('cp1252', 'replace'))
```

Nous voyons plusieurs nouvelles fonctionnalités dans cette *recipe*. En premier, nous avons :

```
timefmt = ' [%a, %d %b, %Y]'
```

Ceci règle le temps affiché sur la page de garde du livre numérique créé au format , `Day, Day_Number Month, Year`. Voir *timefmt* (page 51).

Voyons maintenant un groupe de directives pour nettoyer l" :term :` HTML` téléchargé :

```
remove_tags_before = dict(name='h1')
remove_tags_after  = dict(id='footer')
remove_tags = ...
```

Celle-ci supprime tout avant la première balise <h1> et tout ce qui se trouve après le première balise dont l'id est `footer`. Voir *remove_tags* (page 50), *remove_tags_before* (page 51), *remove_tags_after* (page 51).

La prochaine fonctionnalité intéressante est :

```
needs_subscription = True
...
def get_browser(self):
    ...
```

`needs_subscription = True` dit à calibre que cette recette nécessite un nom d'utilisateur et un mot de passe pour accéder au contenu. Ceci amène calibre à demander après un nom d'utilisateur et un mot de passe à chaque

fois que vous essayez d'utiliser cette recette. Le code dans *calibre.web.feeds.news.BasicNewsRecipe.get_browser()* (page 43) fait réellement l'ouverture de session sur le site Web de NYT. Une fois connecté, calibre utilisera la même instance, connectée, du navigateur pour récupérer tout le contenu. Voir mechanize[6] pour comprendre le code dans `get_browser`.

Une autre nouvelle fonctionnalité est la méthode *calibre.web.feeds.news.BasicNewsRecipe.parse_index()* (page 45). Sont travail est d'aller sur https://www.nytimes.com/pages/todayspaper/index.html et de récupérer la liste des articles qui apparaissent dans le journal du *jour*. Beaucoup plus complexe que d'utiliser simplement le *RSS*, la recette crée un livre numérique qui correspond pratiquement au journal du jour. `parse_index` fait une forte utilisation de BeautifulSoup[7] pour examiner la page web quotidienne. Vous pouvez aussi en utiliser d'autres, de plus modernes analyseurs si vous n'aimez pas BeautifulSoup. calibre est fourni avec lxml[8] et html5lib[9], qui sont les analyseurs recommandés. Pour les utiliser, remplacer l'appel à `index_to_soup()` avec le suivant :

```
raw = self.index_to_soup(url, raw=True)
# For html5lib
import html5lib
root = html5lib.parse(raw, namespaceHTMLElements=False, treebuilder='lxml')
# For the lxml html 4 parser
from lxml import html
root = html.fromstring(raw)
```

La nouvelle fonctionnalité finale est la méthode *calibre.web.feeds.news.BasicNewsRecipe.preprocess_html()* (page 46). Elle peut être employée pour exécuter des transformations quelconques sur chaque page HTML téléchargée. Ici elle est employée pour éviter les publicités que le nytimes place avant chaque article.

2.3 Astuces pour développer de nouvelles recettes

La meilleure manière de développer de nouvelles recettes est d'utiliser l'interface de commande en ligne. Créer la recette en utilisant votre éditeur Python favori et sauvegardez la dans un fichier nommé `myrecipe.recipe`. L'extension *.recipe* est obligatoire. Vous pouvez télécharger le contenu qu'utilise cette recette avec la commande :

```
ebook-convert myrecipe.recipe .epub --test -vv --debug-pipeline debug
```

La commande **ebook-convert** téléchargera toutes les pages web et les enregistrera dans le fichier EPUB `myrecipe.epub`. L'option `-vv` fait en sorte que ebook-convert renvoie beaucoup d'informations sur ce qu'il fait. L'option `ebook-convert-recipe-input --test` fait qu'il ne télécharge que quelques articles d'au maximum deux flux. En outre, ebook-convert déposera l'HTML téléchargé dans le répertoire `debug/input`, où `debug` est le répertoire que vous avez spécifié dans l'option *ebook-convert --debug-pipeline* (page 348).

Une fois le téléchargement terminé, vous pouvez regarder l'*HTML* téléchargé en ouvrant le fichier `debug/input/index.html` dans un navigateur. Une fois que vous êtes satisfait que le téléchargement et le pré-traitement se sont déroulés correctement, vous pouvez générer des livres numériques dans différents formats comme montré ci-dessous :

```
ebook-convert myrecipe.recipe myrecipe.epub
ebook-convert myrecipe.recipe myrecipe.mobi
...
```

Si vous êtes satisfait de votre recette et que vous sentez qu'il y a suffisamment de demandes pour l'inclure dans le jeu de recettes intégrées, déposez votre recette sur le forum calibre de recettes[10] pour la partager avec les autres utilisateurs de calibre.

[6] https://mechanize.readthedocs.io/en/latest/
[7] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[8] https://lxml.de/
[9] https://github.com/html5lib/html5lib-python
[10] https://www.mobileread.com/forums/forumdisplay.php?f=228

Note : Sous macOS, les outils en de ligne de commande sont à l'intérieur de la suite logicielle calibre, par exemple, si vous installez calibre dans `/Applications` les outils de ligne de commande sont dans `/Applications/calibre.app/Contents/MacOS/`.

Voir aussi :

ebook-convert **(page 339)**
L'interface en ligne de commande pour toute conversion de livre numérique

2.4 Lectures recommandées

Pour en apprendre plus sur l'écriture avancée de recettes utilisant quelques unes des facilités disponibles dans `BasicNewsRecipe`, vous devriez consulter les sources suivantes :

Documentation API **(page 42)**
Documentation sur la classe `BasicNewsRecipe` et toutes ses méthodes importantes et champs.

BasicNewsRecipe[11]
Le code source de `BasicNewsRecipe`

Recettes intégrées[12]
Le code source des recettes intégrées qui est fourni avec calibre

Le forum de recettes calibre[13]
Un bon nombre d'auteurs bien informés des recettes calibre traînent ici.

2.5 Documentation de l'API

2.5.1 Documentation API pour les recettes

L'API pour l'écriture des recettes est définie par *BasicNewsRecipe* (page 42)

class calibre.web.feeds.news.**BasicNewsRecipe**(*options*, *log*, *progress_reporter*)

Classe de base qui contient toute la logique nécessaire dans toutes les recettes. En comprenant progressivement plus des fonctionnalités de cette classe, vous pouvez progressivement faire des recettes plus personnalisées/plus puissantes. Pour un tutoriel d'introduction sur la création des recettes, voir *Ajouter votre site web favori d'actualités* (page 33).

abort_article(*msg=None*)

Appelle cette méthode à l'intérieur de l'une des méthodes pré-traitée pour avorter le téléchargement de l'article actuel. Utile pour sauter les articles qui contiennent du contenu inadéquat, tel que les articles vidéos purs.

abort_recipe_processing(*msg*)

Entraîne le système de téléchargement de recette à abandonner le téléchargement de cette recette, affichant un simple message d'explication à l'utilisateur.

[11] https://github.com/kovidgoyal/calibre/blob/master/src/calibre/web/feeds/news.py
[12] https://github.com/kovidgoyal/calibre/tree/master/recipes
[13] https://www.mobileread.com/forums/forumdisplay.php?f=228

add_toc_thumbnail (*article*, *src*)

Appelez ceci à partir de populate_article_metadata avec l'attribut src d'une étiquette <img> de l'article qu'il est approprié d'utiliser comme la vignette représentant l'article dans la Table des Matières. Si la vignette qui est actuellement utilisée est dépendante du périphérique (actuellement uniquement utilisé par les Kindle). Notez que l'image référencée doit être une image qui a été téléchargée avec succès, autrement elle sera ignorée.

classmethod adeify_images (*soup*)

Si votre recette quand elle est convertie en EPUB a des problèmes avec les images quand il est vu dans Adobe Digital Editions, appelez cette méthode depuis *postprocess_html()* (page 46).

canonicalize_internal_url (*url*, *is_link=True*)

Renvoie un ensemble de représentations classiques de `url`. L'implémentation par défaut utilise juste le hostname du serveur et le chemin de l'URL, ignorant tout paramètres de requête, fragments, etc. Les représentations classiques doivent être uniques tout au long des URLs pour cette source d'actualités. Si elles ne le sont pas, alors les liens internes peuvent être incorrectement résolus.

Paramètres

is_link – Est True si l'URL provient d'un lien interne dans un fichier HTML. False si l'URL est l'URL utilisée pour télécharger un article.

cleanup ()

Appelé après que tous les articles aient été téléchargés. Utilisez-le pour faire n'importe quel nettoyage comme se déconnecter des sites d'abonnement, etc.

clone_browser (*br*)

Clonez le navigateur. Les navigateurs clonés sont utilisés pour les téléchargements en parallèle, puisque mechanize n'est pas un thread sécurisé. Les routines par défaut de clonage devraient capturer la plupart des personnalisations de navigateur, mais si vous faites quelque chose d'exotique dans votre recette, vous devriez outrepasser cette méthode dans votre recette et cloner manuellement.

Les instances navigateurs clonées utilisent le même, thread sécurisé CookieJar par défaut, à moins que vous n'ayez adapté la manipulation des cookies.

default_cover (*cover_file*)

Créer une couverture générique pour les recettes qui n'ont pas de couverture

download ()

Téléchargez et prétraitez tous les articles des flux dans cette recette. Cette méthode devrait être appelée uniquement une fois sur une instance particulière de recette. L'appeler plus d'une fois mènera à un comportement non défini. :return : Path to index.html

extract_readable_article (*html*, *url*)

Extrait le contenu de l'article principal à partir de "html", nettoie et retourne comme un tuple (article_html, extracted_title). Basé sur l'algorithme original de lisibilité par Arc90.

get_article_url (*article*)

Contourne dans une sous-classe pour personnaliser l'extraction de l'*URL* qui pointe vers le contenu de chaque article. Retourne l'URL de l'article. Il est appelé avec *article*, un objet représentant un article analysé depuis un plux. Voir feedparser[14]. Par défaut il regarde après le lien original (pour les flux syndiqués par l'intermédiaire d'un service comme PeedBurner ou Pheedo) et si trouvé, retourne cela ou renvoie à article.link[15].

get_browser (**args*, ***kwargs*)

Retourne une instance de navigateur utilisée pour récupérer des documents à partir du web. Par défaut, retourne une instance de navigateur mechanize[16] qui supporte les cookies, ignore robots.txt, traite les rafraîchissements et a un agent utilisateur mozilla firefox.

Si votre recette requiert que vous vous connectiez au préalable, outrepasser cette méthode dans votre sous-classe. Par exemple, le code suivant est utilisé dans la recette New York Times pour se connecter avec un accès complet :

```
def get_browser(self):
    br = BasicNewsRecipe.get_browser(self)
    if self.username is not None and self.password is not None:
        br.open('https://www.nytimes.com/auth/login')
        br.select_form(name='login')
        br['USERID']   = self.username
        br['PASSWORD'] = self.password
        br.submit()
    return br
```

get_cover_url()

Renvoie une *URL* à l'image de couverture pour cette question ou *None*. Par défaut il retourne la valeur du membre *self.cover_url* qui est normalement *None*. Si vous voulez que votre recette télécharge une couverture pour le livre numérique outrepassez cette méthode dans votre sous-classe, ou paramétrez la variable membre `self.cover_url`avant que cette méthode ne soit appelée.

get_extra_css()

Par défaut renvoie *self.extra_css*. Outrepassez si vous voulez générer par programme extra_css

get_feeds()

Retourne une liste de flux *RSS* à récupérer pour ce profil. Chaque élément de la liste doit être un tuple de 2 éléments de la forme (title,url). Si title est None ou une chaîne vide, le titre du flux est utilisé. Cette méthode est utile si votre recette doit faire quelques traitements pour déterminer la liste des flux à télécharger. Si oui, outrepassez dans votre sous-classe.

get_masthead_title()

Outrepasser dans la sous-classe pour utiliser quelque chose d'autre que le titre de la recette

get_masthead_url()

Retourne une *URL* à l'image masthead pour cette question ou *None*. Par défaut il retourne la valeur du membre *selfl.masthead_url* qui est normalement *None*. Si vous voulez que votre recette télécharge une masthead pour le livre numérique outrepasser cette méthode dans votre sous-classe, ou paramétrer la variable membre `self.masthead_url`avant que cette méthode soit appelée. Les images Masthead sont utilisées dans les fichiers MOBI de Kindle.

get_obfuscated_article(*url*)

Si vous avez défini *articles_are_obfuscated*, cette méthode est appelée avec chaque URL d'article. Elle doit retourner le chemin à un fichier sur le système de fichier qui contient l'article HTML. Ce ficher est traité par le moteur de récupération récursif d'HTML, aussi il peut contenir des liens vers des pages/images sur le web. Vous pouvez également renvoyer un dictionnaire de la forme suivante : {"data" : <HTML data>, "url" : <the resolved URL of the article>}. Cela évite de devoir créer des fichiers temporaires. La clé *url* du dictionnaire est utile si l'URL effective de l'article est différente de l'URL passée dans cette méthode, par exemple, à cause de redirections. Elle peut être omise si l'URL reste inchangée.

Cette méthode est typiquement utile pour les sites qui essayent de rendre difficile l'accès automatique au contenu d'article.

get_url_specific_delay(*url*)

Renvoie le délai en secondes avant le téléchargement de cette URL. Si vous souhaitez déterminer par programme le délai pour l'URL spécifié, surchargez cette méthode dans votre sous-classe, en renvoyant self.delay par défaut pour les URL que vous ne souhaitez pas affecter.

Renvoie

Un nombre à virgule flottante, le délai en secondes.

classmethod image_url_processor(*baseurl*, *url*)

Effectue un certain traitement sur les URLs d'images (peut-être en supprimant les restrictions de taille pour les images générées dynamiquement, etc.) et renvoie l'URL prétraitée. Retourne None ou une chaîne vide pour ne pas récupérer l'image.

index_to_soup(*url_or_raw*, *raw=False*, *as_tree=False*, *save_raw=None*)

Méthode commode qui prend une URL vers la page index et retourne une BeautifulSoup[17] de celle-ci.

url_or_raw : Soit une URL soit la page d'index téléchargée comme une chaîne

is_link_wanted(*url*, *tag*)

Renvoie True si le lien devrait être suivi ou autrement False. Par défaut, augmente NotImplementedError qui amène le téléchargement à l'ignorer.

Paramètres

- **url** – L'URL à suivre
- **tag** – La balise depuis laquelle l'URL a été dérivée

parse_feeds()

Crée une liste d'articles depuis la liste des flux renvoyés par *BasicNewsRecipe.get_feeds()* (page 44). Renvoie une liste d'objets `Feed`.

parse_index()

Cette méthode devrait être appliquée dans les recettes qui analysent un site Web au lieu des flix pour produire une liste d'articles. Les utilisations typiques sont pour les sources d'actualités qui ont une page Web « édition imprimable » qui énumère tous les articles dans l'actuelle édition imprimable. Si cette fonction est mise en application, elle sera employée de préférence à meth :*BasicNewsRecipe.parse_feeds*.

Il doit renvoyer une liste. Chaque élément de la liste doit être un tuple de 2 éléments de la forme `('feed title', list of articles)`.

Chaque liste d'articles doit contenir des dictionnaires de la forme :

```
{
'title'       : article title,
'url'         : URL of print version,
'date'        : The publication date of the article as a string,
'description' : A summary of the article
'content'     : The full article (can be an empty string). Obsolete
                do not use, instead save the content to a temporary
                file and pass a file:///path/to/temp/file.html as
                the URL.
}
```

Pour un exemple, regardez la recette pour télécharger *The Atlantic*. En plus, vous pouvez ajouter *author* pour l'auteur de l'article.

Si vous voulez arrêtez le traitement pour quelque raison et que vous avez calibre qui affiche à l'utilisateur un simple message plutôt qu'une erreur appelez *abort_recipe_processing()* (page 42).

populate_article_metadata(*article*, *soup*, *first*)

Appelé quand chaque page HTML appartenant à l'article est téléchargée. Conçu afin d'obtenir les métadonnées de l'article comme auteur/résumé/etc. à partir du HTML analysé (soup).

Paramètres

- **article** – Un objet de la classe `calibre.web.feeds.Article`. Si vous changez le résumé, n'oubliez pas de changer également le text_sumary

- **soup** – HTML analysé appartenant à cet article
- **first** – Vrai si et seulement si le HTML filtré est la première page de l'article.

postprocess_book (*oeb*, *opts*, *log*)

Exécutez n'importe quel post-traitement nécessaire sur le livre numérique téléchargé et analysé.

Paramètres

- **oeb** – Un objet OEBBook
- **opts** – Options de conversion

postprocess_html (*soup*, *first_fetch*)

Cette méthode est appelée avec la source de chaque fichier *HTML* téléchargé, après qu'il soit analysé pour des liens et des images. Il peut être employé pour faire arbitrairement un puissant post traitement sur l'*HTML*. Il devrait renvoyer *soup* après l'avoir traité.

Paramètres

- **soup** – Une instance BeautifulSoup[18] contenant l'*HTML* téléchargé.
- **first_fetch** – Vrai si c'est la première page d'un article.

preprocess_html (*soup*)

Cette méthode est appelée avec la source de chaque fichier *HTML* téléchargé, avant qu'il soit analysé pour des liens et des images. Il est appelé après le nettoyage comme spécifié par remove_tags etc. Il peut être employé pour faire arbitrairement un puissant post traitement sur l'*HTML*. Il devrait renvoyer *soup* après l'avoir traité.

soup : Une instance BeautifulSoup[19] contenant l'*HTML* téléchargé.

preprocess_image (*img_data*, *image_url*)

Exécute certains traitements sur les données d'une image téléchargée. Ceci est appliqué sur les données brutes avant que le redimensionnement soit terminé. Dois restituer les données brutes traitées. Restitue Aucun pour sauter l'image.

preprocess_raw_html (*raw_html*, *url*)

Cette méthode est appelée avec la source de chaque fichier *HTML* téléchargé, avant d'être analysé dans un arbre d'objet. raw_html est une chaîne unicode représentant l'HTML brut téléchargé du web. url est l'URL à partir de laquelle l'HTML a été téléchargé.

Notez que cette méthode agit *avant* preprocess_regexps.

La méthode doit retourner le rw_html traité comme un objet unicode.

classmethod print_version (*url*)

Prendre une *url* pointant vers une page web avec du contenu d'article et renvoie l'*URL* pointant vers la version imprimable de l'article. Par défaut ne fait rien. Par exemple :

```
def print_version(self, url):
    return url + '?&pagewanted=print'
```

publication_date ()

Utiliser cette méthode pour définir la date à laquelle ce numéro a été publié. La valeur par défaut est le moment du téléchargement. Doit retourner un objet `datetime.datetime`.

skip_ad_pages (*soup*)

Cette méthode est appelée avec la source de chaque fichier *HTML* téléchargé, avant tout les attributs de nettoyage comme remove_tags, keep_only_tags soient appliqués. Notez que preprocess_regexps aura déjà été appliqué. Il est censé permettre à la recette d'ignorer les pages de pub. Si la soup représente une page de pub, renvoie l'HTML de la page réelle. Autrement renvoie None.

soup : Une instance BeautifulSoup[20] contenant l'*HTML* téléchargé.

sort_index_by (*index*, *weights*)

Méthode commode pour trier les titres dans *index* selon *weights*. *index est trié sur place. Retourne `index*.

index : Une liste de titres.

weights : Un dictionnaire qui mappe les poids aux titres. Si quelques titres dans index n'ont pas de poids, ils sont supposés avoir un poids de 0.

classmethod tag_to_string (*tag*, *use_alt=True*, *normalize_whitespace=True*)

Méthode commode pour prendre une `Balise` BeautifulSoup[21] et extraire le texte de celle-ci récursivement, incluant n'importe quelles sections CDATA et les attributs alt tag. Renvoie une chaîne Unicode éventuellement vide.

usr_°alt : Si `True`essaye d'utiliser l'attribut alt pour les balises qui n'ont pas de contenu textuel.

tag : BeautifulSoup[22] `Tag`

articles_are_obfuscated = False

Paramétrez à True et implémentez *get_obfuscated_article()* (page 44) pour manipuler les sites Web qui essayent de le rendre difficile de ratisser le contenu.

auto_cleanup = False

Extrait automatiquement tout le texte des pages d'articles téléchargés. Utilise les algorithmes du projet lisibilité. Paramétrer ceci à vrai, signifie que vous n'avez pas à vous inquiéter à propos du nettoyage manuel de l'HTML téléchargé (cependant un nettoyage manuel sera toujours supérieur).

auto_cleanup_keep = None

Spécifier les éléments que l'algorithme de nettoyage automatique ne devrait jamais supprimer. La syntaxe est une expression XPath. Par exemple :

```
auto_cleanup_keep = '//div[@id="article-image"]' will keep all divs with
                                                 id="article-image"
auto_cleanup_keep = '//*[@class="important"]' will keep all elements
                                              with class="important"
auto_cleanup_keep = '//div[@id="article-image"]|//span[@class="important"]'
                  will keep all divs with id="article-image" and spans
                  with class="important"
```

center_navbar = True

Si True la barre de navigation est alignée au centre, autrement elle est alignée à gauche.

compress_news_images = False

Paramétrer ceci à False pour ignorer tous les paramètres de dimensionnement et de compression et pour analyser des images non modifiées. Si True et les autres paramètres de compression sont laissés à leurs valeurs par défaut, des images JPEG seront dimensionnées pour s'adapter aux dimensions d'écran réglées par le profil de sortie et comprimées pour mesurer tout au plus (w * h)/16 où w x h sont les dimensions mesurées d'image.

compress_news_images_auto_size = 16

Le facteur utilisé lors de la compression automatique des images JPEG. Si réglé à None, la compression automatique est désactivée. Autrement, les images seront réduites de taille à (w * h)/compress_news_images_auto_size bytez si possible en réduisant le niveau de qualité, où w x h sont les dimensions de l'image en pixels. La qualité minimale JPEG sera de 5/100 aussi est-il possible que la contrainte ne sera pas rencontrée.Ce paramètre peut être outrepassé par le paramètre compress_news_images_max_size qui procure une taille fixe pour les images. Notez que si vous activez scale_news_images_to_device alors l'image sera d'abord mise à l'échelle et ensuite sa qualité diminuée jusqu'à ce que sa taille soit moins que (w * h)/factor où w et h sont maintenant les dimensions de l'image *mise à l'échelle*,

compress_news_images_max_size = None

Paramétrer la qualité JPEG afin que les images n'excèdent pas la taille donnée (en KBytes). Si réglé, ce paramètre outrepasse la compression automatique par l'intermédiaire de compress_news_images_auto_size. La qualité minimale JPEG sera de 5/100 ainsi il est possible que cette contrainte ne sera pas rencontrée.

conversion_options = {}

Options spécifiques de recette pour contrôler la conversion du contenu téléchargé dans un livre numérque. Ceci outrepassera n'importe quelle valeur spécifiée par l'utilisateur ou une extension, aussi ne l'utilisr que si absolument nécessaire. Par exemple :

```
conversion_options = {
  'base_font_size'   : 16,
  'linearize_tables' : True,
}
```

cover_margins = (0, 0, '#ffffff')

Par défaut, l'image de couverture renvoyée par get_cover_url() sera utilisée comme couverture du périodique. Remplacer ceci dans votre recette donne l'instruction à calibre de rendre la couverture téléchargée dans un cadre dont la largeur et la hauteur sont exprimées en pourcentage de la couverture téléchargée. cover_margins = (10, 15, "#ffffff") rembourre la couverture avec une marge blanche de 10px à gauche et à droite, 15px en haut et en bas. Les noms des couleurs sont définis ici[23]. Notez que pour une raison quelconque, le blanc ne fonctionne pas toujours sous Windows. Utilisez #ffffff à la place

delay = 0

Délai par défaut entre deux téléchargements consécutifs, en secondes. L'argument peut être un nombre à virgule flottante pour indiquer un temps plus précis. Voir *get_url_specific_delay()* (page 44) pour implémenter des délais par URL.

description = ''

Quelques lignes qui décrivent le contenu téléchargé par cette recette. Ceci sera employé principalement dans un GUI qui présente une liste de recettes.

encoding = None

Spécifiez un codage de priorité pour les sites qui ont des spécifications incorrectes de jeu de caractère . Le plus commun étant de spécifier `latin1` et d'utiliser `cp1252`. Si aucun, essai pour détecter theencoding. Si None, essaye de détecter l'encodage. S'il est appelable, L'appelable est appelé avec deux arguments : L'objet de recette et la source à décoder. Elle doit renvoyer la source décodée.

extra_css = None

Spécifiez n'importe quel *CSS* supplémentaire qui devrait être ajouté aux fichiers *HTML* téléchargés. Il sera inséré dans les balises *<style>* juste avant la balise de fermeture *</head>* outrepassant tous les *CSS* excepté ce qui est déclaré utilisant un attribut de style sur des balises *HTML* individuelles. Notez que si vous voulez générer par programme extra_css écrasera la méthode *get_extra_css()* (page 44) à la place. Par exemple :

```
extra_css = '.heading { font: serif x-large }'
```

feeds = None

Liste des flux à télécharger. Peut être soit `[url1, url2, ...]` ou `[('titre1', url1), ('titre2', url2),...]`

filter_regexps = []

Liste d'expressions régulières qui détermine quels liens à ignorer. Si vide, elle est ignorée. Utilisé seulement si is_link_wanted n'est pas implémenté. Par exemple :

```
filter_regexps = [r'ads\.doubleclick\.net']
```

supprimera toutes les URL qui contiennent *ads.doubleclick.net*.

Seulement une de *BasicNewsRecipe.match_regexps* (page 49) ou de *BasicNewsRecipe.filter_regexps* (page 48) devrait être définie.

handle_gzip = True

Mettre à False si vous ne voulez pas utiliser les transferts gzippés. Notez que certains vieux serveurs ne fonctionnent plus avec gzip.

ignore_duplicate_articles = None

Ignorer les articles en double qui sont présents dans plus d'une section. Un article en doublon est un article qui a le même titre et/ou URL. Pour ignorer les articles avec le même titre réglez ceci à :

```
ignore_duplicate_articles = {'title'}
```

Pour utiliser les URLs à la place réglez le à :

```
ignore_duplicate_articles = {'url'}
```

Pour correspondre au titre ou à l'URL, réglez à :

```
ignore_duplicate_articles = {'title', 'url'}
```

keep_only_tags = []

Garder uniquement les balises spécifiées et leurs descendants. Pour le format pour spécifier une balise voir *BasicNewsRecipe.remove_tags* (page 50). Si la liste n'est pas vide, alors la balise `<body>`sera vidée et sera remplie avec les balises qui correspondent aux entrées dans la liste. Parc exemple :

```
keep_only_tags = [dict(id=['content', 'heading'])]
```

gardera uniquement les balises qui ont un attribut *id* de *« content »* ou de *« heading »*.

language = 'und'

La langue dans laquelle les actualités sont. Doit être un code ISO-639 soit deux ou trois caractères de long

masthead_url = None

Par défaut, caparlibre emploiera une image défaut pour le masthead (Kindle seulement). Outrepassez ceci dans votre recette pour fournir une URL à utiliser comme masthead.

match_regexps = []

Liste d'expressions régulières qui détermine quels liens à suivre. Si vide, elle est ignorée. Utilisé seulement si is_link_wanted n'est pas implémenté. Par exemple :

```
match_regexps = [r'page=[0-9]+']
```

correspondra à toutes les URL qui contiennent *page=some number*.

Seulement une de *BasicNewsRecipe.match_regexps* (page 49) ou de *BasicNewsRecipe.filter_regexps* (page 48) devrait être définie.

max_articles_per_feed = 100

Nombre maximum d'articles à téléchargé de chaque flux. Ceci est principalement utile pour les flux qui n'ont pas de date d'article. Pour plus de flux, vous devriez utiliser *BasicNewsRecipe.oldest_article* (page 50)

needs_subscription = False

Si True le GUI demandera à l'utilisateur pour un nom d'utilisateur et un mot de passe à utiliser lors du téléchargement Si réglé à « optionnal » l'utilisation du nom d'utilisateur et du mot de passe devient optionnel

no_stylesheets = False

Indicateur commode pour désactiver le chargement des feuilles de style pour les sites Web qui ont des feuilles de style inappropriées excessivement complexes pour la conversion de formats de livres numériques Si True les feuiiles de style ne sont pas téléchargées et ne sont pas traitées

oldest_article = 7.0

L'article le plus ancien à télécharger de cette source d'actualités. En jours.

preprocess_regexps = []

Liste de règles de substitution *term :`regexp* à exécuter sur l'*HTML* téléchargé. Chaque élément de la liste devrait un tuple à deux éléments. Le premier élément du tuple devrait être une expression régulière compilée et la seconde et le second un appelable qui prend un seul objet qui correspond et retourne une chaîne qui remplace la correspondance. Par exemple :

```
preprocess_regexps = [
    (re.compile(r'<!--Article ends here-->.*</body>', re.DOTALL|re.IGNORECASE),
     lambda match: '</body>'),
]
```

supprimera tout de *<!–Article ends here–>* à *</body>*.

publication_type = 'unknown'

Le type de publication a placé au journal, au magazine ou au blog. Si réglé à None, aucune métadonnée de type publication ne sera écrite dans le fichier opf.

recipe_disabled = None

Régler à une chaîne non vide pour désactiver la recette. La chaîne sera utilisée comme message de désactivation

recursions = 0

Nombre de niveaux de liens à suivre sur des pages web d'article

remove_attributes = []

Liste des attributs à supprimer de toutes les balises Par exemple :

```
remove_attributes = ['style', 'font']
```

remove_empty_feeds = False

Si True les flux vides seront supprimés de la sortie. Cette option n'a pas d'effet si parse_index est outrepassée dans la sous classe. Cela s'indique seulement pour les recettes qui retournent une liste de flux utilisant *flux* ou *get_feeds()* (page 44). C'est aussi utilisé si vous employez l'option ignore_duplicate_article.

remove_javascript = True

Indicateur commode pour retirer toutes les balises Javascript de l'HTML téléchargé

remove_tags = []

Liste de balises qui doivent être supprimées. Les balises spécifiées sont supprimées de l'HTML téléchargé. Une balise est spécifiée comme un dictionnaire de la forme :

```
{
 name       : 'tag name',   #e.g. 'div'
 attrs      : a dictionary, #e.g. {'class': 'advertisment'}
}
```

Toutes les clés sont optionnelles. Pour une explication complète du critère de recherche, voir *Beautiful Soup <https ://www.crummy.com/software/BeautifulSoup/bs4/doc/#searching-the-tree>* __ Un exemple commun :

```
remove_tags = [dict(name='div', class_='advert')]
```

Ceci supprimera toutes les balises *<div class= »advert »>* et tous leurs descendants de l'*HTML* téléchargé.

remove_tags_after = None

Supprimer toutes les balises qui surviennent après la balise spécifiée. Pour le format pour spécifier une balise voir *BasicNewsRecipe.remove_tags* (page 50). Par exemple :

```
remove_tags_after = [dict(id='content')]
```

supprimera toutes les balises après le premier élément avec *id= »content »*.

remove_tags_before = None

Supprimer toutes les balises qui surviennent avant la balise spécifiée. Pour le format pour spécifier une balise voir *BasicNewsRecipe.remove_tags* (page 50). Par exemple :

```
remove_tags_before = dict(id='content')
```

supprimera toutes les balises avant le premier élément avec *id= »content »*.

requires_version = (0, 6, 0)

Version minimale de calibre pour utiliser cette recette

resolve_internal_links = False

Si réglé à True alors les liens dans les articles téléchargés qui pointent vers d'autres articles téléchargés sont changés pour pointer vers la copie téléchargée de l'article plutôt que son URL web originale. Si vous réglez à True, vous aurez aussi besoin d'implémenter *canonicalize_internal_url()* (page 43) pour travailler avec le schéma d'URL de votre site web particulier.

reverse_article_order = False

Renversez l'ordre des articles dans chaque flux

scale_news_images = None

Dimensions maximales (h,l) pour mettre les images à l'échelle. Si scale_news_images_to_device est True est réglé aux dimensions de l'écran du périphérique sauf s'il n'y a pas de profil paramétré, dans quel cas elle est laissé à n'importe quelle valeur qui lui a été assignée (par défaut None)

scale_news_images_to_device = True

Remet les images à l'échelle pour s'ajuster aux dimensions de l'écran du périphérique paramétré par le profil de sortie ? Ignoré si aucun profil de sortie n'a été paramétré.

simultaneous_downloads = 5

Nombre de téléchargements simultanés. Réglé à 1 si le serveur est difficile. Automatiquement réduit à 1 si *BasicNewsRecipe.delay* (page 48) > 0

summary_length = 500

Nombre de caractères maximum dans la description courte

template_css = '\n .article_date {\n color: gray; font-family: monospace;\n }\n\n .article_description {\n text-indent: 0pt;\n }\n\n a.article {\n font-weight: bold; text-align:left;\n }\n\n a.feed {\n font-weight: bold;\n }\n\n .calibre_navbar {\n font-family:monospace;\n }\n '

Le CSS qui est utilisé pour styliser les modèles, c-à-d les barres de navigation et les Tables des Matières. Plutôt qu'outrepasser cette variable, vous devriez utiliser *extra_css* dans votre recette pour personnaliser l'apparence.

timefmt = ' [%a, %d %b %Y]'

Le format de chaîne pour la date affichée sur la première page. Par défaut : Day_Name, Day_Number Month_Name Year

timeout = 120.0

Délai en secondes pour récupérer les fichiers depuis le serveur

title = 'Source d'Actualités Inconnue'

Le titre à utiliser pour le livre numérique

use_embedded_content = None

Normalement nous essayons de deviner si un flux a des articles complets inclus basé sur la longueur du contenu inclus. Si *None*, alors l'estimation par défaut est employée. Si *True* alors nous supposons toujours que les flux ont du contenu inclus et si *False* nous assumons toujours que le flux n'a pas de contenu inclus.

[14] https://pythonhosted.org/feedparser/
[15] https://pythonhosted.org/feedparser/reference-entry-link.html
[16] https://github.com/jjlee/mechanize
[17] https://www.crummy.com/software/BeautifulSoup/bs4/doc
[18] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[19] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[20] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[21] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[22] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[23] https://www.imagemagick.org/script/color.php

CHAPITRE 3

La visionneuse de livre numérique

calibre inclut une Visionneuse de livre numérique qui peut déchiffrer tous les principaux formats de livre numérique. La Visionneuse de livre numérique est hautement personnalisable et possède de nombreuses fonctionnalités avancées.

- *Démarrer la Visionneuse de livre numérique* (page 54)
- *Naviguer dans un livre numérique* (page 54)
- *Mise en évidence du texte* (page 55)
- *Lire à voix haute* (page 56)
- *Recherche dans le texte* (page 56)
- *Suivre les liens en utilisant uniquement le clavier* (page 57)
- *Personnalisation de l'aspect et de la convivialité de votre expérience de lecture* (page 57)
- *Consultation du dictionnaire* (page 57)
- *Copier du texte et des images* (page 57)
- *Zoom sur les images* (page 58)
- *Raccourcis clavier* (page 58)
- *Contenu non fluide* (page 60)
- *Concevoir votre livre pour qu'il fonctionne bien avec la visionneuse de calibre* (page 60)

3.1 Démarrer la Visionneuse de livre numérique

Vous pouvez consulter n'importe lequel des livres de votre bibliothèque calibre en sélectionnant le livre et en appuyant sur le bouton *Visualiser". Cela ouvrira le livre dans la visionneuse de livres numériques. Vous pouvez également lancer a visionneuse de livres numériques à partir du menu Démarrer de Windows. Sous macOS, vous pouvez l'épingler sur le dock et le lancer à partir de là. Sous Linux, vous pouvez utiliser son lanceur dans les menus du bureau ou exécuter la commande :command :`ebook-viewer*.

3.2 Naviguer dans un livre numérique

Vous pouvez « tourner les pages » d'un livre soit :

- En cliquant dans la marge gauche ou droite de la page avec la souris
- En appuyant sur les touches `Barre d'espace`, `page haut`, `page bas` ou les touches fléchées
- Sur un écran tactile en tapotant sur le texte ou en glissant à gauche et à droite

Vous pouvez accéder aux commandes de la visionneuse soit :

- En faisant un clic droit sur le texte
- En appuyant sur les touches `Echap` ou `Menu`
- Sur un écran tactile en tapotant sur le tiers supérieur de l'écran

La visionneuse a deux modes, « paginé » et « continu ». En mode paginé, le contenu du livre est présenté sous forme de pages, comme un livre papier. En mode continu, le texte est présenté en continu, comme dans un navigateur Web. Vous pouvez passer de l'un à l'autre en utilisant les *Préférences* de la visionneuse sous *Mise en page* ou en appuyant sur la touche `Ctrl+M`.

3.2.1 Signets

Lorsque vous êtes au milieu d'un livre et fermez la visionneuse, elle se souviendra où vous avez arrêté la lecture et y retournera la prochaine fois que vous ouvrirez le livre. Vous pouvez également définir des signets dans le livre en utilisant le bouton *Signet* ou en pressant `Ctrl+B` dans la visionneuse de livre numérique. Lors de l'affichage des livres au format EPUB, ces signets sont enregistrés dans le fichier EPUB lui-même. Vous pouvez ajouter des signets, puis envoyer le fichier à un ami. Quand il ouvrira le fichier, il sera en mesure de voir vos signets. Vous pouvez désactiver ce comportement dans la section *Divers* des préférences de la visionneuse.

3.2.2 Table des Matières

Si le livre que vous lisez a une Table des Matières définie, vous pouvez y accéder en appuyant sur le bouton *Table des Matières*. Cela fera apparaître une liste des sections du livre. Vous pouvez cliquer sur n'importe laquelle de celles-ci pour accéder à cette partie du livre.

3.2.3 Navigation par emplacement

Les livres numériques, contrairement aux livres papier, n'ont aucune notion des pages. Vous pouvez vous référer à des emplacements précis dans les livres numériques en utilisant la fonctionnalité *Aller à → Emplacement* dans les contrôles de visualisation.

Vous pouvez utiliser cette information d'emplacement pour désigner sans ambiguïté une partie du livre quand vous en discutez avec des amis ou pour y faire référence dans d'autres œuvres. Vous pouvez entrer ces emplacements sous *Aller à → Emplacement* dans les contrôles de la visionneuse.

Ici ce trouve une URL que vous pouvez copier vers le presse-papiers et coller dans d'autres programmes ou documents. En cliquant sur cette URL vous ouvrirez le livre dans la visionneuse de livre numérique de calibre à l'emplacement actuel.

Si vous cliquez sur les liens à l'intérieur du livre numérique, vous serez amenés vers différentes parties du livre, vers les notes de fin, par exemple. Vous pouvez utiliser le boutons *Précédent* et *Suivant* dans le coin supérieur gauche des contrôles de la visionneuse. Ces boutons se comportent exactement comme ceux d'un navigateur internet.

3.2.4 Mode référence

calibre possède également un *Mode référence* très pratique. Vous pouvez l'activer en cliquant sur le bouton *Mode référence* dans les commandes de la visionneuse. Une fois que vous avez fait cela, chaque paragraphe aura un numéro unique affiché au début, composé des numéros de section et de paragraphe.

Vous pouvez utiliser ce nombre pour vous référer sans ambiguïté à certaines parties des livres lorsque vous en discutez avec des amis ou que vous y faites référence dans d'autres ouvrages. Vous pouvez entrer ces nombres dans la fonction *Aller à* pour naviguer vers un endroit de référence particulier.

3.3 Mise en évidence du texte

Lorsque vous sélectionnez du texte dans la visionneuse, une petite barre contextuelle apparaît à côté de la sélection. Vous pouvez cliquer sur le bouton de mise en évidence dans cette barre pour créer une mise en évidence. Vous pouvez ajouter des notes et modifier la couleur de la surbrillance. Sur un écran tactile, appuyez longuement sur un mot pour le sélectionner et afficher la barre contextuelle. Une fois en mode surbrillance, vous pouvez modifier le texte sélectionné à l'aide de poignées de sélection adaptées aux écrans tactiles. Faites glisser les poignées vers les marges supérieures ou inférieures pour faire défiler le texte tout en le sélectionnant. Vous pouvez également `Maj+clic` ou `clic droit` pour étendre la sélection, ce qui est particulièrement utile pour les sélections sur plusieurs pages.

Vous pouvez utiliser le bouton *Mises en évidence* dans les commandes de la visionneuse pour afficher un panneau séparé avec une liste de toutes les mises en évidence du livre, triée par chapitre.

Vous pouvez parcourir *toutes les mises en évidence* de votre bibliothèque calibre en cliquant avec le bouton droit sur le bouton *Voir* et en choisissant *Parcourir les annotations*.

Enfin, si vous utilisez le Serveur de contenu calibre Content dans le visionneuse du navigateur, vous pouvez faire en sorte que le visionneuse synchronise ses annotations avec le visionneuse du navigateur en allant sur *Préférences → Divers* dans les préférences de la visionneuse et en entrant le nom d'utilisateur du Serveur de Contenu de la visionneuse avec lequel il doit se synchroniser. Utilisez la valeur spéciale « * » pour la synchronisation avec les utilisateurs anonymes.

3.4 Lire à voix haute

La visionneuse peut lire le texte du livre à voix haute. Pour l'utiliser, il suffit de cliquer sur le bouton *Lire à voix haute* dans les commandes de la visionneuse pour commencer à lire le texte du livre à voix haute. Le mot en cours de lecture est mis en évidence. La parole est synthétisée à partir du texte en utilisant les services de votre système d'exploitation pour le text-to-speech. Vous pouvez changer la voix utilisée en cliquant sur l'icône de l'engrenage dans la barre qui s'affiche lorsque *Lire à voix haute* est actif.

Vous pouvez également lire à haute voix les passages surlignés en ajoutant le bouton *Lire à voix haute* à la barre de sélection dans les préférences de la visionneuse sous *Comportement de sélection*.

Note : La prise en charge de la synthèse vocale dans les navigateurs est très incomplète et truffée de bogues, de sorte que la qualité de *Lire à voix haute* dans le navigateur dépend de la qualité de la prise en charge de la synthèse vocale dans le navigateur sous-jacent. En particulier, le surlignage du mot courant ne fonctionne pas, et le changement de vitesse ou de voix fera que la lecture recommencera depuis le début.

Note : Sous Linux, *Lire à voix haute* nécessite que Speech Dispatcher[24] soit installé et fonctionnel.

Note : Sous Windows, toutes les voix installées peuvent ne pas être visibles par le sous-système SAPI utilisé pour le text-to-speech. Il existe des instructions pour rendre toutes les voix visibles <https://www.mobileread.com/forums/showpost.php?p=4084051&postcount=108>`_.

3.5 Recherche dans le texte

La visionneuse dispose de capacités de recherche très puissantes. Appuyez sur la touche `Ctrl+F` ou accédez aux commandes de la visionneuse et cliquez sur rechercher. La forme la plus simple de recherche consiste à rechercher le texte que vous entrez dans la zone de texte. Les différentes formes de recherche sont choisies par la boîte de mode de recherche située sous l'entrée de recherche. Les modes disponibles sont les suivants :

1. *Contains* - Le mode par défaut le plus simple. Le texte saisi dans le champ de recherche est recherché partout. La ponctuation, les accents et les espaces sont ignorés. Par exemple, la recherche : `Pena` correspondra à tout ce qui suit : `pénal, pen a, pen.a et Peña`. Si vous sélectionnez la case *Sensible à la casse*, les accents, les espaces et la ponctuation ne sont plus ignorés.
2. *Mots entiers* - Recherche des mots entiers. Ainsi, par exemple, la recherche `pena` correspondra au mot `Peña` mais pas au mot `Pénal`. Comme pour les recherches *Contains* ci-dessus, les accents et la ponctuation sont ignorés, sauf si la case *Sensible à la casse* est cochée.
3. *Mots proches* - Recherche des mots entiers qui sont proches les uns des autres. Ainsi, par exemple, la recherche `calibre cool` correspondra aux endroits où les mots `calibre` et `cool` apparaissent à moins de soixante caractères l'un de l'autre. Pour modifier le nombre de caractères, ajoutez le nouveau nombre à la fin de la liste de mots. Par exemple, « calibre cool awesome 120 » correspondra aux endroits où les trois mots se trouvent à moins de 120 caractères l'un de l'autre. Notez que la ponctuation et les accents ne sont *pas* ignorés pour ces recherches.
4. *Regex* - Interprète le texte de recherche comme une *expression régulière*. Pour en savoir plus sur l'utilisation des expressions régulières, consultez le tutoriel.

[24] https://freebsoft.org/speechd

3.6 Suivre les liens en utilisant uniquement le clavier

La visionneuse de livres numériques dispose d'un *Mode Indices" qui vous permet de cliquer sur les liens dans le texte sans utiliser la souris. Appuyez sur la touche :kbd :`Alt+f* et tous les liens de l'écran en cours seront mis en évidence par un chiffre ou une lettre. Appuyez sur la lettre de votre clavier pour cliquer sur le lien.En appuyant sur la touche `Echap`, vous annulerez le *Mode indices* sans sélectionner aucun lien.

Si plus de trente-cinq liens sont affichés à l'écran, certains d'entre eux comporteront plusieurs lettres, auquel cas tapez la première et la deuxième, ou la première et appuyez sur `Entrer` pour les activer. Vous pouvez également utiliser la touche `Retour arrière` pour annuler une erreur de frappe.

3.7 Personnalisation de l'aspect et de la convivialité de votre expérience de lecture

Vous pouvez changer la taille des polices à la volée en utilisant *Taille de police* dans les contrôles de la visionneuse ou `Ctrl++` ou `Ctrl+-` ou en maintenant la touche :kbd :`Ctrl" et en utilisant la molette de la souris.

Les couleurs peuvent être changées dans la section :guilabel :`Couleurs" des préférences de la visionneuse.

Vous pouvez modifier le nombre de pages affichées à l'écran ainsi que les marges de page dans :guilabel :`Mise en page" dans les préférences de la visionneuse.

Vous pouvez afficher des en-têtes et pieds de page personnalisés tels que le temps restant à lire, le titre du chapitre en cours, la position dans le livre, etc. via la section *En-têtes et pieds de page* des préférences de la visionneuse.

Une personnalisation plus avancée peut être réalisée par les réglages *Styles*. Ici, vous pouvez spécifier une image d'arrière-plan à afficher sous le texte ainsi qu'une feuille de style que vous pouvez définir et qui sera appliquée à chaque livre. En l'utilisant, vous pouvez faire des choses comme changer les styles de paragraphe, la justification du texte, etc. Pour des exemples de feuilles de style personnalisées utilisées par les utilisateurs de calibre, voir les forums[25].

3.8 Consultation du dictionnaire

You can look up the meaning of words in the current book by double clicking or long tapping the word you want to lookup and then clicking the lookup button that looks like a library.

3.9 Copier du texte et des images

Vous pouvez sélectionner du texte et des images avec votre souris puis faire un clic droit et sélectionner *Copier* pour copier dans le presse-papiers. Le matériel copié peut être collé dans une autre application sous forme de texte et d'images.

[25] https://www.mobileread.com/forums/showthread.php?t=51500

3.10 Zoom sur les images

Vous pouvez zoomer pour afficher une image en taille réelle dans une fenêtre séparée en double-cliquant ou en appuyant longuement sur celle-ci. Vous pouvez également faire un clic droit sur l'image et choisir *Voir l'image*.

3.11 Raccourcis clavier

La visionneuse dispose de nombreux raccourcis clavier, comme le reste de calibre. Ils peuvent être personnalisés dans la visionneuse *Preferences*. Les raccourcis par défaut sont listés ci-dessous :

Tableau 1 : Raccourcis clavier pour la visionneuse de calibre

Raccourci clavier	Action
`Home`, `Ctrl+A`	Défiler jusqu'au début du fichier actuel dans un livre à plusieurs fichiers
`Ctrl+A`	
`Ctrl+H`	Faire défiler jusqu'au début du livre
`Ctrl+E`	Faire défiler jusqu'à la fin du livre
`End`, `Ctrl+A`	Défiler jusqu'a la fin du fichier actuel dans un livre à plusieurs fichiers
`Ctrl+A`	
`ArrowU`	Défilement vers l'arrière, en douceur en mode flux et par écran plein en mode paginé
`ArrowD`	Défilement vers l'avant, en douceur en mode flux et par écran plein en mode paginé
`ArrowL`	Défiler un peu vers la gauche en mode flux et d'une page en mode paginé
`ArrowR`	Défiler un peu vers la droite en mode flux et d'une page en mode paginé
`PageUp`	Faire défiler vers l'arrière par écrans pleins.
`Shift+`	
`PageDo`	Faire défiler vers l'avant par écrans pleins.
`Spaceb`	
`Ctrl+P`	Faire défiler vers la section précédente
`Ctrl+P`	Faire défiler vers la section suivante
`Alt+Ar`	Reculer
`Alt+Ar`	Avancer
`Ctrl+T`	Afficher/masquer la Table des Matières
`Ctrl+S`	Lire à voix haute
`Alt+P`	Modifiez rapidement les paramètres en créant et en utilisant les *Profiles*
`Alt+f`	Suivre les liens avec le clavier
`Ctrl+C`	Copier dans le presse-papier
`Alt+C`	Copier l'emplacement actuel dans le presse-papiers
`Ctrl+S`	Copier l'emplacement actuel comme URL calibre :// dans le presse-papiers
`/`, `Ctrl+f` `Cmd+f`	Lancer la recherche
`F3`, `Enter`	Trouver le suivant

suite sur la page suivante

Tableau 1 – suite de la page précédente

Raccourci clavier	Action
Shift+	Trouver le précédent
Shift+	
Ctrl+P	Augmenter la taille de la police
Meta+P	
Ctrl+M	Diminuer la taille de la police
Meta+M	
Ctrl+0	Restaurer la taille de police par défaut
Ctrl+]	Augmenter le nombre de pages par écran
Ctrl+[	Diminuer le nombre de pages par écran
Ctrl+A	Rendre automatique le nombre de pages par écran
F11, Ctrl+S	Passer en mode plein écran
Ctrl+M	Basculer entre le mode paginé et le mode continu pour la mise en page du texte
Ctrl+W	Basculer la barre de défilement
Ctrl+X	Activer le mode référence
Ctrl+B	Afficher/masquer les signets
Ctrl+A	Nouveau signet
Ctrl+N	Afficher les métadonnées du livre
Ctrl+E	
Ctrl+A	Recharger le livre
Ctrl+A	
Ctrl+S	Modifier la sélection actuelle d'un mot
Ctrl+S	Modifier la sélection actuelle d'un mot
Shift+	Modifier la sélection actuelle en avançant d'un caractère
Shift+	Modifier la sélection actuelle en reculant d'un caractère
Shift+	Modifier la sélection actuelle en avançant d'une ligne
Shift+	Étendre la sélection actuelle au début de la ligne
Shift+	Étendre la sélection actuelle à la fin de la ligne
Ctrl+A	Sélectionner tout
Shift+	Modifier la sélection actuelle en reculant d'une ligne
Ctrl+S	Modifier la sélection actuelle en avançant d'un paragraphe
Ctrl+S	Modifier la sélection actuelle en reculant d'un paragraphe
Esc, MenuKe	Afficher les contrôles de la Visionneuse de livre numérique
Ctrl+C	Afficher les préférences de la Visionneuse de livre numérique
Ctrl+E	
Meta+E	
Meta+C	
Ctrl+G ;, :	Aller à un emplacement ou une position spécifique du livre
Ctrl+S	Basculer vers le défilement automatique

suite sur la page suivante

Tableau 1 – suite de la page précédente

Raccourci clavier	Action
Alt+Ar	Défilement automatique plus rapide
Alt+Ar	Défilement automatique plus lent
Ctrl+I	Afficher/masquer l'Inspecteur
Ctrl+L	Afficher/masquer le panneau de recherche de mot
Ctrl+Q (Cmd+Q on macOS)	Quitter
Ctrl+P	Imprimer le livre au format PDF
Ctrl+F	Basculer la barre d'outils
Ctrl+H	Basculer le panneau des mises en évidence
Ctrl+D	Éditer ce livre

3.12 Contenu non fluide

Certains livres ont un contenu tellement grand qu'ils ne peuvent pas être divisés aux limites de la page. Par exemple tables ou balises `<pre>`. Dans ce cas, vous devez passer en mode *flow* en appuyant sur `Ctrl+M` pour lire ce contenu. Alternativement, vous pouvez également ajouter le CSS suivant à la section *Styles* des préférences de la visionneuse pour forcer le visionneuse à décomposer des lignes de texte en balises `<pre>`

```
code, pre { white-space: pre-wrap }
```

3.13 Concevoir votre livre pour qu'il fonctionne bien avec la visionneuse de calibre

La visionneuse calibre définira la classe « is-calibre-viewer » sur l'élément racine. Vous pouvez donc écrire des règles CSS qui ne s'appliquent qu'à cet élément. En outre, la visionneuse définira les classes suivantes sur l'élément `body` :

`body.calibre-viewer-dark-colors`
: Défini lors de l'utilisation d'un schéma de couleur sombre

`body.calibre-viewer-light-colors`
: Défini lors de l'utilisation d'un schéma de couleurs claires

`body.calibre-viewer-paginated`
: Défini en mode pagination

`body.calibre-viewer-scrolling`
: Défini en mode flux (non paginé)

`body.calibre-footnote-container`
: Défini lors de l'affichage d'un popup d'une note de bas de page

Pour finir, vous pouvez utiliser les couleurs du schéma de couleurs de calibre via les variables CSS[26]. La visionneuse calibre définit les variables suivantes : `--calibre-viewer-background-color`, `--calibre-viewer-foreground-color` et éventuellement `--calibre-viewer-link-color` dans les thèmes de couleurs qui définissent une couleur de lien.

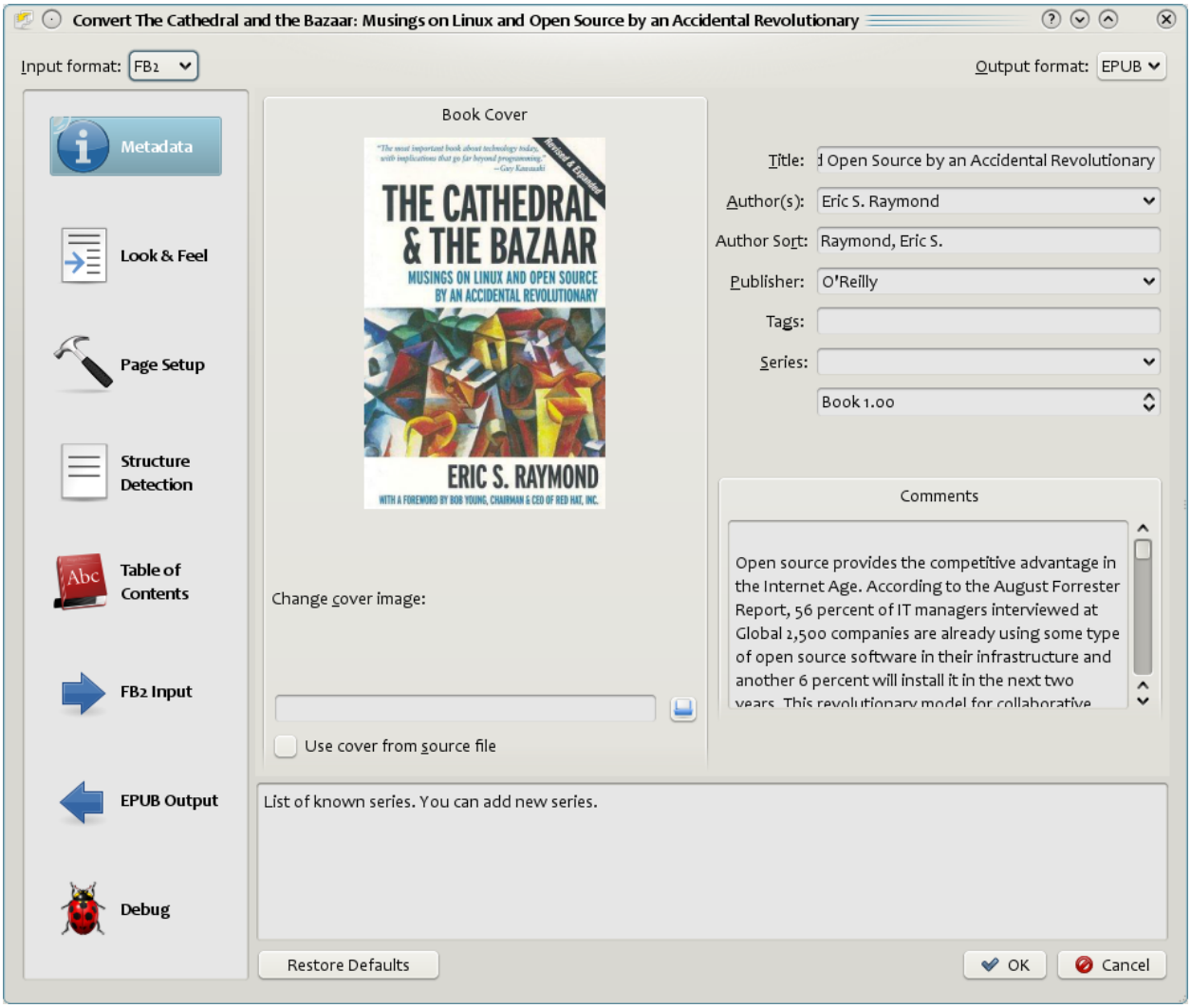
[26] https://developer.mozilla.org/fr/docs/Web/CSS/Using_CSS_custom_properties

CHAPITRE 4

Conversion de livre numérique

calibre a un système de conversion qui est prévu pour être très simple d'utilisation. Normalement, il suffit d'ajouter un livre à calibre, cliquer sur « convertir » et calibre s'efforcera de générer un résultat aussi proche que possible de l'original. calibre accepte un très large panel de formats, certains étant plus appropriés que d'autres à la conversion en livre numérique. Dans le cas de ces formats, ou si vous souhaitez simplement avoir plus de contrôle sur le système de conversion, calibre a beaucoup d'options pour affiner le processus de conversion. Cependant, il faut noter que le système de conversion de calibre ne peut pas remplacer un programme d'édition de livre numérique. Pour éditer des livres numériques, je recommande de les convertir d'abord en EPUB ou AZW3 en utilisant calibre et ensuite, utiliser la fonction *Éditer le livre* pour les perfectionner. Vous pouvez alors utiliser le livre numérique édité comme base pour la conversion vers d'autres formats avec calibre.

Ce document fera principalement référence aux paramètres de conversion comme trouvé dans la boîte de dialogue de conversion, représentée ci-dessous. Tous ces paramètres sont aussi disponible par l'interface en invite de commande pour la conversion, documenté dans *ebook-convert* (page 339). Dans calibre, vous pouvez obtenir de l'aide sur n'importe quel paramètre individuel en passant par-dessus avec votre souris, une info-bulle décrivant le paramètre apparaîtra.

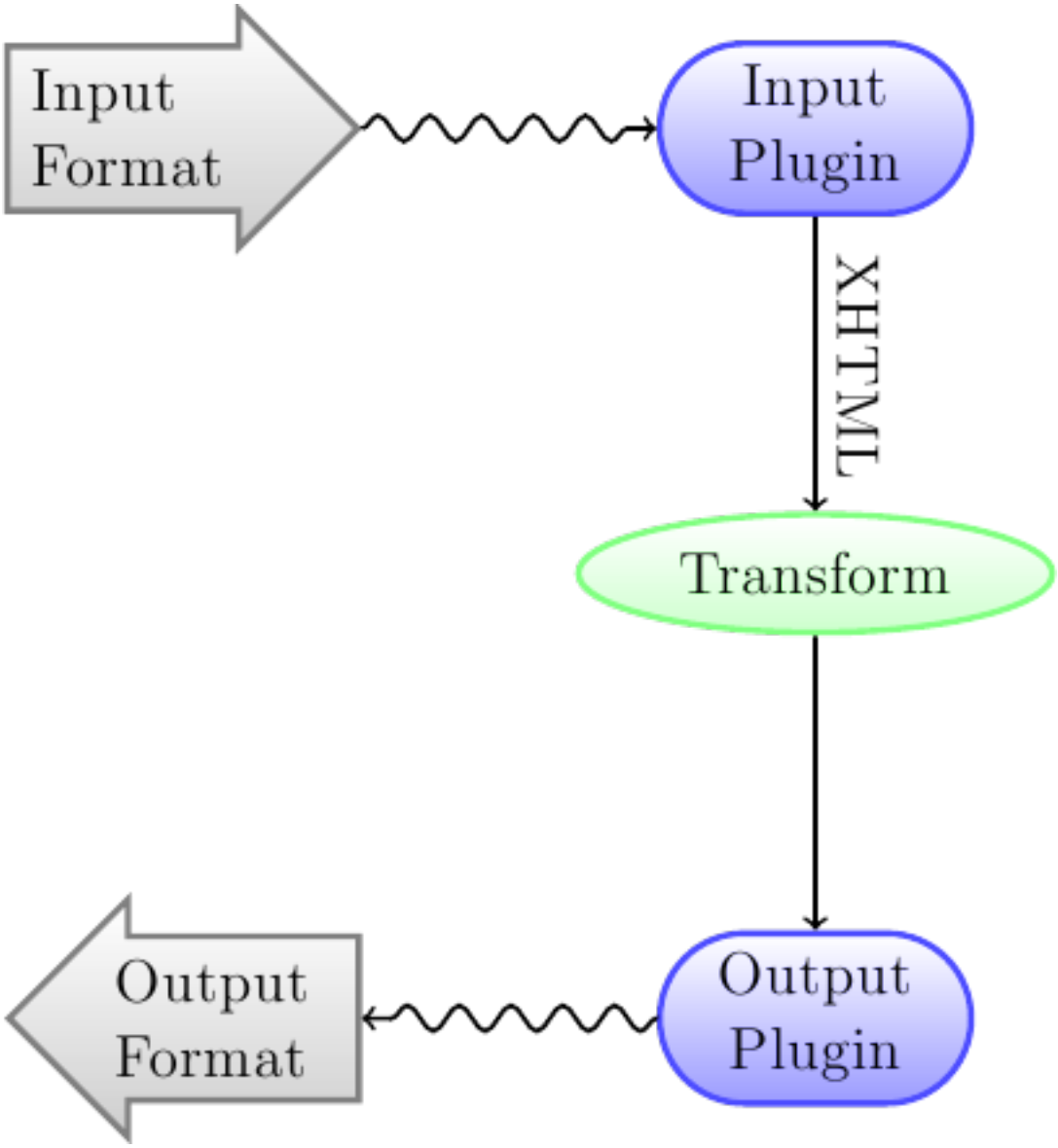


Contenu

- *Introduction* (page 63)
- *Apparence* (page 64)
- *Mise en page* (page 67)
- *Traitement heuristique* (page 68)
- *Rechercher & remplacer* (page 69)
- *Détection de la structure* (page 70)
- *Table des matières* (page 71)
- *Utiliser des images comme titre de chapitre en convertissant des documents sources en HTML.* (page 72)
- *Utiliser les attributs de balise pour le texte des entrées dans la Table des Matières* (page 73)
- *Comment les options sont définies/sauvegardées pour la conversion* (page 73)
- *Conseil spécifique au format* (page 74)

4.1 Introduction

La première chose à comprendre à propos du système de conversion, c'est qu'il fonctionne comme un pipeline. Schématiquement, cela ressemble à ceci :



Le format d'entrée est tout d'abord converti en XHTML par l'*Extension d'Entrée* appropriée. Ce fichier HTML est alors *transformé*. En dernier lieu, le fichier XHTML traité est converti vers le format spécifié par l'*Extension de Sortie* appropriée. Les résultats de la conversion peuvent grandement varier, en fonction du format entrant. Certains formats se convertissent bien mieux que d'autres. Une liste des meilleures sources de format pour la conversion peut être trouvée *ici* (page 134).

C'est dans les manipulations qui s'opèrent sur la sortie XHTML que tout le travail se fait. Il existe de nombreuses manipulations, par exemple, l'insertion des métadonnées en tant que première page du livre électronique, la détection des titres de chapitres pour la création d'une Table des Matières, l'ajustement des tailles des polices, et cetera. Il est important de se souvenir que toutes les manipulations se font sur le fichier XHTML par l' *Extension d'entrée*, pas sur le fichier de base lui-même. Donc, par exemple, si vous demandez à calibre de convertir un fichier RTF vers de l'EPUB, il convertira d'abord en XHTML en interne, les multiples manipulations seront appliquées au fichier XHTML et ensuite l'

Extension de sortie créera le fichier EPUB, générant automatiquement toutes les métadonnées, la Table des Matières, et cetera.

Vous pouvez voir ce procédé en action en utilisant l'option de débogage . Spécifiez juste le chemin vers un dossier pour le fichier de débogage. Pendant la conversion, calibre placera le fichier XHTML généré aux différentes étapes du pipeline de conversion dans différents sous-dossier. Les quatre sous-dossiers sont :

Tableau 1 : Étapes du pipeline de conversion

Dossi	Description
entrée	Ceci contient le fichier HTML fr sortie créé par l'Extension d'origine. Utilisez le pour déboguer l'Extension d'entrée.
analys	Le résultat du prétraitement du fichier et de sa conversion en XHTML par l'Extension d'entrée. Utiliser ceci pour déboguer la détection de structures.
structu	Après la détection de structure, mais avant l'aplanissement du CSS et la conversion des tailles de polices. Utiliser pour déboguer la conversion de la taille des polices et les transformations CSS.
traité	Juste avant que le livre numérique ne soit passé à l'extension de sortie. Utiliser ceci pour déboguer l'extension de sortie.

Si vous souhaitez légèrement éditer le document d'entrée avant que calibre ne le convertisse, la meilleure chose à faire est d'éditer les fichiers dans le sous-dossier `Input`, puis de les archiver, et d'utiliser l'archive ZIP en tant que fichier d'entrée pour les conversions suivantes. Pour ceci, utiliser la boîte de dialogue *Édition des méta informations* pour ajouter l'archive ZIP en tant que format pour le livre et ensuite, dans le coin supérieur gauche de la boîte de dialogue de conversion, sélectionner ZIP comme le format d'entrée.

Ce document s'occupera principalement des différentes manipulations qui agiront sur le fichier XHTML intermédiaire et sur comment les contrôler. À la fin se trouve des astuces pour chaque format d'entrée/de sortie.

4.2 Apparence

Contenu

- *Polices* (page 65)
- *Texte* (page 66)
- *Mise en page* (page 66)
- *Esthétique* (page 67)
- *Transformer les styles* (page 67)
- *Transformer du HTML* (page 67)

Ce groupe d'options contrôle de nombreux aspects de l'apparence et de la présentation du livre numérique converti.

4.2.1 Polices

Une des meilleures particularités lors de l'expérience de lecture électronique est la possibilité de facilement réajuster la taille des polices pour s'adapter aux besoins personnels et conditions d'éclairage. calibre a des algorithmes sophistiqués pour s'assurer que tous les livres ont une taille de police cohérente, quelle que soit la taille de police spécifiée dans le fichier d'entrée

La taille de police de base est la taille de police la plus utilisée dans ce document, càd, la taille de la masse de texte dans le document. Quand vous spécifiez un *Taille de police de base*, calibre redimensionne automatiquement toutes les tailles de police proportionnellement, pour que la taille de police la plus utilisée devienne la taille spécifiée et que les autres tailles se dimensionnent adéquatement. En choisissant une taille plus large, vous pouvez augmenter la taille des polices du document, et vice versa. Quand vous fixez la taille de police de base, il vaut mieux aussi fixer la clef de taille de police .

Normalement, calibre choisira automatiquement une taille de police de base appropriée au profil de sortie que vous avez choisi (voir *Mise en page* (page 67)). Toutefois, vous pouvez outrepassez ceci si le choix par défaut ne vous convient pas.

L'option *Clef de taille de police* vous permet de contrôler comment la taille des polices qui ne sont pas celles de base sont redimensionnées. L'algorithme de redimensionnement des polices utilise une clef de taille de police, qui est simplement une liste séparée par des virgules des tailles de police. La clef des tailles de police donne à calibre combien de fois plus grosse ou plus petite une police devrait être par rapport à la police de base. L'idée est qu'il devrait y avoir une quantité limitée de tailles de police dans un document. Par exemple, une taille pour le corps de texte, quelques tailles pour les différents niveaux de titres et quelques tailles pour les indices, exposants et pied de pages. La clef de taille de police permettra à calibre de compartimenter les tailles de police dans le document entrant en différentes « boîtes » correspondantes aux différentes tailles de police logique.

Illustrons avec une exemple. Supposons que le document source que l'on convertit soit produit par quelqu'un qui possède une excellente acuité visuelle et utilise une taille de police de base de 8pt. Ce qui veut dire que la masse de texte du document est à 8pt, alors que les titres sont légèrement plus large (disons 10 et 12pt) et les pied de pages un peu plus petit à 6pt. Maintenant, si nous utilisons les paramètres suivant

```
Base font size : 12pt
Font size key  : 7, 8, 10, 12, 14, 16, 18, 20
```

Le document de sortie aura une taille de police de base de 12pt, des titres de 14pt et 16pt et des pieds de pages de 8pt. Supposons maintenant que nous voulions que les plus gros titres ressortent mieux et que les pieds de pages soient aussi légèrement plus larges. Pour arriver à ce résultat, la clef de taille de police devrait être changée en

```
New font size key : 7, 9, 12, 14, 18, 20, 22
```

Le plus gros titres feront maintenant 18pt, alors que les pieds de pages feront 9pt. Vous pouvez jouer avec ces options pour essayer de trouver quels seraient les conditions optimale pour vous en utilisant l'assistant de redimensionnement des polices, qui peut être accédé en cliquant sur le petit bouton à coté du paramètre *Clef de taille de police*.

Tous les redimensionnements de taille de police peuvent aussi être désactivé ici, si vous désirez conserver les tailles de police du document original.

Un paramètre lié est *Hauteur de ligne*. Hauteur de ligne contrôle la longueur verticale des lignes. Par défaut, (une hauteur de ligne de 0), aucune manipulation n'est effectuée sur la hauteur des lignes. Si vous spécifiez une valeur différente de 0, la hauteur des lignes sera fixée à tous les endroits qui ne spécifient pas leur propre hauteur de ligne. Toutefois, cet outil est « brutal » et devrait être utilisé avec modération. Si vous voulez ajuster la hauteur des ligne de certaines section du fichier original, il vaut mieux utiliser le *Extra-CSS* (page 67).

Dans cette section, vous pouvez également dire à calibre d'incorporer toutes les polices référencées dans le livre. Cela permettra aux polices de fonctionner sur les liseuses même si elles ne sont pas disponibles sur le périphérique.

4.2.2 Texte

Le texte peut être justifié ou non. Le texte justifié comporte des espaces supplémentaires entre les mots pour donner une marge droite lisse. Certains préfèrent le texte justifié, d'autres non. Normalement, calibre conserve la justification dans le document original. Si vous voulez l'écraser, vous pouvez utiliser l'option *Justification du texte* dans cette section.

Vous pouvez également indiquer le calibre d'*Améliorer la ponctuation* qui remplacera les guillemets, tirets et ellipses par leurs alternatives typographiquement correctes. Notez que cet algorithme n'est pas parfait et qu'il vaut la peine d'examiner les résultats. L'inverse, à savoir, *Simplifier la ponctuation* est également disponible.

Enfin, il y a *Encodage des caractères d'entrée*. De vieux documents peuvent parfois ne pas spécifier leur encodage de caractère. Une fois convertis, cela peut entraîner une corruption des caractères ne faisant pas partie du français ou des caractères spéciaux comme les guillemets intelligents. calibre essaye de détecter automatiquement l'encodage de caractère du document source, mais n'y arrive pas toujours. Vous pouvez le forcer à supposer un encodage de caractère particulier en utilisant ce paramètre. *cp1252* est un encodage classique pour les documents qui ont été réalisé par des programmes utilisant Windows. Vous devriez aussi lire *Comment puis-je convertir mon fichier contenant des caractères non-anglais, ou des guillemets intelligents ?* (page 134) pour plus d'information sur les problèmes d'encodage.

4.2.3 Mise en page

Normalement, les paragraphes d'un fichier XHTML sont rendus avec un saut de ligne et sans retrait de texte. calibre a plusieurs paramètres qui permettent de contrôler cela. *Retirer l'espace entre les paragraphes* s'assure qu'aucun des paragraphes n'ont d'espace entre eux. Il fixe aussi le retrait de texte à 1.5em (cela peut être changé) pour marquer le début de chaque paragraphe. *Insère une ligne vide* fait l'opposé, garantissant qu'il y ait bien un saut de ligne entre chaque paragraphe. Chacun de ces paramètres sont complet, retirant les espaces, ou en insérant pour « tous » les paragraphes (techniquement, les balises <p> et). Ceci est fait afin que vous puissiez régler le paramètre et être sûr qu'il fonctionne comme annoncé, quel que soit l'état du fichier original. La seule exception est lorsque le fichier d'origine utilise des sauts de ligne poussifs pour augmenter l'espace entre les paragraphes

Si vous souhaitez retirer l'espace entre tous les paragraphes à part une petite partie, n'utilisez pas ces options. Ajouter ces lignes de CSS au fichier *Extra-CSS* (page 67) :

```
p, div { margin: 0pt; border: 0pt; text-indent: 1.5em }
.spacious { margin-bottom: 1em; text-indent: 0pt; }
```

Ensuite, dans le document source, marquer les paragraphes qui nécessite un espacement avec *class= »spacious »*. Si votre document original n'est pas en HTML, utiliser l'option de Débogage, décrite dans l'Introduction pour obtenir de l'HTML (utiliser le sous dossier `input`).

Une autre option utile est *Transformer les tableaux en texte*. Certains documents mal conçus utilisent les tableaux pour contrôler l'affichage du texte dans la page. Convertis, ces documents ont souvent le texte qui déborde de la page et d'autres dysfonctionnements. Cette option extraira le contenu des tableaux et le présentera dans un mode linéaire. Noter que cette option linéarise tous les tableaux, à utiliser uniquement si vous êtes sûr que le document source n'utilise pas de tableaux à des fins appropriées, comme la présentation de données.

4.2.4 Esthétique

L'option *CSS supplémentaire* vous permet de spécifier du CSS quelconque qui sera appliqué à tous les fichiers HTML dans la source. Ce CSS est appliqué en priorité et devrait donc outrepasser la plupart du CSS présent dans le **document source** lui-même. Vous pouvez utiliser ce paramètre pour mettre au point finement la présentation/disposition de votre document. Par exemple, si vous voulez que tous les paragraphes de class *endnote* soient alignés sur la droite, ajoutez juste

```
.endnote { text-align: right }
```

ou si vous souhaitez changer la découpure de chaque paragraphe

```
p { text-indent: 5mm; }
```

Extra CSS est une option très puissant, mais il est nécessaire de comprendre comment le CSS fonctionne pour pouvoir l'utiliser à son plein potentiel. Vous pouvez utiliser l'option pipeline de débogage ci-dessus pour voir quel CSS est présent dans votre document d'origine.

Une option plus simple est d'utiliser *Filtrer l'information de style*. Ceci vous permet de supprimer toutes les propriétés CSS, des types spécifiés, du document. Par exemple, vous pouvez l'utiliser pour supprimer toutes les couleurs ou polices.

4.2.5 Transformer les styles

Il s'agit de l'instrument le plus puissant en matière de style. Vous pouvez l'utiliser pour définir des règles qui modifient les styles en fonction de diverses conditions. Par exemple, vous pouvez l'utiliser pour changer toutes les couleurs vertes en bleu, ou supprimer tous les styles gras du texte ou colorier tous les titres d'une certaine couleur, etc.

4.2.6 Transformer du HTML

Similaire aux transformations de styles, mais vous permet d'apporter des modifications au contenu HTML du livre. Vous pouvez remplacer une balise par une autre, ajouter des classes ou d'autres attributs aux balises en fonction de leur contenu, etc.

4.3 Mise en page

Les options de *Mise en page* sont faites pour contrôler la disposition de l'écran, comme les marges et les tailles d'écran. Il comprend des options permettant de régler les marges des pages, qui seront utilisées par l'extension de sortie si le format de sortie supporte les marges. De plus, il vous faudra décider d'un profil d'entrée et d'un profil de Sortie. Chaque ensemble de profil gère comment interpréter les mesures dans les documents d'entrée/de sortie, les tailles d'écrans et les clefs de redimensionnement de police par défaut.

Si vous savez que le fichier que vous convertissez était destiné à être utilisé sur un dispositif/une plate-forme logicielle particulière, choisissez le profil d'entrée correspondant, sinon choisissez simplement le profil d'entrée par défaut. Si vous savez que les fichiers que vous produisez sont destinés à un type d'appareil particulier, choisissez le profil de sortie correspondant. Sinon, choisissez l'un des profils de sortie génériques. Si vous convertissez en MOBI ou AZW3, vous voudrez presque toujours choisir l'un des profils de sortie Kindle. Sinon, votre meilleur choix pour les appareils modernes de lecture de livres électroniques est de choisir le profil de sortie *Generic e-ink HD*.

Le profil de Sortie contrôle aussi la taille de l'écran. Ceci induira, par exemple, le redimensionnement automatique des images afin qu'elle tienne à l'écran pour certains formats de sortie. Choisissez donc un profil de périphérique qui a une taille proche de votre taille d'écran.

4.4 Traitement heuristique

Le traitement heuristique offre une variété de fonctions qui peuvent être utilisée pour détecter et corriger des problèmes communs dans des fichiers source médiocrement formatés. Utilisez cette option si votre document source souffre d'un formatage médiocre. Vu que ces fonctions reposent sur des modèles communs, sachez que dans certain cas le résultat peut s'avérer pire, à utiliser avec précaution donc. Par exemple, plusieurs de ces options retireront tous les espaces qui ne sont pas insécable, ou pourrait inclure des faux positifs dépendant de la fonction.

Activer le traitement heuristique
Cette option active l'étape *Traitement heuristique* du pipeline de conversion de calibre. Elle doit être activée pour que différentes sous-fonctions s'appliquent.

Redéfinir les lignes
Activer cette option forcera calibre à essayer de détecter et corriger les sauts de ligne brutaux qui existe au sein du document en utilisant des indices dans la ponctuation et la longueur des lignes. calibre essayera d'abord de déterminer si des sauts de ligne brutaux existent, et s'ils n'en apparaissent aucun calibre n'essayera pas de redéfinir les lignes. Le facteur de redéfinition des lignes peut-être réduit sir vous souhaitez « forcer » calibre à redéfinir les lignes

Facteur de redéfinition des lignes
Cette option contrôle l'algorithme utilisé par calibre pour supprimer les ruptures de ligne. Par exemple, si la valeur de cette option est de 0,4, cela signifie que calibre supprimera les sauts de ligne à la fin des lignes dont la longueur est inférieure de 40 % de toutes les lignes du document. Si votre document ne comporte que quelques sauts de ligne qui doivent être corrigés, cette valeur doit être réduite à une valeur comprise entre 0,1 et 0,2.

Détecter et mettre en forme les titres de chapitre et les sous-titres sans mise en forme
Si votre document n'a pas de format différent pour les titres et en-têtes de chapitre par rapport au reste du texte, calibre peut utiliser cette option pour essayer de les détecter et de les entourer d'une balise de titre. Des balises <h2> sont utilisées pour les en-têtes de chapitres ; Des balises <h3> sont utilisées pour tout titre détectés.

Cette fonction ne créera pas une TdM, mais dans de nombreux cas, cela permettra à la détection des chapitres par défaut de calibre de détecter les chapitres correctement et de construire une TdM. Ajuster l'XPath sous Détection de la Structure si une TdM n'est pas créée automatiquement. S'il n'y a aucun autre titre utilisé dans le document, alors régler Détection de Structure à « //h :h2 » sera le moyen le plus simple de créer une TdM pour votre document.

Les titres insérés ne sont pas formaté, pour appliquer un format, utiliser l'option *Extra CSS* sous le paramètre Apparence et Présentation de conversion. Par exemple, pour centrer les titres, utiliser ceci

```
h2, h3 { text-align: center }
```

Renuméroter les séquences de balises <h1> ou <h2>
Certains formats de publication utilise de multiple balises <h1> ou <h2> successivement pour leurs titres. Les paramètres de conversion par défaut de calibre couperont ces titres en 2 parties. Cette option renumérotera les balises de titre pour éviter cette scission.

Supprime les lignes blanches entre les paragraphes
Cette option permet à calibre d'analyser les lignes vides incluses dans le document. Si chaque paragraphe est entrecoupé d'une ligne blanche, calibre supprimera tous ces paragraphes blancs. Les séquences de plusieurs lignes vides seront considérées comme des sauts de scène et conservées comme un seul paragraphe. Cette option diffère de l'option *Supprimer l'espacement des paragraphes* sous *Apparence et présentation" Cette option peut également supprimer les paragraphes qui ont été insérés à l'aide de l'option :guilabel :`Insérer une ligne vide* de calibre.

Garantir une mise en forme cohérente des séparations entre deux scènes
Grâce à cette option, calibre essayera de détecter des marqueurs de ruptures de scènes commun et s'assurera qu'ils sont alignés au centre. Des marqueurs de rupture de scène « doux », càd les rupture de scène qui ne sont définies que par des espaces blanc additionel, sont créer pour s'assurer qu'ils ne s'affichent pas de pairs avec de saut de ligne.

Remplacer les symboles de séparation de scène

Si cette option est configurée alors calibre remplacera les marqueurs de rupture de scène qu'il trouve avec le texte spécifié par l'utilisateur. Noter que certains caractères ornementaux peuvent ne pas être supportés sur tous les périphériques.

En général il vaudrait mieux éviter d'utiliser des balises HTML, calibre écartera toutes les balises et utilisera un balisage pré-défini. Les balises <hr />, càd les règles horizontales, et les balises <img> font exceptions. Les règles horizontales peuvent optionnellement être spécifiées dans les styles, si vous choisissez d'ajouter votre propre style pour vous assurer qu'il comprend le paramètre "largeur", sinon les informations de style seront écartées. Les balises d'images peuvent être utilisées, mais calibre ne permet pas d'ajouter des images durant la conversion, cela doit être faire après coup en utilisant la fonctionnalité "Éditer le livre".

Exemple de balise d'image (placer l'image dans un dossier "Images" dans l'EPUB après conversion) :

Exemple de règle horizontale avec styles :

<hr style=»width :20%;padding-top : 1px;border-top : 2px ridge black;border-bottom : 2px groove black;»/>

Enlever les traits d'union superflus

calibre analysera tout le contenu présentant des traits d'union dans le document lorsque cette option est activée. Le document lui-même est utilisé en tant que base d'analyse. Cela permet à calibre de retirer avec précision les traits d'unions pour n'importe quel mot du document quelque soit la langue, ainsi que pour les mots inventés et les mots scientifiques. Le problème principal est que les mots n'apparaissant qu'une fois dans le document ne seront pas modifiés. L'analyse s'effectue en deux passages, le premier analyse les fins de lignes. Les lignes sont redéfinies uniquement si le mot existe avec ou sans trait d'union dans le document. Le second passage analyse tous les mots comprenant des traits d'union dans le document, les traits d'unions sont retirés si les mots existent autre part dans le document sans tiret.

Mettre en italique les mots et les schémas usuels

Si active, calibre cherchera des mots et modèles qui dénote de l'italique et les mettra en italique. Par exemple les conventions habituelles de texte comme « mot » ou les phrases qui devrait normalement être en italique, par exemple les phrases latines comme "etc" ou "et cetera"

Remplacer les mises en retrait par des retraits CSS

Certains documents prennent comme convention d'utiliser des espaces insécables pour les retraits de texte. Lorsque cette option est activée calibre tentera de détecter ce type de formatage et les convertira à un retrait de texte de 3% en utilisant le CSS.

4.5 Rechercher & remplacer

Ces options sont principalement utiles pour la conversion de documents PDF ou OCR, mais elles peuvent aussi être utilisée pour arranger beaucoup de documents avec des problèmes spécifiques. Par exemple, certaines conversion laissent des en-têtes ou des pieds de page dans le texte. Ces options utilisent des expressions régulière pour essayer de détecter en-têtes, pied de page et autres textes quelconques pour les retirer ou les remplacer. Gardez en tête qu'ils opèrent sur le fichier XHTML intermédiaire produit par le pipeline de conversion. Il existe un assistant pour vous aidez à customiser les expressions habituelles dans votre document. Cliquez sur la baguette magique sous le la boîte de dialogue, et puis sur le bouton "Test" après avoir entrer votre expression de recherche. Les résultats probant seront surlignés en Jaune

La recherche fonctionne en utilisant une expression habituelle en Python. Tout le texte identifié sera retiré ou remplacer par le modèle de remplacement. Le modèle de remplacement est optionnel, si laissé blanc alors le texte correspondant à la recherche sera supprimé du document. Vous pouvez en apprendre plus sur les expressions régulières et leur syntaxe dans *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225).

4.6 Détection de la structure

La détection de structure pousse calibre à essayer de détecter au mieux les éléments structuraux dans le document original, quand ils ne sont pas spécifiés correctement. Par exemple, les chapitres, saut de page, en-tête, pied de page, etc. Comme vous pouvez l'imaginer, ce processus change grandement d'un livre à l'autre. Heureusement, calibre a de puissantes options pour le contrôler. Avec la puissance vient la complexité, mais une fois que vous avez pris le temps de comprendre cette complexité, vous vous rendrez compe que votre effort en valait le coup.

4.6.1 Chapitres et sauts de page

calibre a deux ensemble d'options pour *détection des chapitres* et *insertion des sauts de page*. Cela peut parfois être légèrement déboussolant, car par défaut, calibre insérera automatiquement un saut de page après un chapitre détecté mais aussi après les endroits perçu par l'option de détection de saut de page. La raison est qu'il existe de nombreux emplacements où des sauts de page devrait être inséré mais qui ne sont pas des limites de chapitres. Aussi, les chapitres détecté peuvent optionnellement être inséré dans la Table des Matières générée automatiquement.

calibre utilise *XPath*, un langage puissant pour permettre à l'utilisateur de spécifier les limites de chapitres/sauts de page. XPath peut sembler un peu décourageant au premier abord, mais heureusement, il y a le *tutoriel XPath* (page 167) dans le Manuel de l'Utilisateur. Gardez en tête que la Détection de la structure s'opère sur le fichier XHML intermédiaire produit par le pipeline de conversion. Utilisez l'option de débogage décrite dans *Introduction* (page 63) pour comprendre les paramètres appropriés pour votre livre. Il existe aussi un bouton pour un assistant XPath pour vous aider à générer des expressions XPath simples.

Par défaut, calibre utilise les expressions suivantes pour la détection de chapitre :

```
//*[((name()='h1' or name()='h2') and re:test(., 'chapter|book|section|part\s+', 'i
↪')) or @class = 'chapter']
```

Cette expression est assez complexe, car elle essaye de gérer un certain nombre de cas généraux simultanément. Ce qui veut dire que calibre considérera que les chapitres commencent soit à des balises *<h1>* ou *<h2>* qui ont les mots *(chapitre, livre, section ou partie)* en elles ou qui ont l'attribut *class= »chapter »*.

Une option apparentée est *Marque de chapitre*, qui vous permet de contrôler ce que calibre fait lorsqu'il détecte des chapitres. Par défaut, il insérera un saut de page avant le chapitre. Vous pouvez aussi insérez une ligne à la place, ou en plus du saut de page. Il peut aussi ne rien faire.

Le paramètre par défaut pour détecter les sauts de page est

```
//*[name()='h1' or name()='h2']
```

ce qui veut dire que calibre insérera un saut de page avant chaque balise *<h1>* et *<h2>* par défaut.

Note : Les expressions par défaut peuvent changer en fonction du format d'origine que vous convertissez.

4.6.2 Divers

Il y a quelque options de plus dans cette section.

Insérer les métadonnées sur une page au début du livre
: Une des choses pratique avec calibre c'est qu'il vous permet d'entretenir des métadonnées très complètes sur tous vos livres, par exemple un classement, les étiquettes, les commentaires, etc. Cette option créera une page unique avec toutes les métadonnées et l'insérera dans le livre numérique converti, habituellement juste après la couverture. Pensez-y pour créer votre propre jaquette personnalisée.

Retirer la première image
: Parfois, le document source que vous convertissez inclue la couverture dans le livre, au lieu de la séparer. Si vous spécifiez aussi une couverture dans calibre, alors le livre converti en aura deux. Cette option enlèvera simplement la première image du document source, s'assurant ainsi que le livre converti à uniquement une couverture, celle spécifiée dans calibre.

4.7 Table des matières

Quand le document source a une Table des Matières dans ses métadonnées, calibre l'utilisera simplement. Cependant, soit certain vieux formats ne supportent pas les Table des Matières intégrée dans les métadonnées, soit certains livres n'en ont tout simplement pas. Dans ces cas là, les options dans cette section peuvent vous aider à générer automatiquement une Table des Matières dans le livre numérique converti, basée sur le contenu réel du document source.

Note : L'utilisation de ces options peut être un peu difficile pour un résultat parfait. Si vous préférez créer/éditer la Table des Matières à la main, convertissez en EPUB ou en AZW3 et sélectionner la case en bas de la section Table des Matières dans la boite de dialogue de conversion intitulé *Affiner la Table des matière manuellement après conversion*. Ceci lancera l'éditeur de TdM après la conversion. Cela permet de créer des entrées dans la Table des Matières simplement en cliquant dans le livre à l'endroit où vous voulez que l'entrée pointe. Vous pouvez aussi utiliser l'éditeur de TdM seul, sans conversion. Allez à *Préférences → Interface → Barre d'Outils* et ajouter *Éditeur de TdM* à la barre d'outil principale. Sélectionner alors simplement le livre que vous souhaitez éditer et cliquez sur le bouton *Éditeur de TdM*.

La première option est *Utilisation forcée de la Table des Matières générée automatiquement*. En cochant cette option vous pouvez forcer calibre à outrepasser toutes les Table des Matières qu'il trouve dans les métadonnées du document source et à en générer une.

La manière de fonctionner par défaut de la création de Table des Matières automatique est que, calibre essayera d'abord d'ajouter tout chapitres détectés à la table des matières générée. Vous pouvez apprendre comment personnaliser la détection des chapitres dans la section *Détection de la structure* (page 70) ci-dessus. Si vous ne souhaitez pas inclure les chapitres détectés dans la table des matières générée, coché l'option *Ne pas ajouter les chapitres détectés*.

Si moins de chapitres que *Seuil de chapitre* sont détecté, calibre ajoutera alors tout hyperlien qu'il trouve dans le document à la Table des Matières. Cela fonctionne souvent bien car beaucoup de document source incluent dans la Table des Matières en lien hypertexte. L'option *Nombre de lien* peut être utilisée pour contrôler ce comportement. Si fixé à zéro, aucun lien ne sera ajouté. Si fixé à un nombre plus grand que zéro, le nombre de chapitre ajouté sera maximum de ce nombre.

calibre filtrera automatiquement les doublons dans la Table des Matières générée. Cependant, s'il y a d'autres entrées indésirable, vous pouvez les filtrer en utilisant l'option *Filtre TdM*. Ceci est une expression régulière qui fera correspondre le titre des entrées dans la table des matières générée. Dès qu'une correspondance est trouvée, elle sera supprimée. Par exemple, pour retirer toutes les entrée « Next » ou « Previous » utiliser

```
Next|Previous
```

L'option *TdM niveau 1,2,3* vous permet de créer une Table des Matières sophistiquée à plusieurs niveaux. Ce sont des expressions XPath qui correspondent à des balises dans le fichier XHTML intermédiaire produit par le pipeline

de conversion. Voir *Introduction* (page 63) sur comment avoir accès à ce fichier XHTML. Lire aussi le *Tutoriel XPath* (page 167), pour apprendre comment construire des expressions XPath. À côté de chacune des options il y a un bouton qui lance un assistant pour aider à la création d'expressions XPath basiques. Les exemples suivant illustreront comment utiliser ces options.

Supposons que vous avez un document source qui donne un fichier XHTML qui ressemble à ceci :

```
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>Sample document</title>
    </head>
    <body>
        <h1>Chapter 1</h1>
        ...
        <h2>Section 1.1</h2>
        ...
        <h2>Section 1.2</h2>
        ...
        <h1>Chapter 2</h1>
        ...
        <h2>Section 2.1</h2>
        ...
    </body>
</html>
```

Alors, nous fixons les options à

```
Level 1 TOC : //h:h1
Level 2 TOC : //h:h2
```

Ceci résultera en une Table des Matières générée automatiquement à deux niveaux qui ressemblera à

```
Chapter 1
    Section 1.1
    Section 1.2
Chapter 2
    Section 2.1
```

Avertissement : Les formats de sortie ne supportent pas tous les Tables des Matières à plusieurs niveaux. Il vaut mieux d'abord essayer en format EPUB. Si cela fonctionne, alors essayez votre format de préférence.

4.8 Utiliser des images comme titre de chapitre en convertissant des documents sources en HTML.

Supposons que vous souhaitiez utiliser une image comme titre de chapitre, mais vous souhaitez toujours que calibre soit capable de générer automatiquement une Table des Matières pour vous à partir des titres des chapitres. Utilisez le balisage HTML suivant pour faire cela :

```
<html>
    <body>
        <h2>Chapter 1</h2>
        <p>chapter 1 text...</p>
        <h2 title="Chapter 2"><img src="chapter2.jpg" /></h2>
```

(suite sur la page suivante)

(suite de la page précédente)

```
        <p>chapter 2 text...</p>
    </body>
</html>
```

Fixer le paramètre *TdM à 1 niveau* à `//h:h2`. Ensuite, pour le chapitre deux, calibre prendra le titre à partir de la valeur de l'attribut ``title` sur la balise <h2>, vu que la balise ne contient aucun texte.

4.9 Utiliser les attributs de balise pour le texte des entrées dans la Table des Matières

Si vous avez des chapitres particulièrement long et que vous souhaitez des versions raccourcies dans la Table des Matières, vous pouvez utiliser les attributs de titre pour atteindre cela, par exemple :

```
<html>
    <body>
        <h2 title="Chapter 1">Chapter 1: Some very long title</h2>
        <p>chapter 1 text...</p>
        <h2 title="Chapter 2">Chapter 2: Some other very long title</h2>
        <p>chapter 2 text...</p>
    </body>
</html>
```

Fixer le paramètre *TdM à 1 niveau* à `//h:h2/@title`. Alors calibre prendra le titre dans la valeur de l'attribut `title` sur les balises <h2>, au lieu d'utiliser le texte dans la balise. Noter le `/@title` restant dans l'expression XPath, vous pouvez utiliser ceci pour dire à calibre de prendre le texte de n'importe quel attribut.

4.10 Comment les options sont définies/sauvegardées pour la conversion

Il existe deux endroits où les options de conversions peuvent être fixées dans calibre. La première est dans Préférences->Conversion. Ces paramètres sont ceux par défaut pour les options de conversions. Dès que vous essayerez de convertir un nouveau livre, les paramètres fixés ici seront utilisés par défaut.

Vous pouvez aussi changer les paramètres dans la boite de dialogue de conversion à chaque conversion de livre. Quand vous convertissez un livre, calibre se souvient des paramètres utilisés pour ce livre, et si vous souhaitez le convertir à nouveau, les paramètres sauvés pour ce livre auront priorité sur les paramètres par défaut fixés dans *Préférences*. Vous pouvez ramener les paramètres individuels à ceux par défaut en utilisant le bouton *Restaurer par défaut* dans la boite de dialogue de conversion du livre susmentionné. Vous pouvez retirer des paramètres sauvegardés pour un groupe de livre en sélectionnant tous les livres et en cliquant sur le bouton *Édition des métadonnées*, pour ouvrir la boîte de dialogue de l'éditeur de métadonnées multiples, il existe une option pour retirer des paramètres de conversion sauvegardés au pied de cette fenêtre

Lorsque vous convertissez par lot un ensemble de livres, les paramètres sont pris dans l'ordre suivant (le dernier l'emporte) :

- De paramètres par défaut fixés dans Préférences->Conversion
- Des paramètres de conversion sauvé pour chaque livre converti (s'ils existent). Ceci peut-être désactivé par l'option dans le coin supérieur gauche de la boîte de dialogue de Conversion par Lot.
- Des paramètres fixés dans la boîte de dialogue de la Conversion par Lot

Notez que les paramètres finaux pour chaque livre dans une Conversion par Lot sera sauvegardé et réutilisé si ce livre est à nouveau converti. Vu que la priorité la plus haute dans la Conversion par Lot est donnée aux paramètres de la Conversion par Lot, ceux-ci surpasseront tous les paramètres spécifiques de livre. Convertissez donc uniquement par lot des livres qui nécessitent des paramètres similaires. Les exceptions sont les métadonnées et les paramètres de formats entrant spécifique. Vu que la boîte de dialogue de Conversion par Lot n'a pas de paramètres pour ces deux catégories, ils seront pris des paramètres spécifiques du livre (s'ils existent) ou de ceux par défaut.

Note : Vous pouvez voir quels sont les paramètres qui ont été utilisés durant n'importe quelle conversion en cliquant sur l'icone tournante dans le coin inférieur droit, puis en double cliquant sur la tache de conversion individuelle. Ceci fera apparaître un journal de conversion qui contiendra les paramètres qui ont effectivement été utilisé, en haut.

4.11 Conseil spécifique au format

Vous trouverez ici des conseils spécifique quant à la conversion de formats particulier. Les options spécifiques à chaque format, d'entrée ou de sortie, sont disponibles dans le dialogue de conversion sous leur propre section, par exemple *Entrée TXT* ou *Sortie EPUB*.

4.11.1 Convertir des documents Mircosoft Word

calibre peut convertir automatiquement des fichiers `.docx` créer par Microsoft Word 2007 et plus récent. Ajoutez simplement le fichier dans calibre et cliquez sur convertir.

Note : Il y a un fichier .docx de démonstration[27] qui illustrent les capacités du moteur de conversion de calibre. Téléchargez-le et convertissez-le en EPUB ou en AZW3 pour voir ce que calibre peut faire.

---

calibre générera automatiquement une Table des Matières à partir des titres si vous marquez vos titres avec les styles `Titre 1`, `Titre 2`, etc. dans Microsoft Word. Ouvrez le livre numérique sorti dans la Visionneuse de livre numérique de calibre et cliquez sur le bouton *Table des Matières* pour voir la Table des Matières générée.

Des fichiers .doc plus ancien

Pour des fichiers .doc plus ancien, vous pouvez sauver le document en HTML avec Microsoft Word et ensuite convertir le fichier HTML qui en résulte avec calibre. En sauvant en HTML, soyez sûr d'utiliser l'option « Sauvegarder comme une page Web, Filtrée » car cela produira un HTML propre qui se convertira facilement. Notez que Word produit de très mauvais fichier HTML, la conversion peut prendre beaucoup de temps, soyez donc patient. Si vous possédez une version plus récente de Word, vous pouvez aussi le sauver directement en .docx.

Une autre alternative est d'utiliser le logiciel gratuit LibreOffice. Ouvrez votre fichier .doc dans LibreOffice et enregistrez-le au format .docx, qui peut être directement converti dans calibre.

[27] https://calibre-ebook.com/downloads/demos/demo.docx

4.11.2 Convertir des documents TXT

Les documents TXT n'ont aucun moyen bien définis de spécifié le formatage de texte comme le gras, l'italique, etc., ou la structure du document comme les paragraphes, les titres, les sections et ainsi de suite, mais il y une variété de conventions qui sont habituellement utilisées. Par défaut, calibre tente de détecter automatiquement le formatage et le balisage adéquat en fonction de ces conventions.

L'entrée TXT comporte un certain nombre d'option pour différencié comment les paramètres sont détectés.

Style de paragraphe : Auto
Analyse le fichier texte et essaye de déterminer automatiquement comment les paragraphes sont définis. Cette option fonctionne généralement bien, si les résultats obtenus ne sont pas bon, essayez l'une des options manuelles.>

Style de paragraphe : Bloc
Considère que les limites de paragraphes sont un saut de ligne ou plus

```
This is the first.

This is the
second paragraph.
```

Style de paragraphe : Unique
Considère que chaque ligne est un paragraphe :

```
This is the first.
This is the second.
This is the third.
```

Style de paragraphe : Imprimé
Considère que chaque paragraphe commence avec un retrait de texte (soit un tab ou plus de 2 espaces). Les paragraphes finissent quand la prochaine ligne commençant avec un retrait est atteinte.

```
  This is the
first.
  This is the second.

  This is the
third.
```

Style de paragraphe : Non formaté
Considère que le document n'a pas de formatage, mais utilise des sauts de lignes « dûr ». La ponctuation et des lignes de demi-longueur sont utilisée pour recréer les paragraphes.

Style de formatage : Automatique
Essaye de détecter le type de balisage de formatage. Si aucun balisage n'est utilisé alors le formatage heuristique sera utilisé.

Style de formatage : Heuristique
Analyse le document pour des en-têtes de chapitres communs, des ruptures de scènes, et des mots en italique et applique le balisage HTML approprié durant la conversion.

Style de formatage : Markdown
calibre supporte également la syntaxe Markdown pour les entrées TXT. Le Markdown permet d'ajouter un formatage basique au document TXT, comme du gras, de l'italique, des en-têtes de section, des tableaux, des listes, une Table des Matières, etc. Marquer le titre des chapitres avec un # (p. ex. : # Mon titre) et paramétrer l'expression XPath de détection des chapitres à « //h :h1 » est la manière la

plus simple de générer une bonne Table des Matières à partir d'un document TXT. Vous pouvez en apprendre plus sur la syntaxe Markdown ici : daringfireball[28].

Style de formatage : Aucun
N'applique aucun formatage spécial au texte, le document est converti en HTML sans aucun changement.

4.11.3 Conversion de documents PDF

Les documents PDF sont les pires formats à partir desquels on peut convertir. C'est un format à taille de page et placement de texte fixe. Ce qui veut dire qu'il est très difficile de déterminer où un paragraphe commence et l'autre finit. calibre essayera de déballer mes paragraphes en utilisant un *Facteur de déballage de ligne*. Ceci est une échelle utilisée pour déterminer la longueur à laquelle les lignes devraient être déballée. Les valeurs valide sont des décimaux se trouvant entre 0 et 1. LA valeur par défaut est 0.45, juste en dessous de la moitié de la longueur d'une ligne. Descendez cette valeur si vous souhaitez inclure plus de texte au déballage. Augmentez pour en inclure moins. Vous pouvez ajuster cette valeur dans les paramètres de conversion sous guilabel :*Entrée PDF*.

Aussi, ils ont souvent des en-têtes et des pieds de page qui font partie du document et qui seront inclus dans le texte. Utilisez l'option *Rechercher et remplacer* pour retirer les en-têtes et pieds de pages pour réduire ce problème. Si les en-têtes et pieds de page ne sont pas supprimés cela peu endommager le déballage des paragraphes. Pour apprendre à utiliser les options de suppression d'en-tête et de pieds de page, lire *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225).

Quelques limitations des entrée PDF sont :

- Les documents complexe, à multiple colonne et basé sur des images ne sont pas supportés
- L'extraction d'image vectorielle et de tableau à partir du document ne sont pas supportées non plus.
- Certains PDF utilisent des glyphes spéciaux pour représenter ll ou ff ou fi, etc. La conversion ce ceux-ci pourrait ne pas marcher simplement en fonction de comment ils sont représenté à l'intérieur du PDF.
- Les liens et les Tables des Matières ne sont pas supportés
- Les PDFs qui utilisent des polices non-Unicode intégrée pour représenter des caractères qui ne font pas partie du français engendreront une sortie tronquée pour ces caractères
- Certain PDF sont fait de photographies de page avec du texte reconnu par OCR derrière elles. Dans ces cas là, calibre utilise le texte venant de l'OCR, qui peut être fort différent de ce que vous voyez dans le fichier PDF.
- Les PDFs utilisés pour afficher du texte complexe, comme des langages lu de droite à gauche et les composition mathématiques ne se convertiront pas bien

Pour répéter **le PDF est un vraiment, vraiment mauvais** format à utilisé comme source. Si vous devez absolument utiliser du PDF, alors soyez prêt à trouver un fichier de sortie se trouvant n'importe où entre décent et inutilisable, dépendant du PDF source.

[28] https://daringfireball.net/projects/markdown/syntax

4.11.4 Collections de Bandes Dessinées

Une collection de bandes dessinée est un fichier .cbc. Un fichier .cbc est un ZIP qui contient d'autres fichiers CBZ/CBR. Le fichier .cbc doit aussi contenir un simple fichier texte appelé comics.txt, encodé en UTF-8. Le fichier comics.txt doit contenir une liste des bandes dessinées contenue dans le fichier .cbc, sous la forme nom_du_fichier :titre, comme illustré ci-dessous :

```
one.cbz:Chapter One
two.cbz:Chapter Two
three.cbz:Chapter Three
```

Le fichier .cbc contiendra alors :

```
comics.txt
one.cbz
two.cbz
three.cbz
```

calibre convertira automatiquement ce fichier .cbc en un livre numérique avec une Table des Matières pointant vers chaque entrée dans comics.txt

4.11.5 Démonstration de formatage d'EPUB avancée

Divers formatages avancés pour les fichiers EPUB sont expliqués dans ce fichier de démonstration[29]. Le fichier a été créé à partir d'HTML codé à la main en utilisant calibre et est destiné à être utilisé comme modèle pour vos efforts de création d'EPUB.

L'HTML source à partir duquel il a été créé est disponible ici demo.zip[30]. Les paramètres utilisés pour créer l'EPUB à partir du ZIP sont

```
ebook-convert demo.zip .epub -vv --authors "Kovid Goyal" --language en --level1-toc '/
↪/*[@class="title"]' --disable-font-rescaling --page-breaks-before / --no-default-
↪epub-cover
```

Notez que vu que ce fichier explore le potentiel de l'EPUB, la plupart des formatages avancés ne fonctionneront pas sur des lecteurs moins capable que le lecteur intégré de calibre.

4.11.6 Convertir des documents ODT

calibre peut convertir directement les fichiers ODT (OpenDocument Text). Vous devriez utilisé les styles pour formater votre document et minimiser l'utilisation de formatage direct. En insérant des images dans votre document, vous devez les ancrer aux paragraphes, les images qui sont ancrées aux pages finiront au début de la conversion.

Pour permettre la détection automatique des chapitres, vous devez les marquer avec les styles intégrés appelés *Heading 1*, *Heading 2*, …, *Heading 6* (*Heading 1* équivaut à la balise HTML `<h1>`, *Heading 2* à `<h2>`, etc). Lorsque vous convertissez dans calibre, vous pouvez indiquer le style que vous avez utilisé dans la case *Détecter les chapitres à*. Exemple :

- Si vous marquez les chapitres avec le style *Heading 2*, vous devez définir la case « Détecter les chapitres à » sur *`//h :h2`*.
- Pour une table des matières imbriquée avec des sections marquées par *Heading 2* et des chapitres marqués par *Heading 3*, vous devez entrer `//h:h2|//h:h3`. Sur la page Convertir - Table des matières, mettez la case *Niveau 1 de la Table des matières à `//h :h2`* et la case *Niveau 2 de la Table des matières à `//h :h3`*.

[29] https://calibre-ebook.com/downloads/demos/demo.epub
[30] https://calibre-ebook.com/downloads/demos/demo.zip

Les propriétés de document bien connues (Titre, Mots-clefs, Description, Créateur) sont reconnues et calibre utilisera la première image (pas trop petite, et avec de bonnes proportions) comme image de couverture.

Il y a aussi un mode de propriété avancé de conversion qui est activé en définissant la propriété personnalisée `opf.metadata` (de type "Oui ou Non") à Oui dans votre document ODT (Fichier->Propriétés->Propriétés personnalisées). Si cette propriété est détectée par calibre, les propriétés personnalisées suivantes sont reconnues (`opf.authors` remplace le créateur du document) :

```
opf.titlesort
opf.authors
opf.authorsort
opf.publisher
opf.pubdate
opf.isbn
opf.language
opf.series
opf.seriesindex
```

De plus, vous pouvez spécifier l'image à utiliser comme couverture en la nommant `opf.cover` (clic droit, Image->Options->Nom) dans le fichier ODT. Si aucune image de ce nom n'est trouvée, la méthode « intelligente » est utilisée. La détection de la couverture pourrait entraîner des couvertures en double dans certains formats de sortie, le processus supprimera le paragraphe (uniquement si le seul contenu est la couverture !) du document. Mais cela ne fonctionne que avec l'image nommée !

Pour désactiver la détection de couverture vous pouvez définir la propriété personnalisée `opf.nocover` (de type "Oui ou Non") à Oui en mode avancé.

4.11.7 Convertir en PDF

Le premier paramètre, le plus important, à décider lors de la conversion en PDF est la taille de la page. Par défaut, calibre utilise une taille de page « U.S. Letter ». Vous pouvez changer ceci vers un autre standard de taille de page ou une taille complètement personnalisée dans la section *Sortie PDF* de la boîte de dialogue de conversion. Si vous générez un PDF à utiliser sur un périphérique spécifique, vous pouvez activer, à la place, l'option pour utiliser la taille de page à partir de *Profil de sortie* de l'onglet *Mise en page* de la boîte de dialogue de conversion. De cette façon, si votre profil de sortie est paramétré sur Kindle, calibre créera un PDF avec une taille de page adaptée pour la lecture sur le petit écran Kindle.

En-têtes et pieds de page

Vous pouvez ajouter des en-têtes et pieds de page quelconques sur chaque page du PDF en spécifiant des modèles d'en-tête et de pied de page. Les modèles sont simplement des extraits de code HTML qui sont interprétés dans les régions d'en-tête et de pied de page. Par exemple, pour afficher le numéro des pages centré au bas de chaque page, en vert, utilisé le modèle de pied de page suivant :

```
<footer><div style="margin: auto; color: green">_PAGENUM_</div></footer>
```

calibre remplacera automatiquement `_PAGENUM_` avec le numéro de page actuel. Vous pouvez même afficher du contenu différent sur les pages paires et impaires, par exemple, le modèle d'en-tête suivant affiche le titre sur les pages impaires et l'auteur sur les pages paires :

```
<header style="justify-content: flex-end">
    <div class="even-page">_AUTHOR_</div>
    <div class="odd-page"><i>_TITLE_</i></div>
</header>
```

calibre remplacera automatiquement _TITLE_ et _AUTHOR_ avec le titre et l'auteur du document converti. Régler `justify-content` sur `flex-end` fera en sorte que le texte soit aligné à droite.

Vous pouvez aussi afficher du texte aux bords gauche et droit et changer la taille des polices, comme montré dans ce modèle d'en-tête :

```
<header style="justify-content: space-between; font-size: smaller">
    <div>_TITLE_</div>
    <div>_AUTHOR_</div>
</header>
```

Ceci affichera le titre à gauche et l'auteur à droite, dans une taille de police plus petite que le texte principal.

Vous pouvez aussi utiliser la section actuelles dans les modèles, comme montré ci-dessous

```
<header><div>_SECTION_</div></header>
```

SECTION est remplacé par le nom de la section actuelle, quelle qu'il soit. Ces noms sont pris dans la métadonnée Table des Matières dans le document (le PDF sortant). Si le document n'a pas de table des matières, alors il sera remplacé par du texte vide. Si une page PDF unique à plusieurs sections, la première section de la page sera utilisée. Pareillement, il y a une variable nommée _TOP_LEVEL_SECTION_ qui peut être utilisée pour obtenir le nom de la section supérieure actuelle.

Vous pouvez même utiliser du javascript dans les modèles d'en-tête et de pied de page, par exemple, le modèle suivant fera commencer les numéros de page à 4 au lieu de 1 :

```
<footer>
    <div></div>
    <script>document.currentScript.parentNode.querySelector("div").innerHTML = "" + (_
↪PAGENUM_ + 3)</script>
</footer>
```

En outre, il existe d'autres variables que vous pouvez utiliser dans les en-têtes et les pieds de page, documentées ci-dessous :

- _TOTAL_PAGES_ - nombre total de pages dans le fichier PDF, utile pour mettre en place un compteur de progrès, par exemple.
- _TOP_LEVEL_SECTION_PAGES_ - nombre total de pages dans la section haute en cours d'utilisation
- _TOP_LEVEL_SECTION_PAGENUM_ - le numéro de page de la page en cours dans la section haute en cours d'utilisation

---

Note : Lorsque vous ajoutez des en-têtes et des pieds de page, assurez vous que les valeurs des marges de haut et bas de page soit assez large, sous la section *PDF Output* dans la boite de dialogue de conversion.

Table des Matières imprimable

Vous pouvez aussi ajouter une Table des Matières imprimable à la fin du PDF qui liste le numéro de page de chaque section. Ceci est très pratique si vous comptez imprimer le PDF sur papier. Si vous souhaitez utiliser le PDF sur un support électronique, alors l'Aperçu PDF permet cette fonctionnalité et est généré par défaut.

Vous pouvez personnaliser l'aspect de la Table des Matières générée en utilisant le paramètre de conversion Extra CSS sous Apparence dans la boite de dialogue de conversion. Le CSS utilisé par défaut est listé ci-dessous, copier le simplement et appliquer tous les changements que vous souhaitez.

```
.calibre-pdf-toc table { width: 100%% }

.calibre-pdf-toc table tr td:last-of-type { text-align: right }

.calibre-pdf-toc .level-0 {
    font-size: larger;
}

.calibre-pdf-toc .level-1 td:first-of-type { padding-left: 1.4em }
.calibre-pdf-toc .level-2 td:first-of-type { padding-left: 2.8em }
```

Marges de page personnalisées pour les fichiers HTML individuels

Si vous convertissez un EPUB ou AZW3 avec plusieurs fichiers HTML individuels à l'intérieur et que vous souhaitez modifier les marges de page pour un fichier HTML particulier, vous pouvez ajouter le bloc de style suivant au fichier HTML en utilisant l'éditeur de livre numérique de calibre :

```
<style>
@page {
    margin-left: 10pt;
    margin-right: 10pt;
    margin-top: 10pt;
    margin-bottom: 10pt;
}
</style>
```

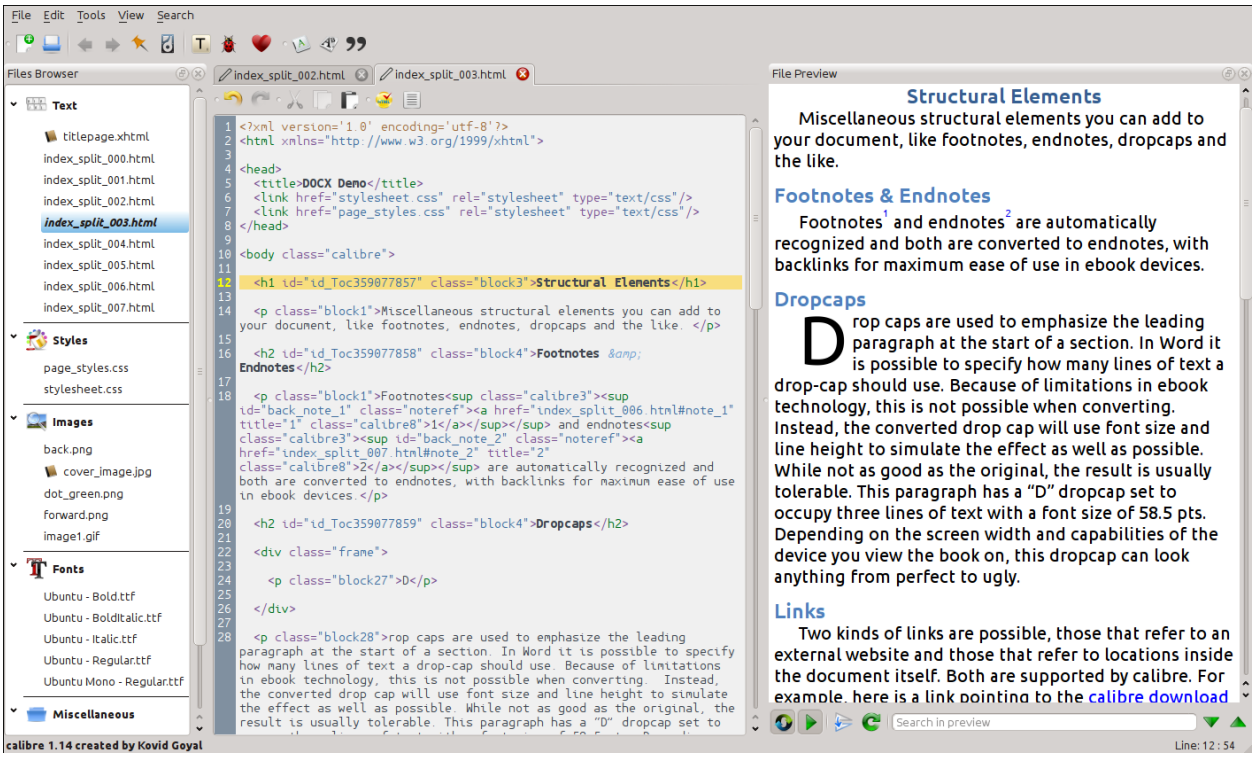
Ensuite, dans la section sortie PDF de la boîte de dialogue de conversion, activez l'option *Utiliser les marges de page du document à convertir*. Maintenant toutes les pages générées à partir de ce fichier HTML auront des marges `10pt`.

CHAPITRE 5

Éditer des livres numériques

calibre a un éditeur de livre numérique intégré qui peut être utilisé pour éditer des livres aux formats EPUB et AZW3 (Kindle). L'éditeur vous montre l'HTML et le CSS qui sont utilisés à l'intérieur des fichiers livre, avec une prévisualisation directe qui se met à jour au fur et à mesure que vous faites des changements. Il contient aussi divers outils automatisés pour effectuer un nettoyage usuel et des tâches de réparation.

Vous pouvez utiliser cet éditeur en faisant un clic droit sur n'importe quel livre dans calibre et en sélectionnant guilabel :*Éditer le livre*.



Contenu

- *Fonctionnement de base* (page 83)
- *L'Explorateur de fichiers* (page 86)
  - *Renommer des fichiers* (page 87)
  - *Fusionner des fichiers* (page 87)
  - *Ordre de fichier texte changeant* (page 88)
  - *Marquage de la couverture* (page 88)
  - *Suppression des fichiers* (page 88)
  - *Exportation de fichiers* (page 88)
  - *Ajout de nouvelles images/polices/etc. ou création de nouveaux fichiers vides* (page 89)
  - *Remplacement des fichiers* (page 89)
  - *Lier des feuilles de style aux fichiers HTML efficacement* (page 89)
- *Rechercher & remplacer* (page 89)
  - *Recherches sauvegardées* (page 90)
  - *Mode fonction* (page 90)
  - *Recherche ignorant les balises HTML* (page 90)
- *Outils automatisés* (page 90)
  - *Edition de la Table des Matières* (page 90)
  - *Vérification du livre* (page 92)
  - *Ajout d'une couverture* (page 93)
  - *Intégration des polices référencées* (page 93)
  - *Créer un sous-ensemble de polices* (page 93)
  - *Amélioration de la ponctuation* (page 93)
  - *Transformation des propriétés CSS* (page 94)
  - *Suppression en cours des règles CSS inutilisées* (page 94)
  - *Correction de l'HTML* (page 94)
  - *Embellissement des fichiers* (page 94)
  - *Insertion d'une Table des Matières* (page 95)
  - *Configuration de Sémantique* (page 95)
  - *Filtrage des informations de style* (page 95)
  - *Mise à jour des éléments du livre* (page 95)
- *Points de Contrôle* (page 95)
- *Le Panneau de prévisualisation directe* (page 97)
  - *Fractionnement des fichiers HTML* (page 98)
- *Le panneau CCS en temps réel* (page 99)

- *Outils divers* (page 100)
  - *La vue Table des Matières* (page 100)
  - *Vérification de l'orthographe des mots dans le livre* (page 100)
  - *Insérer des caractères spéciaux* (page 102)
  - *La vue inspecteur de code* (page 103)
  - *Vérifications des liens externes* (page 103)
  - *Téléchargement de ressources externes* (page 103)
  - *Organisation des fichiers par type dans des dossiers* (page 103)
  - *Importation de fichiers dans d'autres formats de livres numériques comme l'EPUB* (page 103)
  - *L'outil Rapports* (page 115)
- *Fonctionnalités spéciales dans l'éditeur de code* (page 115)
  - *Mise en évidence de la syntaxe* (page 115)
  - *Aide sensible au contexte* (page 116)
  - *Auto-complétion* (page 116)
  - *Fragments* (page 116)

5.1 Fonctionnement de base

Note : Une présentation vidéo de l'éditeur de livre numérique de calibre est disponible ici[31].

Quand vous ouvrez un livre la première fois avec l'outil d'Édition de livre, vous serez en présence d'une liste de fichiers sur la gauche. Ceux-ci sont les fichiers HTML individuels, les feuilles de style, les images, etc. qui forment le contenu du livre. Double cliquez simplement sur un fichier pour démarrer son édition. Notez que si vous voulez faire quelque chose de plus sophistiqué que de faire quelques petits ajustements, vous aurez besoin de connaître le Tutoriel HTML[32] et le Tutoriel CSS[33].

Au fur et à mesure que vous faites des changements au HTML ou au CSS dans l'éditeur, les changements seront prévisualisés, en direct, dans le panneau de prévisualisation à gauche. Quand vous êtes content avec l'apparence des changements que vous avez fait, cliquez sur le bouton *Enregistrer* ou utilisez *Fichier → Enregistrer* pour sauvegarder vos changements dans le livre.

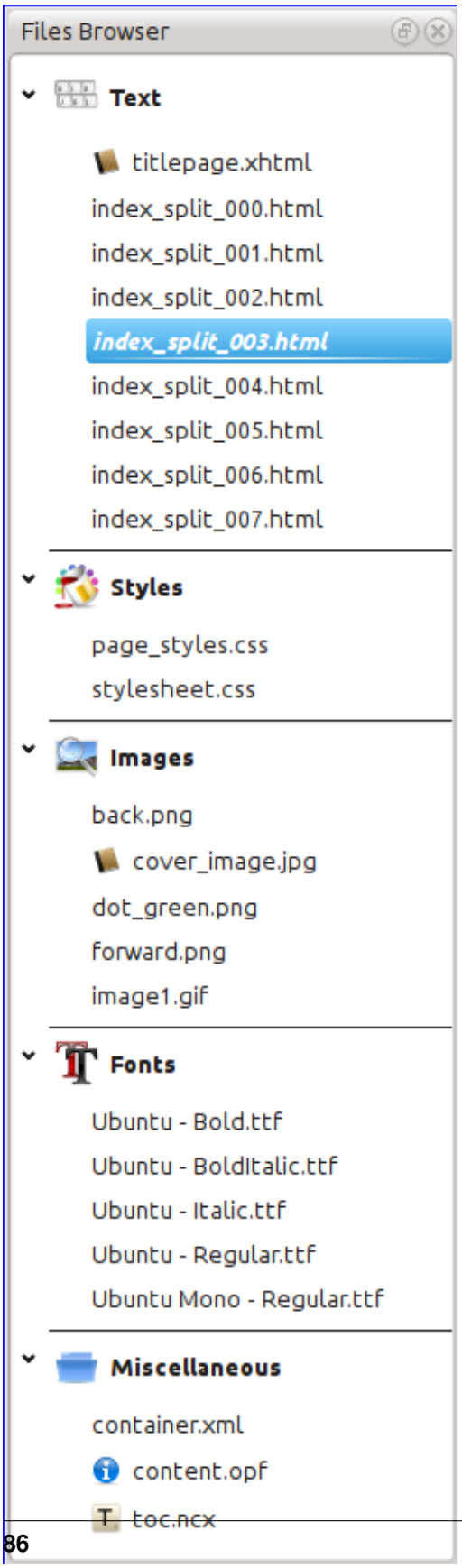
Une caractéristique intéressante est *Points de contrôle*. Avant de vous embarquer dans un ensemble ambitieux d'éditions, vous pouvez créer un point de contrôle. Le point de contrôle préservera l'état actuel de votre livre, dès lors si dans le futur vous décidez que vous n'appréciez pas les changements que vous lui avez fait vous pouvez revenir à l'état au moment où vous avez créer le point de contrôle. Pour créer un point de contrôle, utilisez *Éditer → Créer un point de contrôle*. Des points de contrôle seront aussi créés automatiquement pour vous au moment où vous exécutez n'importe quel outil automatisé comme le rechercher et remplacer global. La fonctionnalité point de contrôle est en complément au mécanisme normal Annuler/Restaurer lors de l'édition de fichiers individuels. Les points de contrôle vous sont utiles quand les changements sont disséminés sur de multiples fichiers dans le livre.

C'est le flux de travail basique pour l'édition de livres – Ouvrir un fichier, faire des changements, prévisualiser et sauver.

[31] https://calibre-ebook.com/fr/demo#tutorials
[32] http://html.net/tutorials/html/
[33] http://html.net/tutorials/css/

Le reste de ce manuel traitera des divers outils et caractéristiques présents pour vous permettre d'exécuter efficacement des tâches spécifiques.

5.2 L'Explorateur de fichiers



Le *Navigateur de fichiers* vous donne un aperçu des différents fichiers à l'intérieur du livre que vous êtes occupé à éditer. Les fichiers sont arrangés par catégories, avec les fichiers texte (HTML) au sommet, suivis par les fichiers de feuilles de style (CSS), les images et ainsi de suite. Double cliquez simplement sur un fichier pour en commencer l'édition. L'édition est supportée pour le HTML, le CSS et les fichiers images. L'ordre des fichiers texte est le même ordre que celui dans lequel ils devraient être affichés, si vous êtes occupé de lire le livre. Tous les autres fichiers sont arrangés alphabétiquement.

En passant votre souris sur une entrée, vous pouvez voir sa taille, et aussi, au bas de l'écran, le chemin complet du fichier à l'intérieur du livre. Notez que les fichiers à l'intérieur des livres numériques sont compressés, aussi la taille du livre final n'est pas la somme des tailles des fichiers individuels.

Beaucoup de fichiers ont une signification spéciale, dans le livre. Ceux-ci auront généralement une icône à la suite de leurs noms, indiquant leur signification spéciale. Par exemple, dans l'image sur la gauche, vous pouvez voir que les fichiers *cover_image.jpg* et *titlepage.xhtml* ont l'icône d'une couverture à leur suite, ceci indique qu'ils sont l'image de couverture du livre et la page de titre. Similairement, le fichier *content.opf* à une icône métadonnée à sa suite, indiquant que les métadonnées du livre s'y trouvent et le fichier *toc.ncx* à une icône T à sa suite, indiquant que c'est la Table des Matières.

Vous pouvez effectuer beaucoup d'actions sur les fichiers individuels, en faisant un clic droit sur ceux-ci.

5.2.1 Renommer des fichiers

Vous pouvez renommer un fichier individuel en cliquant droit dessus et en sélectionnant *Renommer*. Renommer un fichier met automatiquement à jour les liens et les références à celui-ci tout au long du livre. Donc tout ce que vous avez à faire est de fournir un nouveau nom, calibre prendra soin du reste.

Vous pouvez également renommer beaucoup de fichiers en une fois. C'est utile si vous voulez que les fichiers aient un certain modèle de nom élémentaire. Par exemple, vous pourriez vouloir renommer tous les fichiers HTML pour avoir des noms Chapitre-1.html, Chapitre-2.html et ainsi de suite. Sélectionnez les fichiers que vous voulez renommer par lot en maintenant la touche `Maj` ou la touche `Ctrl` et en cliquant sur les fichiers. Puis cliquez droit et sélectionnez *Renommer par lot*. Entrez un préfixe et à quel nombre vous voudriez que la numérotation automatique commence, Cliquez OK et vous avez fini. La boîte de dialogue de renommage par lot vous permet également de renommer les fichiers selon l'ordre dans lequel ils apparaissent dans le livre au lieu de l'ordre dans lequel vous les avez sélectionnés, ce qui est utile, par exemple pour renommer toutes les images selon l'ordre dans lequel elles apparaissent.

Finalement, vous pouvez modifier l'extension de fichier par lot pour tous les fichiers sélectionnés. Sélectionnez plusieurs fichiers, comme ci-dessous, faites un clic droit et choisissez *Modifier l'extension de fichier pour les fichiers sélectionnés*.

5.2.2 Fusionner des fichiers

Parfois, vous pouvez vouloir fusionner deux fichiers HTML ou deux fichiers CSS ensemble. Il peut parfois être utile d'avoir tout dans un fichier unique. Soyez attentif, toutefois, que mettre beaucoup de contenu dans un fichier unique pose des problèmes de performance lorsque vous visualisez le livre dans un lecteur de livre numérique spécifique.

Pour fusionner de multiples fichiers ensemble, sélectionnez les en maintenant enfoncée la touche `Ctrl` (assurez vous de ne sélectionner des fichiers que d'un type, soit tous les fichiers HTML ou tous les fichers CSS et ainsi de suite). Faites alors un clic droit et sélectionner Fusionner. C'est tout. calibre fusionnera les fichiers, prendra automatiquement soin de faire la migration de tous les liens et références aux fichiers fusionnés. Notez que fusionner des fichiers peut parfois entraîner un changement du style de texte, puisque les fichiers individuels pourraient utiliser différentes feuilles de style.

Vous pouvez également sélectionner des fichiers texte, puis les glisser et les déposer sur un autre fichier texte pour fusionner les fichiers texte déposés dans le fichier texte cible.

5.2.3 Ordre de fichier texte changeant

Vous pouvez réorganiser l'ordre dans lequel les fichiers texte (HTML) sont ouverts lors de la lecture du livre en les faisant simplement glisser dans le *Explorateur de fichiers* ou en cliquant sur le fichier à déplacer puis en appuyant sur les modificateurs `Ctrl+Shift` avec les touches `Haut`, `Bas`, `Début` ou `Fin`. Pour les utilisateurs techniques, c'est ce qu'on appelle réordonner la structure organisationnelle du livre.

Notez que vous devez déposer les éléments *entre* d'autres éléments, et non sur eux, ce qui peut être un peu difficile jusqu'à ce que vous vous y habituiez. Si vous déposez un fichier au-dessus d'un autre, les fichiers seront fusionnés.

5.2.4 Marquage de la couverture

Les livres numériques ont normalement une image de couverture. Cette image est signifiée dans le *Navigateur de fichiers* par l'icône d'un livre brun suivant le nom de l'image. Si vous voulez désigner une autre image comme couverture, vous pouvez faire cela en faisant un clic droit sur le fichier et en choisissant *Marquer comme image de couverture*.

En complément, les fichiers EPUB ont le concept de « page de titre ». Une page de titre est un fichier HTML qui agit comme une page de titre/couverture pour le livre. Vous pouvez marquer un fichier HTML comme page de titre lors de l'édition des EPUBs en faisant un clic droit. Veillez que le fichier que vous marquez contient uniquement l'information de couverture. S'il contient une autre information, comme le premier chapitre, alors le contenu sera perdu si jamais l'utilisateur convertit l'EPUB dans calibre à un autre format. C'est parce que lors de la conversion, calibre suppose que la page de titre indiquée contient uniquement la couverture et pas d'autre contenu.

5.2.5 Suppression des fichiers

Vous pouvez supprimer des fichiers soit en faisant un clic droit sur ceux-ci, soit en appuyant sur la touche Supprimer. Supprimer un fichier retire toutes les références au fichier du fichier OPF, en enregistrant cette tâche. Cependant, les références dans les autres emplacements ne sont pas retirées, vous pouvez utiliser l'outil de Vérification de Livre pour facilement les trouver et les les supprimer/remplacer.

5.2.6 Exportation de fichiers

Vous pouvez exporter un fichier depuis l'intérieur du livre vers quelque part d'autre sur votre ordinateur. Ceci est utile si vous voulez travailler sur le fichier en isolation, avec des outils spécialisés. Pour faire ceci faire simplement un clic droit sur le fichier et choisissez *Exporter*.

Une fois que vous avez fini de travailler sur le fichier exporté, vous pouvez le ré-importer dans le livre, en faisant à nouveau un clic droit sur le fichier dans le livre et en choisissant *Remplacer avec le fichier…* qui vous permet de le fichier dans le livre avec le fichier précédemment exporté.

Vous pouvez également copier des fichiers entre plusieurs instances de l'éditeur. Sélectionnez les fichiers que vous voulez copier dans l'*Explorateur de fichiers*, puis cliquez droit et choisissez, *Copier le fichier sélectionné vers une autre instance de l'éditeur*. Ensuite, dans l'autre instance de l'éditeur, faites un clic droit dans l'*Explorateur de fichiers* et choisissez *Coller le fichier à partir d'une autre instance de l'éditeur*.

5.2.7 Ajout de nouvelles images/polices/etc. ou création de nouveaux fichiers vides

Vous pouvez ajouter une nouvelle image, police, feuille de style depuis votre ordinateur dans le livre en cliquant *Fichier → Nouveau fichier*. Ceci vous permet d'importer un fichier en cliquant sur le bouton *Importer un fichier de ressources* ou de créer un fichier HTML ou une feuille de style vierge en entrant simplement le nom du nouveau fichier.

Vous pouvez aussi importer de multiples fichiers dans le livre en une fois en utilisant Fichier->Importer des fichiers dans le livre.

5.2.8 Remplacement des fichiers

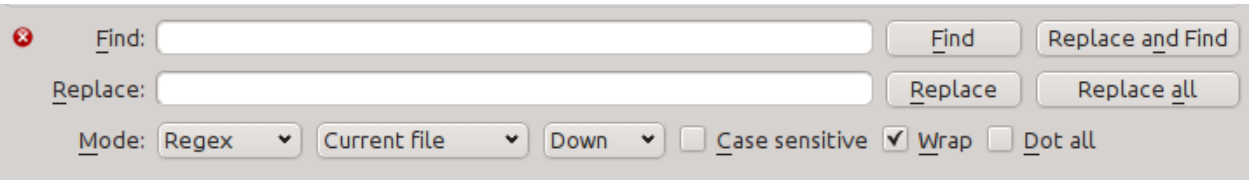
Vous pouvez facilement remplacer des fichiers existant dans le livre en faisant un clic droit sur le fichier et en choisissant remplacer. Ceci mettra automatiquement tous les liens et références à jour, dans le cas où le fichier de remplacement a un nom différent que le fichier en cours de remplacement.

5.2.9 Lier des feuilles de style aux fichiers HTML efficacement

Par commodité, vous pouvez sélectionner de multiples fichiers HTML dans le Navigateur de fichiers, faire un clic droit et choisir Lier les feuilles de style pour que calibre insère automatiquement les balises <link> pour ces feuilles de style dans tous les fichiers HTML sélectionnés.

5.3 Rechercher & remplacer

Éditer un livre possède une très puissante interface rechercher et remplacer qui vous permet de rechercher et remplacer du texte dans le fichier courant, à travers tous les fichiers et même dans une région marquée du fichier courant. Vous pouvez rechercher en utilisant une recherche normale ou en utilisant des expressions régulières. Pour apprendre comment utiliser des expressions régulières pour des recherches avancées, voir *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225).



Démarrer le Rechercher et Remplacer via l'entrée de menu *Rechercher → Rechercher/Remplacer* (vous devez être occupé à éditer un fichier HTML ou CSS).

Taper le texte que vous voulez trouver dans la case Chercher et sont remplacement dans la case Remplacer. Vous pouvez cliquer les boutons appropriés pour trouver la correspondance suivante, remplacer la correspondance actuelle et remplacer toutes les correspondances.

En utilisant les menus déroulants en bas de la boîte de dialogue, vous pouvez opérer la recherche sur le fichier courant, tous les fichiers textes, tous les fichiers de style ou tous les fichiers. Vous pouvez aussi choisir le mode de recherche à être une recherche normale (chaîne) ou une recherche par expression régulière.

Vous pouvez compter toutes les correspondances pour une expression de recherche via *Rechercher → Tout compter*. Le comptage s'effectuera sur n'importe quels fichiers/régions que vous avez sélectionnés dans les menus déroulants.

Vous pouvez aussi vous rendre à une ligne spécifique dans l'éditeur actuellement ouvert via *Rechercher → Aller à la ligne*.

Note : Rappelez-vous, pour vous armer de toute la puissance de la recherche et du remplacement, vous aurez besoin d'utiliser des expressions régulières. Voir : référence *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225).

5.3.1 Recherches sauvegardées

Vous pouvez sauvez les expressions rechercher/remplacer fréquemment utilisées ((inclus les expressions mode fonction) et les réutilisés de multiples fois. Pour sauver une recherche faites simplement un clic droit dans la case Recherche et sélectionner *Sauvegarder la recherche actuelle*.

Vous pouvez afficher les recherches sauvegardées via *Rechercher → Recherches Sauvegardées*. Ceci vous présentera une lise des expressions de recherche et de remplacement que vous pouvez appliquer. Vous pouvez même sélectionner des entrées multiples dans la liste en maintenant la touche `Ctrl` enfoncée pendant que vous cliquez pour exécuter de multiples expressions de recherche et remplacement en une seule opération.

5.3.2 Mode fonction

Le mode fonction vous permet d'écrire des fonctions Python arbitrairement puissantes qui sont exécutées sur chaque Rechercher/remplacer. Vous pouvez faire à peu près n'importe quelle manipulation de texte que vous souhaitez en mode fonction. Pour plus d'information, voir *Mode fonction pour Rechercher & remplacer dans l'Éditeur* (page 103).

5.3.3 Recherche ignorant les balises HTML

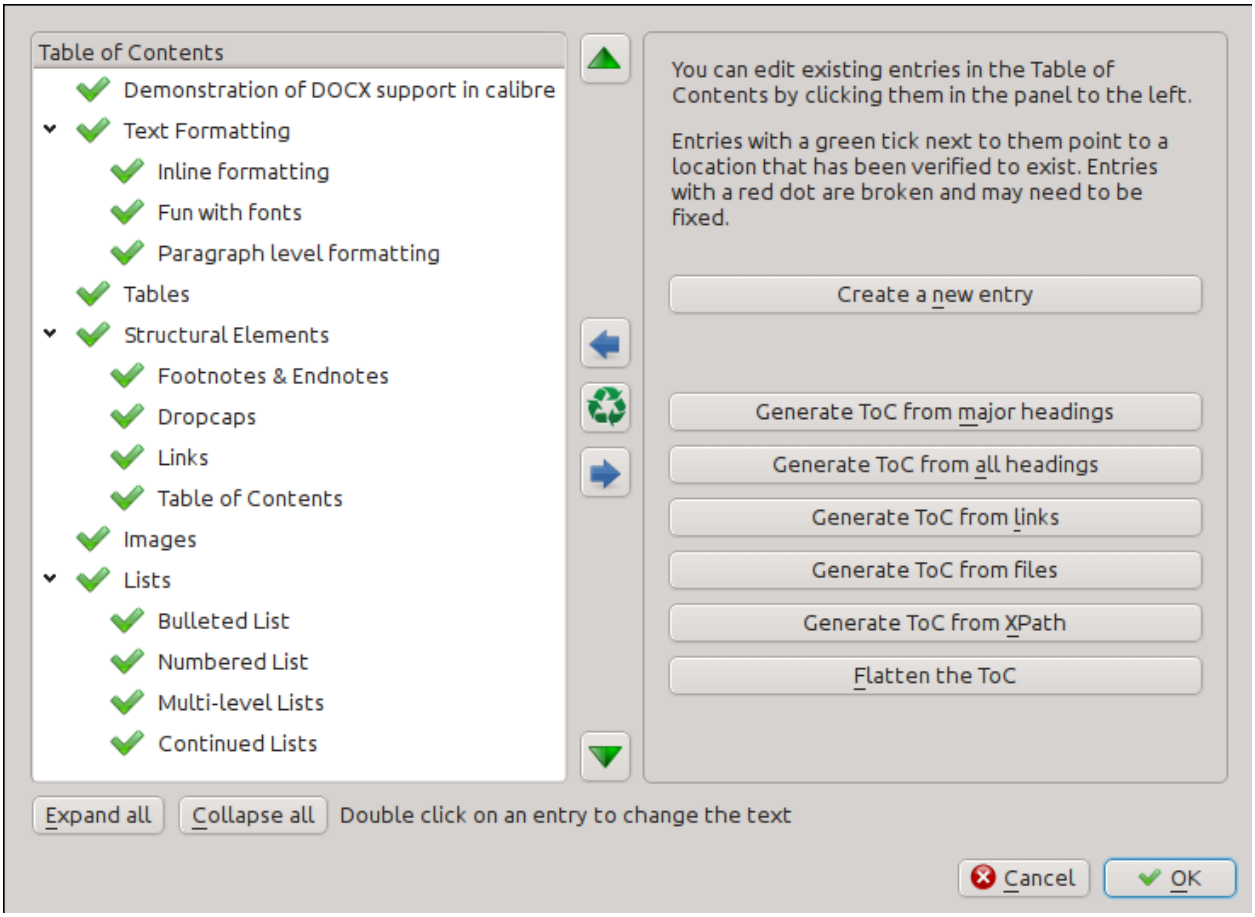
Il y a aussi un outil dédié pour la recherche de texte, ignorant toutes les balises HTML s'y trouvant. Par exemple, si le livre contient l'HTML `Accent sur un <i>mot</i>.` vous pouvez chercher `sur un mot` et il sera trouvé même s'il y a une balise `<i>` au milieu. Utilisez cet outil via l'élément du menu *Rechercher → Rechercher en ignorant le balisage HTML*

5.4 Outils automatisés

Editer le livre à divers outils pour aider avec les tâches communes. Ceux-ci sont accessibles via le menu *Outils*.

5.4.1 Edition de la Table des Matières

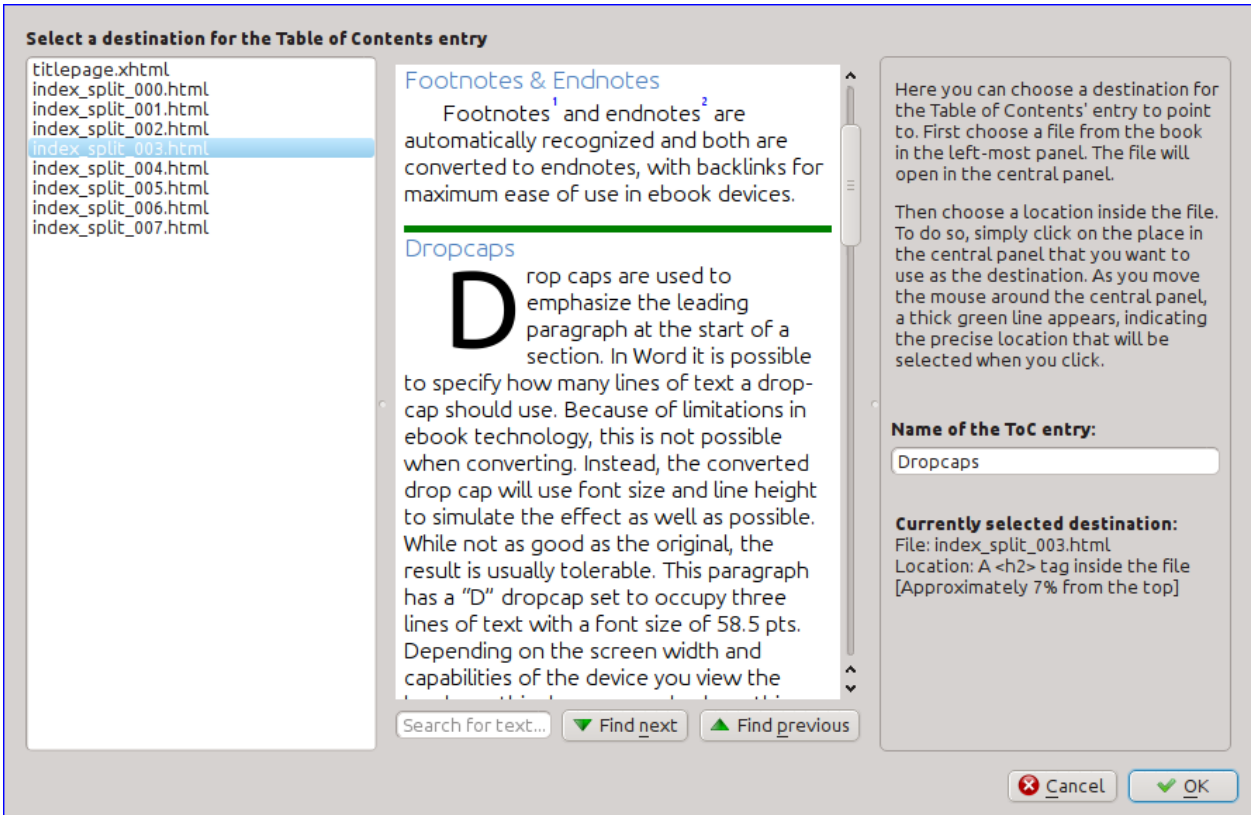
Il y a un outil dédié pour l'édition facile de la Table des Matières. Lancez le avec *Outils → Table des Matières → Modifier Table des Matières*.



L'outil éditer la Table des Matières vous affiche la Table des Matières actuelle (s'il y en a une) sur la gauche. Double cliquez simplement sur n'importe quelle entrée pour changer son texte. Vous pouvez aussi ré-organiser les entrées par glisser et déposer ou en utilisant les boutons sur la droite.

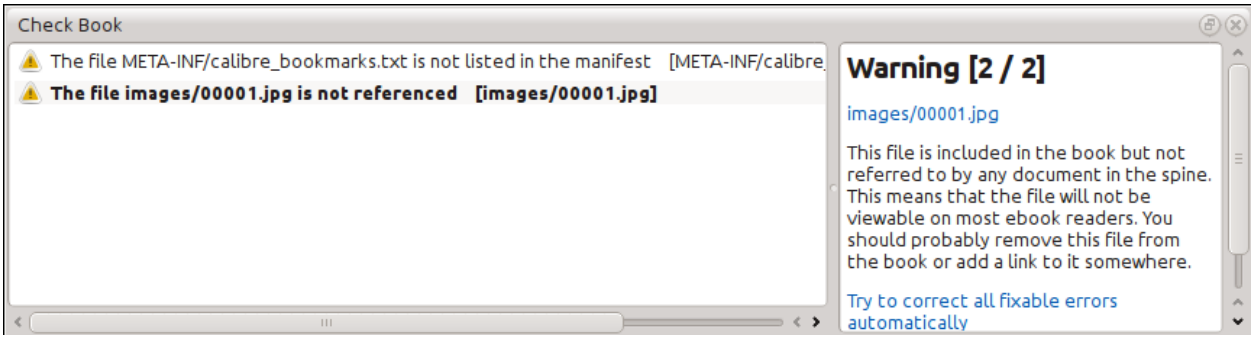
Pour les livres qui n'ont pas une Table des Matières préexistante. l'outil vous donne diverses options pour auto-générer une Table des Matières à partir du texte. Vous pouvez générer à partir des en-têtes dans le document, de liens, de fichiers individuels et ainsi de suite.

Vous pouvez éditer des entrées individuelles en cliquant dessus et en cliquant le bouton *Changer l'emplacement vers lequel pointe cette entrée*. Ceci ouvrira une mini prévisualisation du livre, déplacer simplement le curseur de souris sur le panneau de visualisation du livre, et cliquez où vous voulez que l'entrée pointe. Une ligne verte épaisse vous affichera l'emplacement. Cliquez OK une fois que vous êtes content avec l'emplacement.



5.4.2 Vérification du livre

L'outil *Vérifier le livre* recherche dans votre livre les problèmes qui pourraient l'empêcher de fonctionner comme prévu sur les périphériques de lecture actuels. Activez le via *Outils → Vérifier le livre*.



Tout problème trouvé est reporté dans une liste agréable, facile à utiliser. Cliquer n'importe quelle entrée dans la liste vous affiche une certaine aide à propos de cette erreur aussi bien que de vous donner l'option d'auto-réparer cette erreur, si cette erreur peut être réparée automatiquement. Vous pouvez aussi double cliquer sur l'erreur pour ouvrir l'emplacement de celle-ci dans un éditeur, vous pouvez ainsi la réparer vous même.

Certains des contrôles exécutés sont :

- Balise HTML mal formée. Toute balise HTML qui n'a pas été analysée comme XML bien formé est signalé. La corriger vous assurera que votre balise fonctionne comme attendu dans tous les contextes. calibre peut aussi auto réparé ces erreurs, mais l'auto réparation peut avoir parfois des effets inattendus, utilisez donc avec précaution. Comme toujours, un point de contrôle est créé avant l'auto réparation vous pouvez donc annuler tous

les changements. L'auto réparation fonctionne en analysant la balise utilisant l'algorithme HTML5, qui est fortement sensible aux défaillances et convertit alors en XML bien formé.

- Styles CSS mal formés ou inconnus. Tout CSS qui n'est pas valide ou donrt les propriétés ne sont pas définies dans le standard CSS 2.1 (plus un peu de CSS 2.3) sont signalées. Le CSS est vérifié dans toutes les feuilles de style, attributs de style intégrés et les balises <style> dans les fichiers HTML.
- Les liens cassés. Les liens qui pointent vers des fichiers à l'intérieur du livre qui sont manquant sont signalés.
- Fichiers non référencés. Fichiers dans le livre qui ne sont pas référencés par tout autre fichier ou ne sont pas dans la structure organisationnelle sont signalés.
- Divers problèmes habituels dans les fichiers OPF tels que la structure organisationnelle ou des éléments du manifeste en double, des idrefs ou des étiquettes meta couverture cassées, des sections requises manquantes et ainsi de suite.
- Diverses vérifications de compatibilité pour les problèmes connus qui entraînent un mauvais fonctionnement du livre sur des périphériques de lecture.

5.4.3 Ajout d'une couverture

Vous pouvez facilement ajouter une couverture au livre via *Outils → Ajouter une couverture*. Ceci vous permet soit de choisir une image existante dans le livre comme couverture ou d'importer une nouvelle image dans le livre et en faire la couverture. Lors de l'édition de fichiers EPUB, l'examen de l'HTML pour la couverture est automatiquement généré. Si une couverture existante pour le livre est trouvée, elle est remplacée. L'outil prend automatiquement soin de marquer les fichiers couverture comme couvertures dans l'OPF.

5.4.4 Intégration des polices référencées

Accédé via *Outils → Inclure les polices référencées*, cet outil trouve toutes les polices référencées dans le livre et si elles ne sont pas encore inclusent, les recherche sur votre ordinateur et les inclus dans le livre, s'il les trouve. Assurez-vous s'il vous plaît que vous avez les droits de copie nécessaires pour inclure les polices à licence commerciale, avant de faire ceci.

5.4.5 Créer un sous-ensemble de polices

Accédé via *Outils → Sous-ensemble de polices incluses*, cet outil réduit toutes les polices dans le livre pour contenir uniquement des glyphes pour le texte actuellement présent dans le livre. Ceci réduit habituellement la taille des fichiers de police de ~ 50%. Cependant, soyez conscient qu'une fois les polices sont en sous-ensemble, si vous ajouter du nouveau texte dont les caractères ne sont pas précédemment présents dans le sous-ensemble de police, la police ne fonctionnera pour le nouveau texte. Aussi faites ceci uniquement comme dernière étape de votre plan de travail.

5.4.6 Amélioration de la ponctuation

Converti les tirets, points de suspension, guillemets, multples traits d'union, etc. en texte clair en leurs équivalents typographiquement corrects. Notez que l'algorithme peut parfois générer des résultats incorrects, particulièrement quand des apostrophes au début des contractions sont impliquées. Accédé via ;guillabel :*Outils->Ponctuation intelligente*.

5.4.7 Transformation des propriétés CSS

Créer des règles pour transformer l'esthétique du livre. Par exemple, créer une règle pour convertir tout le texte rouge en vert ou doubler la taille de police de tout le texte dans le livre ou faire le texte d'une certaine famille de police italique, etc.

La création de règles est simple, les règles suivent un format de langage naturel, cela ressemble à :

- Si la propriété *color* est *rouge changez* la en *vert*
- Si la propriété *font-size* est *n'importe quelle valeur multiplier* la valeur par *2*

Accessible via *Outils → Transformer les styles*.

5.4.8 Suppression en cours des règles CSS inutilisées

Supprimer toutes les règles CSS inutilisées des feuilles de style et des balises <style>. Certains livres créés à partir de modèles de production peuvent avoir un grand nombre de règles CSS supplémentaires qui ne correspondent à aucun contenu actuel. Ces règles supplémentaires peuvent ralentir les lecteurs qui ont besoin de toutes les tracer. Accédez via *Outils → Supprimer les règles CSS inutilisées*.

5.4.9 Correction de l'HTML

Cet outil convertit simplement de l'HTML qui ne peut pas être analysé comme du XML en XML bien formé. Il est très courant dans les livres numériques d'avoir du XML mal formé, cette outil automatise donc simplement le processus de réparation d'un tel HTML. L'outil fonctionne en analysant l'HTML en utilisant l'algorithme de l'HTML5 (l'algorithme utilisé dans tous les navigateurs récents) et en convertissant alors le résultat en XML. Soyez conscient que l'auto réparation peut parfois avoir des résultats inattendus. Si vous préférez, vous pouvez utiliser l'outil de vérification de livre numérique présenté plus haut pour trouver et corriger manuellement les problèmes dans l'HTML. On y accède via *Outils → Réparer l'HTML*.

5.4.10 Embellissement des fichiers

Cet outil est utilisé pour auto formater les fichiers HTML et CSS afin qu'ils aient une « belle apparence ». Le code est auto indenté afin qu'il s'aligne joliment, les interlignes sont insérés là où c'est approprié et ainsi de suite. Notez qu'embellir auto répare également l'HTML/CSS abîmé. Dès lors, si vous ne voulez qu'aucune auto réparation ne soit effectuée, utiliser d'abord l'outil Vérifier le livre pour corriger tous les problèmes et seulement alors exécuter embellir. Accédez via *Outils → Embellir tous les fichiers*.

Note : En HTML tout texte peut avoir un espacement significatif, par l'intermédiaire de la directive CSS white-space. Par conséquent, l'embellissement pourrait potentiellement changer le rendu de l'HTML. Pour éviter cela autant que possible, l'algorithme d'embellissement embellit uniquement des balises de bloc de niveau qui contiennent d'autres balises de bloc de niveau.. Ainsi, par exemple, du texte à l'intérieur d'une balise <p> n'aura pas ses espacements changés. Mais une balise <body> qui contient seulement d'autres balises <p> et sera embellie. Ceci peut parfois indiquer qu'un fichier particulier ne sera pas affecté par l'embellissement car il n'a pas les balises de niveau de bloc appropriées. Dans de tels cas vous pouvez essayer différents outils d'embellissement, qui font moins attention, par exemple : HTML Tidy[34].

---

[34] https://infohound.net/tidy/

5.4.11 Insertion d'une Table des Matières

Normalement dans les livres numériques, la Table des Matières est séparée du texte principal et est typiquement accédée par l'intermédiaire d'un bouton/menu spécial dans le périphérique de lecture de livre numérique. Vous pouvez également voir calibre générer automatiquement une Table des Matières « en interne » qui devient une partie du texte du livre. Elle est générée sur base de la Table des Matières actuellement définie.

Si vous utilisez cet outil de multiples fois, chaque invocation entraînera le remplacement de la Table des Matières précédemment créée. L'outil peut être accédé via *Outils → Table des Matières → Insérer une Table des Matières*.

5.4.12 Configuration de Sémantique

Cet outil est utilisé pour paramétrer les « sémantiques » dans les fichiers EPUB. Les sémantiques sont simplement, des liens dans le fichier OPF qui identifient certains emplacements dans le livre comme ayant une signification spéciale. Vous pouvez les utiliser pour identifier l'avant-propos, la dédicace, la couverture, la table des matières, etc. Choisissez simplement le type d'information sémantique que vous voulez spécifier et choisissez dans le livre l'emplacement vers lequel le lien devra pointer. Cet outil peut être accédé via *Outils → Paramétrer sémantique*.

5.4.13 Filtrage des informations de style

Cet outil peut être utilisé pour facilement supprimer des propriétés de style CSS spécifiées de l'entièreté du livre. Vous pouvez lui indiquer quelles propriétés vous voulez supprimer, par exemple, `color, background-color, line-height` et il les supprimera partout où elles surviennent — feuilles de style, étiquettes <style> et attributs internes « style ». Après la suppression les informations de style, un sommaire des tous les changements effectués est affiché ainsi vous pouvez voir exactement ce qui a été changé. L'outil peut être accédé via *Outils → Filtrer l'information de style*.

5.4.14 Mise à jour des éléments du livre

Cet outil peut être utilisé pour mettre à jour les éléments internes du livre, si possible. Par exemple, il mettra à jour les livres EPUB 2 en livres EPUB 3. L'outil est accessible via *Mise à jour des éléments du livre*.

5.5 Points de Contrôle

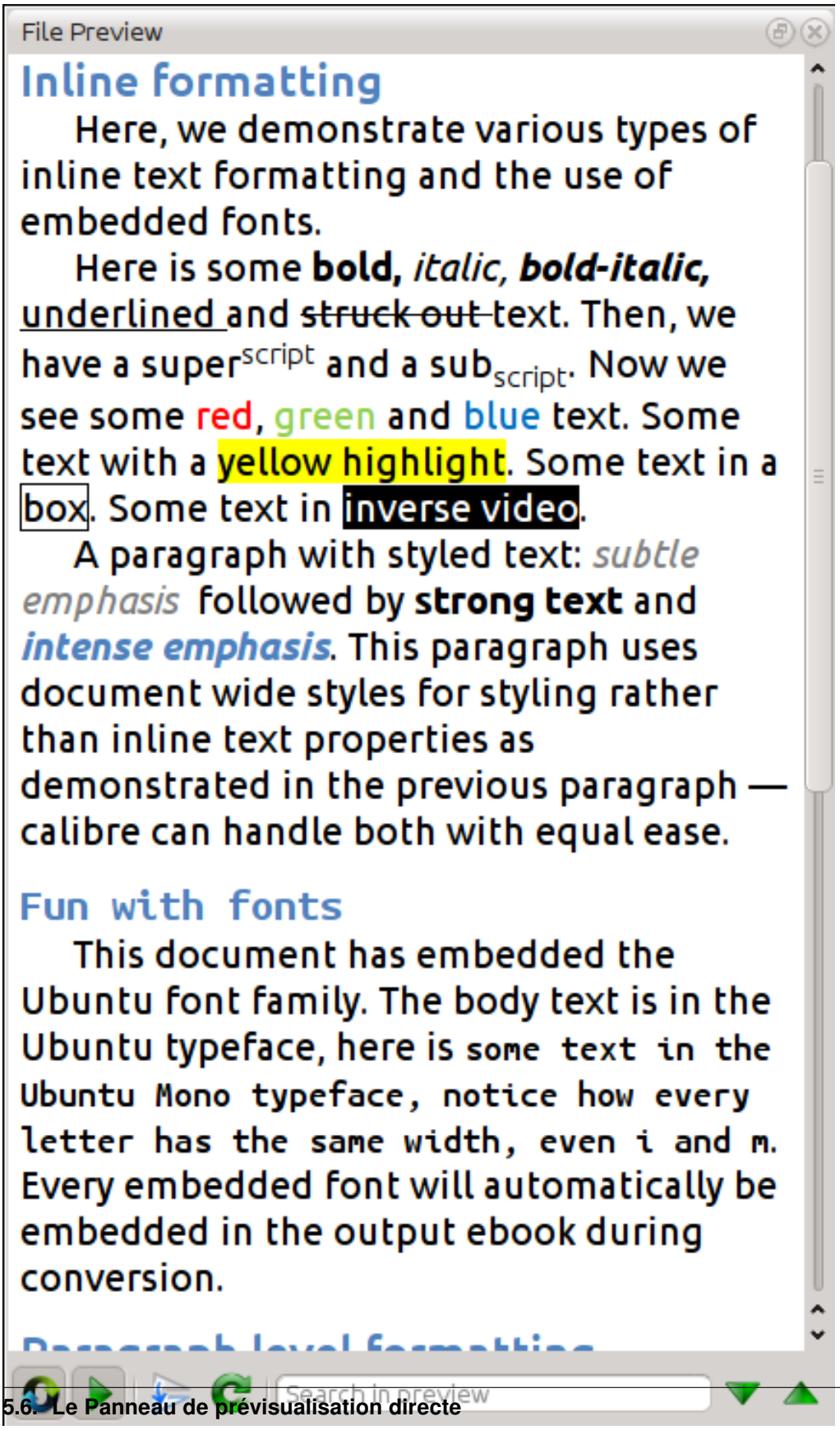
Points de contrôle est un moyen d'indiquer l'état actuel du livre comme étant « spécial ». Vous pouvez alors y aller et faire tous les changements que vous voulez au livre et si vous n'appréciez pas les résultats, revenir au point de contrôle. Les points de contrôle sont créés automatiquement chaque fois que vous exécutez n'importe lequel des outils automatisés décrit dans la section précédente.

Vous pouvez créer un point de contrôle via guilabel :*Editer->Créer un point de contrôle*. Et revenir à un point de contrôle précédent avec *Editer → Rétablir …*

La fonctionnalité point de contrôle est en addition au mécanisme normal Annuler/Refaire lors de l'édition de fichiers individuels. Les points de contrôles sont nécessaires lorsque les changements sont éparpillés sur de multiples fichiers dans le livre ou quand vous souhaitez être en possibilité d'annuler dans leur ensemble un grand nombre de changements qui s'y rapporte.

Vous pouvez voir une liste des points de contrôle disponibles via *Aperçu → Points de contrôle*. Vous pouvez comparer l'état actuel du livre à un point de contrôle particulier en utilisant l'outil *Comparer des livres numériques* (page 125) – en sélectionnant le point de contrôle intéressé et en cliquant le bouton *Comparer*. Le bouton *Revert to* restaure le livre au point de contrôle sélectionné, annulant tous les changements depuis que ce point de contrôle a été créé.

5.6 Le Panneau de prévisualisation directe



La *Prévisualisation de fichier* vous donne un aperçu des différents fichiers intégrés. Le panneau de prévisualisation en direct vous montre les changements que vous êtes occupé à faire (avec deux ou trois secondes de délai). Au fur et à mesure que vous éditez les fichiers HTML et CSS, le panneau de prévisualisation est automatiquement mis à jour pour refléter vos changements. En même temps que vous bougez le curseur sur l'éditeur, le panneau de prévisualisation traquera son emplacement, vous montrant l'emplacement correspondant dans le livre. En cliquant dans le panneau de prévisualisation, cela entraînera le positionnement du curseur dans l'éditeur sur l'élément que vous avez cliqué. Si vous cliquez un lien pointant vers un autre fichier dans le livre, ce fichier sera ouvert dans l'éditeur et le panneau de prévisualisation, automatiquement.

Vous pouvez désactiver la synchronisation automatique entre l'emplacement et la prévisualisation en direct des changements – par les boutons se situant sous le panneau de prévisualisation. La mise à jour en direct du panneau de configuration a lieu uniquement quand vous n'êtes pas activement occupé à taper dans l'éditeur, ceci afin de ne pas vous distraire ou vous ralentir, attendre pour que la prévisualisation soit fournie.

Le panneau de prévisualisation vous montre à quoi ressemblera le texte quand il est visionné. Cependant, le panneau de prévisualisation n'est pas un substitut pour tester votre livre sur un périphérique de lecture. Il est à la fois plus et moins capable qu'un lecteur. Il tolérera des erreurs et des balises négligées bien mieux que beaucoup de périphériques de lecture. Il ne vous affichera pas non plus les marges de page, les sauts de page et les polices intégrées qui utilisent des alias de nom de police. Utiliser le panneau de prévisualisation pendant que vous travaillez sur le livre, mais une fois que vous avez fini, révisez-le dans un périphérique de lecture ou dans un émulateur.

Note : Le panneau de prévisualisation ne supporte pas les fontes intégrées si le nom de la police dans le fichier de police ne correspond pas au nom dans la règle CSS @font-face. Vous pouvez utiliser l'outil Vérifier le livre pour trouver rapidement et réparer tous types de problèmes de police.

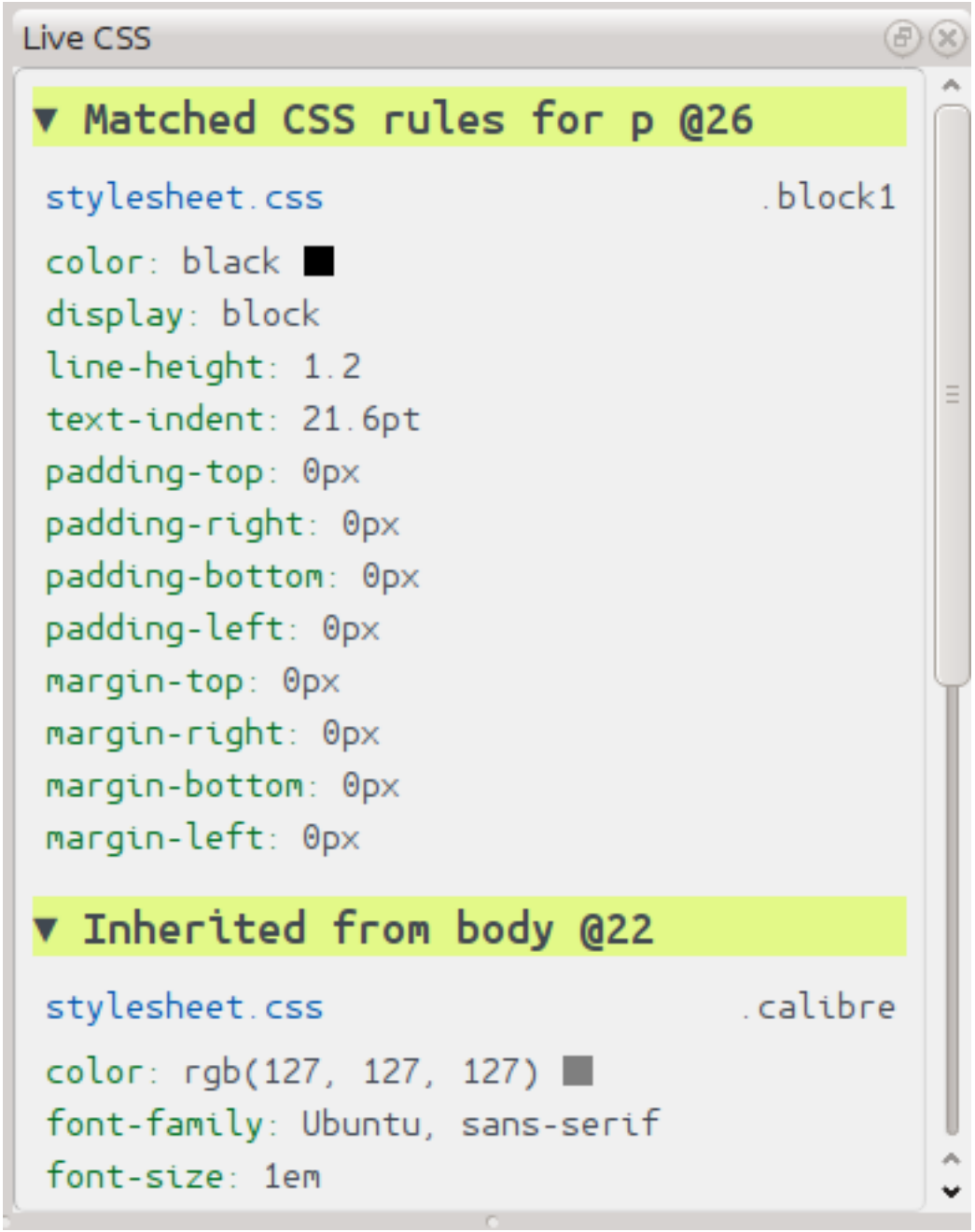
5.6.1 Fractionnement des fichiers HTML

Une utilisation, peut-être non évidente, du panneau de prévisualisation est de fractionner de longs fichiers HTML. Pendant la visualisation du fichier que vous voulez fractionner, cliquez le bouton *Mode fractionné* en dessous du panneau de prévisualisation . Bouger alors simplement votre souris jusqu'à l'endroit où vous voulez fractionner le livre et cliquez. Une épaisse ligne verte vous montrera ou exactement le fractionnement s'effectuera au fur et à mesure que vous bougez votre souris. Une fois que vous avez trouvé l'emplacement que vous voulez, cliquez simplement et le fractionnement sera effectué.

Fractionner le fichier mettra automatiquement à jour tous les liens et les références qui pointent dans la moitié inférieure du fichier et ouvrira le nouveau fichier fractionné dans un éditeur.

Vous pouvez aussi fractionner un fichier HTML simple en de multiples emplacements automatiquement, en faisant un clic droit à l'intérieur du fichier dans l'éditeur et en choisissant *Fractionner en de multiples emplacements*. Ceci vous permettra de fractionner facilement un grand fichier à toutes les balises de titre ou à toutes les balises ayant une certaine classe et ainsi de suite.

5.7 Le panneau CCS en temps réel



Le panneau *CSS en temps réel* vous affiche toutes les règles de style qui s'appliquent à la balise que vous êtes occupé à éditer. Le nom de la balise, avec son numéro de ligne dans l'éditeur est montré, suivi par une liste des règles de style correspondantes.

C'est une excellente manière de voir rapidement quelles règles de style s'appliquent à n'importe quelle balise. La vue à aussi des liens cliquables (en bleu), qui vous amène directement à l'emplacement où le style a été défini, dans le cas où vous souhaitez faire n'importe quel changement à la règle de style. Les règles de style qui s'appliquent directement à la balise, aussi bien que les règles qui sont héritées de balises parents sont montrées.

Le panneau vous montre aussi quels sont les styles finalement calculés pour la balise. Les propriétés dans la liste qui sont

supplantées par des règles de plus haute priorité sont montrées avec une ligne au travers de celles-ci.

Vous pouvez activer le panneau CSS en temps réel via *Aperçu → CSS en temps réel*.

5.8 Outils divers

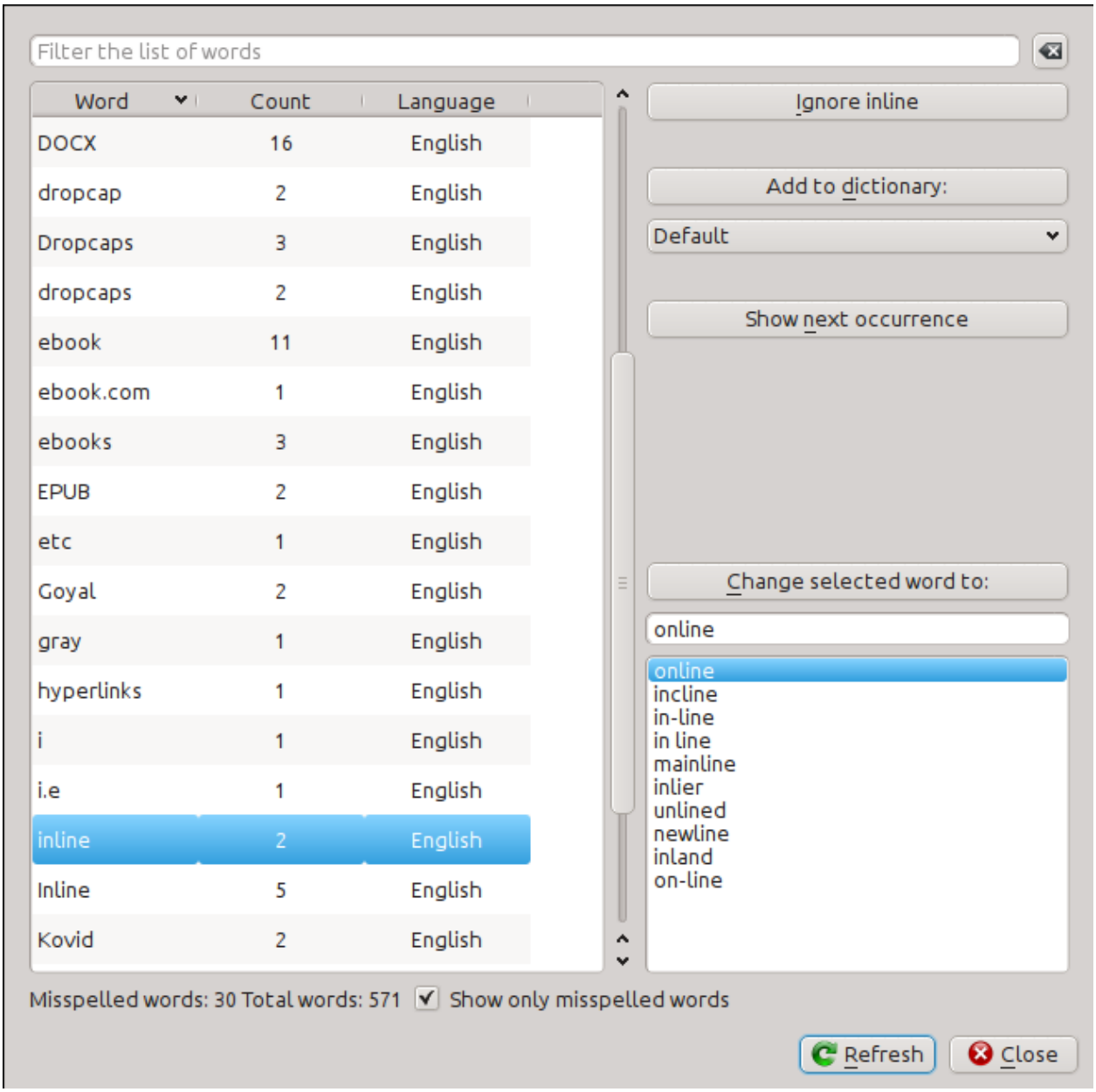
Il y a quelques outils de plus qui peuvent être utiles quand vous éditez le livre.

5.8.1 La vue Table des Matières

La vue Table des Matières vous montre la table des matières actuelle dans le livre. Faire un double clic sur n'importe quelle entrée ouvre l'endroit où cette entrée pointe dans un éditeur. Vous pouvez faire un clic droit pour éditer la Table des Matières, rafraîchir la vue ou développer/réduire tous les éléments. Accédez à cette vue via *Aperçu → Table des Matières*.

5.8.2 Vérification de l'orthographe des mots dans le livre

Vous pouvez exécuter une vérificateur orthographique via *Outils → Vérifier l'orthographe*.



Les mots sont affichés avec le nombre de fois où ils apparaissent dans le livre et la langue à laquelle le mot appartient. L'information de langue est récupérée des métadonnées des livres et des attributs `lang` dans les fichiers HTML. Ceci permet au vérificateur d'orthographe de bien fonctionner même avec des livres qui contiennent du texte dans différentes langues. Par exemple, dans l'extrait HTML qui suit, le mot color sera vérifié en utilisant American English et le mot colour en utilisant British English :

```
<div lang="en_US">color <span lang="en_GB">colour</span></div>
```

Note : Vous pouvez faire double cliquer un mot pour mettre en évidence l'occurrence suivante de ce mot dans l'éditeur. Ceci est utile si vous souhaitez éditer manuellement le mot, ou voir dans quel contexte il se situe.

Pour changer un mot, double cliquez simplement l'une des alternatives suggérée sur la droite, ou entrer votre propre orthographe corrigée et cliquez le bouton *Changer le mot sélectionné par*. Ceci remplacera toutes les occurrences du mot

dans le livre. Vous pouvez aussi faire un clic droit sur un mot dans la liste de mots principale pour changer le mot aisément à partir du menu contextuel.

Vous pouvez faire en sorte que le vérificateur d'orthographe ignore un moment pour la session courante en cliquant le bouton *Ignorer*. Vous pouvez aussi ajouter un mot au dictionnaire utilisateur en cliquant le bouton *Ajouter au dictionnaire*. Le vérificateur d'orthographe supporte de multiples dictionnaires utilisateur, vous pouvez donc sélectionner le dictionnaire auquel vous voulez ajouter le mot.

Vous pouvez aussi faire afficher par le vérificateur d'orthographe tous les mots dans votre livre, pas uniquement ceux qui sont mal orthographiés. Ceci est utile pour voir quels mots sont les plus communs dans votre livre et effectuer un simple rechercher et remplacer sur des mots individuels.

Note : Si vous effectuez n'importe quel changement au livre en l'éditant pendant la vérification orthographique, vous devrez cliquer le bouton *Actualiser* dans l'outil de vérification d'orthographe. Si vous ne faites pas cela et continuez d'utiliser l'outil de Vérification d'orthographe, vous pourriez perdre les changements que vous avez effectués dans l'éditeur.

Note : Pour exclure un fichier individuel de la vérification orthographique lors de l'exécution de l'outil de vérification orthographique, vous pouvez utiliser le bouton *Exclure les fichiers* ou ajouter le commentaire suivant juste sous la balise d'ouverture du fichier

```
<!-- calibre-no-spell-check -->
```

Ajouter de nouveaux dictionnaires

Le vérificateur d'orthographe est fourni avec des dictionnaires intégrés pour les langues Anglaises et Espagnoles. Vous pouvez installer vos propres dictionnaires via *Editer → Préférences → Propriètés de l'éditeur → Organiser les dictionnaires orthographiques*. Le vérificateur orthographique utiliser les dictionnnaires diu programme LibreOffice (au format .oxt). Vous pouvez télécharger ces dictionnaires depuis le dépôt d'Extensions LibreOffice <https://extensions.libreoffice.org/?Tags%5B%5D=50>`_.

5.8.3 Insérer des caractères spéciaux

Vous pouvez insérer des caractères qui sont difficiles à taper en utilisant l'outil *Editer → Insérer un caractère spécial*. Ceci vous montre tous les caractères Unicode, cliquez simplement le caractère que vous voulez taper. Si vous maintenez la touche `Ctrl` pendant que vous cliquez, la fenêtre se fermera d'elle-même après l'insertion du caractère sélectionné. Cet outil peut être utiliser pour insérer des caractères spéciaux dans le texte principal ou dans n'importe quel espace de l'interface utilisateur, comme l'outil Rechercher et remplacer.

Parce qu'il y a beaucoup de caractères, vous pouvez définir vos propres caractères *Préférés*, qui seront montrés en premier. Faites simplement un clic droit sur un caractère pour le marquer comme favori. Vous pouvez aussi faire un clic droit sur un caractère dans préférés pour l'en retirer. Finalement, vous pouvez réarranger l'ordre des caractères dans préférés en cliquant le bouton *Réarranger les préférés* et alors glisser et déposer les caractères dans les favoris.

Vous pouvez aussi saisir directement des caractères spéciaux en utilisant le clavier. Pour faire cela, vous saisissez le code Unicode pour le caractère (en hexadécimal) et appuyez alors les touches `Alt+X` ce qui convertira le code précédemment saisi dans le caractère correspondant. Par exemple, pour saisir ÿ vous devrez saisir ff et ensuite :kbd :Alt+X. Pour saisir un espace insécable vous devrez utiliser a0 et ensuite `Alt+X`, pour saisir des points de suspension vous devrez utiliser 2026 et ensuite `Alt+X` et ainsi de suite.

Finalement, vous pouvez saisir des caractères spéciaux en utilisant les entités nommées HTML. Par exemple, saisir   ; sera remplacé par un espace insécable lorsque vous saisissez le point-virgule. Le remplacement survient uniquement quand vous saisissez le point-virgule.

5.8.4 La vue inspecteur de code

Cette vue vous montre le codage HTML et CSS qui s'applique à l'élément actuel d'intérêt. Vous l'ouvrez par un clic droit sur un emplacement dans le panneau de prévisualisation et en choisissant *Inspecter*. Il vous permet de voir le codage HTML pour cet élément et d'une manière plus primordiale, les styles CSS qui s'y appliquent. Vous pouvez même dynamiquement éditer les styles et voir immédiatement quels effets donnent vos changements. Notez que l'édition des styles n'apporte pas réellement des modifications au contenu de livre, il permet seulement de l'expérimentation rapide. La capacité d'éditer en direct à l'intérieur de l'Inspecteur est en cours de développement.

5.8.5 Vérifications des liens externes

Vous pouvez utiliser cet outil pour vérifier tous les liens dans votre livre qui pointent vers des sites web externes. L'outil essayera de visiter chaque site web lié à l'extérieur, et si la visite échoue, il vous rapportera tous les liens cassés dans un format pratique pour les réparer.

5.8.6 Téléchargement de ressources externes

Vous pouvez utiliser cet outil pour télécharger automatiquement n'importe quel images/feuilles de style/etc. dans le livre qui ne sont pas fournies avec le livre (Par ex : il a des URL pointant vers un endroit sur Internet). L'outil retrouvera tout ces types de ressources et les téléchargera automatiquement, les ajoutera au livre et remplacera toutes les références à celles-ci pour utiliser les fichiers téléchargés.

5.8.7 Organisation des fichiers par type dans des dossiers

Souvent en éditant les fichiers EPUB que vous obtenez de quelque part, vous constaterez que les fichiers à l'intérieur de l'EPUB sont arrangés au petit bonheur la chance, dans différents sous-dossiers. Cet outil vous permet d'entrer automatiquement tous les fichiers dans des sous-dossiers basés sur leurs types. Accédez-y par l'intermédiaire de *Outils → Organiser dans des dossiers*. Notez que cet outil change seulement la manière dont les fichiers sont arrangés à l'intérieur de l'EPUB, il ne change pas la manière dont ils sont affichés dans le Navigateur de fichiers.

5.8.8 Importation de fichiers dans d'autres formats de livres numériques comme l'EPUB

L'éditeur inclut la capacité d'importer des fichiers de certains autres formats de livre numérique directement comme un nouvel EPUB, sans passer par une pleine conversion. C'est particulièrement utile pour créer directement des fichiers EPUB à partir de vos propres fichiers HTML édités à la main. Vous pouvez faire ceci par l'intermédiaire de *Fichier → Importer un fichier HTML ou DOCX en tant que nouveau livre*.

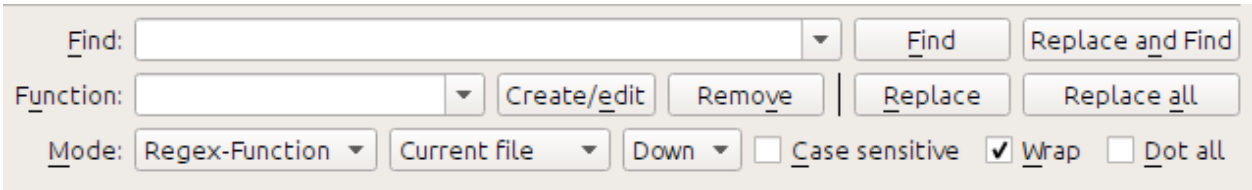
Mode fonction pour Rechercher & remplacer dans l'Éditeur

L'outil *Rechercher & remplacer* dans l'éditeur supporte un *mode fonction*. Dans ce mode, vous pouvez combiner des expressions régulières (voir *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225)) avec de puissantes fonctions Python pour faire toutes sortes de traitements avancés de texte.

Dans le mode standard *regexp* pour rechercher et remplacer, vous spécifiez tant une expression régulière à rechercher qu'un modèle qui est utilisé pour remplacer toutes les correspondances trouvées. Dans le mode fonction, à la place d'utiliser un modèle fixe, vous spécifiez une fonction quelconque, dans le langage de programmation Python[35]. Ceci vous permet de faire beaucoup de choses qui ne sont pas possibles avec de simples modèles.

[35] https://docs.python.org

Les techniques pour utiliser le mode fonction et la syntaxe seront décrites à l'aide d'exemples, vous montrant comment créer des fonctions pour effectuer progressivement des tâches plus complexes.



Réparer automatiquement la casse des titres dans le document

Ici , nous utiliserons l'une des fonctions intégrée dans l'éditeur pour changer automatiquement la casse de tous le texte à l'intérieur des balises titre en casse titre

```
Find expression: <([Hh][1-6])[^>]*>.+?</\1>
```

Pour la fonction, choisissez simplement la fonction intégrée *Texte en casse de titre(ignorer les balises)*. Celle-ci changera les titres qui ressemblent à : `<h1>certains TITRES</h1>` en `<h1>Certains Titres</h1>`. Elle fonctionnera même s'il y a d'autres balises HTML dans les balises de titre.

Votre première fonction personnalisée - les traits d'union d'embellissement

La vraie puissance du mode fonction vient du fait de pouvoir créer vos propres fonctions pour traiter le texte de manières quelconques. L'outil de Ponctuation Intelligente dans l'éditeur laisse les traits d'union individuels de côté, vous pouvez employer cette fonction pour les remplacer par des tirets fins.

Pour créer une nouvelle fonction, cliquez simplement le bouton *Créer/Éditer* pour créer une nouvelle fonction et copiez le code Python qui se trouve ci-dessous.

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    return match.group().replace('--', '—').replace('-', '—')
```

Chaque fonction personnalisée *Rechercher & remplacer* doit avoir un nom unique et consister en une fonction Python nommée replace, qui accepte tous les arguments affichés ci-dessus. Pour le moment, nous ne nous inquiéterons pas à propos de tous les différents arguments de la fonction `replace()`. Focalisons-nous uniquement sur l'argument `match`. Il représente une correspondance lors de l'exécution d'un rechercher et remplacer. Sa documentation complète est disponible ici[36]. `match.group()` renvoie simplement tout le texte correspondant et tout ce que nous faisons est de remplacer les traits d'union dans ce texte par des tirets fins, en remplaçant d'abord les doubles traits d'union et ensuite les traits d'union seuls.

Utilisez cette fonction avec l'expression régulière de recherche

```
>[^<>]+<
```

Et il remplacera tous les traits d'union par des tirets fins, mais seulement dans le texte actuel et non dans les définitions de balises HTML.

[36] https://docs.python.org/library/re.html#match-objects

La force du mode fonction - utiliser un dictionnaire orthographique pour réparer les mots aux mauvais traits d'union

Souvent les livres numériques créés à partir de scans de livres imprimés contiennent des mots avec de mauvais traits d'union – les mots qui sont divisés à la fin de la ligne de la page imprimée. Nous écrirons une fonction simple pour trouver et réparer automatiquement de tels mots.

```
import regex
from calibre import replace_entities
from calibre import prepare_string_for_xml

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):

    def replace_word(wmatch):
        # Try to remove the hyphen and replace the words if the resulting
        # hyphen free word is recognized by the dictionary
        without_hyphen = wmatch.group(1) + wmatch.group(2)
        if dictionaries.recognized(without_hyphen):
            return without_hyphen
        return wmatch.group()

    # Search for words split by a hyphen
    text = replace_entities(match.group()[1:-1])  # Handle HTML entities like &amp;
    corrected = regex.sub(r'(\w+)\s*-\s*(\w+)', replace_word, text, flags=regex.
↪VERSION1 | regex.UNICODE)
    return '>%s<' % prepare_string_for_xml(corrected)  # Put back required entities
```

Utilisez cette fonction avec la même expression de recherche que précédemment, à savoir

```
>[^<>]+<
```

Et il réparera magiquement tous les mots avec de mauvais traits d'union dans le texte du livre. L'astuce principale est d'utiliser l'un des utiles arguments supplémentaires de la fonction de remplacement, `dictionaries`. Celui-ci se réfère aux dictionnaires que l'éditeur utilise lui-même pour vérifier orthographiquement le texte dans le livre. Ce que fait cette fonction est de regarder aux mots séparés par un trait d'union, supprimer le trait d'union et vérifier si le dictionnaire reconnait le mot composé, s'il le fait, les mots originaux sont remplacés par le mot composé libre du trait d'union.

Notez qu'une limitation à cette technique est qu'elle ne fonctionnera uniquement qu'avec les livres unilingues, car, par défaut, `dictionaries.recognized()` utilise la langue principale du livre.

Auto numérotation des sections

Maintenant nous allons voir quelque chose d'un peu différent. Supposons que votre fichier HTML ait beaucoup de sections, chacune avec un titre dans une balise `<h2>` ressemblant à `<h2>Un certain texte</h2>`. Vous pouvez créer une fonction personnalisée qui numérotera automatiquement ces titres avec des numéros de section consécutifs, afin qu'elles ressemblent à `<h2>1. Un certain texte</h2>`.

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    section_number = '%d. ' % number
    return match.group(1) + section_number + match.group(2)

# Ensure that when running over multiple files, the files are processed
# in the order in which they appear in the book
replace.file_order = 'spine'
```

Utilisez la avec l'expression de recherche

```
(?s)(<h2[^<>]*>)(.+?</h2>)
```

Placez le curseur en haut du fichier et cliquez *Remplacez tout*

Cette fonction utilise l'un des utiles arguments supplémentaires pour `replace()` l'argument ``number`. Lorsque vous faites un *Remplacer Tout* le nombre est incrémenté automatiquement pour chaque correspondante successive .

Une autre nouvelle fonctionnalité est l'utilisation de *replace.file_order`* – régler cela à `spine` signifie que si cette recherche est exécutée sur de multiples fichiers HTML, ces fichiers seront traités dans l'ordre dans lequel ils apparaissent dans le livre. Voir *Choisissez l'ordre de fichier lors de l'exécution sur de multiples fichiers HTML* (page 109) pour des détails.

Auto créer une Table des Matières

Finalement, essayons quelque chose d'un peu plus ambitieux. Supposons que votre livre à des titres dans des balises `h1` et `h2` qui ressemblent à <h1 id=»someid»>Un certain texte</h1>``. Nous auto générerons une Table des Matières HTML basée sur ces titres. Créez la fonction personnalisée suivante :

```
from calibre import replace_entities
from calibre.ebooks.oeb.polish.toc import TOC, toc_to_html
from calibre.gui2.tweak_book import current_container
from calibre.ebooks.oeb.base import xml2str

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    if match is None:
        # All matches found, output the resulting Table of Contents.
        # The argument metadata is the metadata of the book being edited
        if 'toc' in data:
            toc = data['toc']
            root = TOC()
            for (file_name, tag_name, anchor, text) in toc:
                parent = root.children[-1] if tag_name == 'h2' and root.children else␣
↪root
                parent.add(text, file_name, anchor)
            toc = toc_to_html(root, current_container(), 'toc.html', 'Table of␣
↪Contents for ' + metadata.title, metadata.language)
            print(xml2str(toc))
        else:
            print('No headings to build ToC from found')
    else:
        # Add an entry corresponding to this match to the Table of Contents
        if 'toc' not in data:
            # The entries are stored in the data object, which will persist
            # for all invocations of this function during a 'Replace All' operation
            data['toc'] = []
        tag_name, anchor, text = match.group(1), replace_entities(match.group(2)),␣
↪replace_entities(match.group(3))
        data['toc'].append((file_name, tag_name, anchor, text))
        return match.group()  # We don't want to make any actual changes, so return␣
↪the original matched text

# Ensure that we are called once after the last match is found so we can
# output the ToC
replace.call_after_last_match = True
# Ensure that when running over multiple files, this function is called,
```

(suite sur la page suivante)

(suite de la page précédente)

```
# the files are processed in the order in which they appear in the book
replace.file_order = 'spine'
```

Et utilisez là pour trouver l'expression :

```
<(h[12]) [^<>]* id=['"]([^'"]+)['"][^<>]*>([^<>]+)
```

Lancez la recherche sur *Tous les fichiers textes* et à la fin de la recherche, une fenêtre apparaîtra avec « Résultat de débogage pour votre fonction » qui contiendra la Table des Matières HTML, prête à être collée dans `toc.html`.

La fonction ci-dessus est fortement commentée, aussi elle devrait être facile à suivre. La nouvelle fonctionnalité clé est l'utilisation d'un autre argument supplémentaire utile à la fonction `replace()`, l'objet `data`. L'objet `data` est un *dictionnaire* Python qui persiste entre les invocations successives de `replace()` pendant une seule opération *Remplacer Tout*.

Une autre nouvelle fonctionnalité est l'utilisation de `call_after_last_match` – paramétrer cela à `True` sur la fonction `replace()` signifie que l'éditeur appellera `replace()` une fois de plus après que toutes les correspondances aient été trouvées. Pour cet appel supplémentaire, l'objet correspondant sera``None``

Ceci était juste une démonstration pour vous montrer la puissance du mode fonction. Si vous avez réellement besoin de générer une Table des Matières à partir des titres dans votre livre, vous aurez mieux en utilisant l'outil Table des Matières dédié dans *Tools → Table des Matières*.

L'API pour le mode fonction

Toutes les fonctions du mode fonction doivent être des fonctions Python nommées replace, avec la signature suivante

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    return a_string
```

Quand un rechercher/remplacer est exécuté, pour chaque correspondance qui est trouvée, la fonction `replace()` sera appelée, elle doit retourner la chaîne de remplacement pour cette correspondance. Si aucuns remplacements ne doivent être effectués, elle devrait retourner `match.group()` qui est la chaîne originale. Les divers arguments de la fonction `replace()` sont documentées ci-dessous.

L'argument `match`

L'argument `match` représente la correspondance actuellement trouvée. C'est un objet Python Match[37]. Sa méthode la plus utile est `group()` qui peut être utilisée pour récupérer le texte apparié correspondant à la capture de groupes individuels dans l'expression régulière de recherche.

[37] https://docs.python.org/library/re.html#match-objects

L'argument `number`

L'argument `number` est le nombre de l'actuelle correspondance. Lorsque vous exécutez *Remplacer Tout*, chaque correspondance successive entraînera `replace()` à être appelé avec un nombre incrémenté. La première correspondance porte le numéro 1.

L'argument `file_name`

Ceci est le nom du fichier dans lequel la correspondance actuelle a été trouvée. Lors d'une recherche à l'intérieur d'un texte marqué, le `file_name` est vide. Le `file_name` est de forme reconnue, une chemin relatif à la racine du livre, utilisant `/` comme séparateur de chemin.

L'argument `metadata`

Ceci représente les métadonnées du livre actuel, comme le titre, les auteurs, la langue, etc. C'est un objet de la classe *`calibre.ebooks.metadata.book.base.Metadata`* (page 221). Les attributs utiles incluent, `title`, `authors` (une liste d'auteur) et `language` (le code de langue).

L'argument `dictionaries`

Ceci représente la collection de dictionnaires utilisés pour la vérification orthographique du livre actuel. Sa méthode la plus utile est `dictionaries.recognized(word)` qui renverra ``True```si le mot analysé est reconnu par le dictionnaire de la langue actuelle du livre.

L'argument `data`

Ceci est un un simple `dictionnaire` Python. Lorsque vous exécutez *Remplacer tout*, toutes les correspondances suivantes entraîneront un appel de `replace()` avec le même `dictionnaire` en tant que données. Vous pouvez donc l'utiliser pour stocker arbitrairement des données entre des invocations de `replace()` pendant une opération *Remplacer tout*.

L'argument `functions`

L'argument `functions` vous donne accès à toutes les autres fonctions définies par l'utilisateur. Ceci est utile pour ré-utiliser du code. Vous pouvez définir les fonctions utilitaires à un seul endroit et les ré-utiliser dans toutes vos autres fonctions. Par exemple, supposons la création d'une fonction nommée `My Function` comme ceci :

```
def utility():
    # do something

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...
```

Alors, dans une autre fonction, vous pouvez accéder à la fonction `utility()` comme ceci :

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    utility = functions['My Function']['utility']
    ...
```

Vous pouvez aussi utiliser les objets fonctions pour stocker des données persistantes, qui peuvent être ré-utilisées par d'autres fonctions. Par exemple, vous pourriez avoir une fonction qui lorsqu'elle s'exécute avec *Remplacer Tout* collecte des données et une autre fonction qui les utilisera lorsqu'elle sera lancée plus tard. Considérez les deux fonctions suivantes :

```
# Function One
persistent_data = {}

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...
    persistent_data['something'] = 'some data'

# Function Two
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    persistent_data = functions['Function One']['persistent_data']
    ...
```

Dépanner vos fonctions

Vous pouvez dépanner les fonctions que vous créez en utilisant la fonction standard `print()` de Python. Le résultat de print sera affiché dans une fenêtre popup après que le Rechercher/remplacer soit terminé. Nous avons vu un exemple de l'utilisation de `print()` pour sortir une table des matières entière plus haut.

Choisissez l'ordre de fichier lors de l'exécution sur de multiples fichiers HTML

Lorsque vous lancez *Remplacer tout* sur de multiples fichiers HTML, l'ordre dans lequel les fichiers sont traités dépend de quels fichiers vous avez ouvert pour l'édition. Vous pouvez forcer la recherche à traiter les fichiers dans l'ordre dans lequel ils apparaissent en paramétrant l'attribut `file_order` de votre fonction, comme ceci :

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...

replace.file_order = 'spine'
```

`file_order` accepte deux valeurs, `spine` et `spine-reverse` qui entraîne le traitement de multiples fichiers dans l'ordre où ils apparaissent dans le livre, soit descendant soit ascendant, respectivement.

Appelle votre fonction une dernière fois après que la dernière correspondance ait été trouvée.

Parfois, comme dans l'exemple de la table des matières auto-générée ci-dessus, il est utile que votre fonction soit appelée une nouvelle fois après que la dernière correspondance ait été trouvée. Vous pouvez faire ceci en paramétrant l'attribut `call_after_last_match` dans votre fonction, comme ceci :

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...

replace.call_after_last_match = True
```

Ajouter le résultat de la fonction au texte marqué

Lorsque vous effectuez un rechercher et remplacer sur du texte marqué, il est quelquefois utile d'ajouter du texte à la fin du texte marqué. Vous pouvez faire cela en paramétrant l'attribut `append_final_output_to_marked` sur votre fonction (notez que vous aurez également besoin de paramétrer ``call_after_last_match`), comme ceci :

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...
    return 'some text to append'

replace.call_after_last_match = True
replace.append_final_output_to_marked = True
```

Suppression du dialogue résultant lorsque des recherches sont effectuées sur du texte marqué

Vous pouvez également supprimer le résultat du dialogue (qui peut ralentir l'application répétée d'un rechercher/remplacer sur beaucoup de blocs de texte) en paramétrant l'attribut `suppress_result_dialog` sur votre fonction, comme ceci :

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...

replace.suppress_result_dialog = True
```

Plus d'exemples

Plus d'exemples utiles, fournis par des utilisateurs de calibre, peuvent être trouvés dans le Forum Éditeur de livre numérique de calibre[38].

[38] https://www.mobileread.com/forums/showthread.php?t=237181

Fragments

L'éditeur de livre numérique de calibre supporte les *fragments*. Un fragment est un morceau de texte qui est soit souvent réutilisé soit contient beaucoup de texte redondant. L'éditeur vous permet d'insérer un fragment avec seulement quelques séquences de touches. Par exemple, supposons que vous devez souvent trouver par vous même les balises link insérées lors de l'édition des fichiers HTML, alors vous pouvez simplement taper `<a` dans l'éditeur et presser `Control+J`. L'éditeur développera cela en

```
<a href="filename"></a>
```

Pas seulement ça, le mot `filename` sera sélectionné, avec le curseur placé sur lui, ainsi vous pouvez facilement taper le vrai nom de fichier, en utilisant la petite fonctionnalité de l'éditeur *Auto-complétion* (page 116). Et une fois que vous avez fini de taper le nom de fichier, pressez `Control+J` à nouveau et le curseur sautera vers la position entre les balises `<a>` comme cela vous pouvez facilement y taper le texte pour le lien.

Le système des fragments dans l'éditeur est très sophistiqué, il y a quelques fragments intégrés et vous pouvez créer les vôtres pour convenir à votre style d'édition.

L'exposé qui suit sur les fragments intégrés devrait aider à illustrer la puissance du système des fragments.

Note : Vous pouvez aussi utiliser les fragments dans les champs d'entrée de texte dans le panneau *Rechercher & Remplacer*, cependant, les espaces réservés (en utilisant `Control+J` pour s'y déplacer) ne fonctionneront pas.

Les fragments intégrés

Les fragments intégrés sont décrits ci-dessous . Notez que vous pouvez les remplacer en créant vos propres fragments avec le même texte déclencheur.

Insertion du texte de remplissage [Lorem]

Le premier fragment intégré, et le plus simple, est utilisé pour insérer du texte de remplissage dans un document. Le texte de remplissage est est issu de De finibus bonorum et malorum[39] une oeuvre philosophique de Cicéron (Traduite en anglais). Pour l'utiliser, taper simplement `Lorem` dans un fichier HTML et pressez `Control+J`. Il sera remplacé par quelques paragraphes de remplissage.

La définition de ce fragment est très simple, le texte déclencheur est défini comme étant `Lorem` et le modèle est défini simplement comme le texte littéral à intégrer. Vous pouvez facilement le personnaliser pour utiliser votre texte de remplissage favori.

Insérer une balise HTML auto-fermante [<>]

Maintenant jetons un oeil à un exemple simple du puissant concept de *placeholders*. Disons que vous voulez insérer une balise auto-fermante `<hr/>`. Tapez juste `<>` et pressez `Control+J`, l'éditeur développera le fragment en

```
<|/>
```

Ici, le symbole `|` représente la position actuelle du curseur. Vous pouvez alors taper `hr` et presser `Control+J` pour déplacer le curseur après la fin de la balise. Ce fragment est défini comme

[39] https://fr.wikipedia.org/wiki/De_finibus_bonorum_et_malorum

```
Trigger: <>
Template: <$1/>$2
```

Les espaces réservés sont simplement le signe dollar ($) suivi par un nombre. Lorsque le fragment est développé en appuyant sur `Control+J` le curseur est positionné au premier espace réservé (l'espace réservé avec le nombre le plus bas). Quand vous pressez `Control+J` à nouveau le curseur saute vers l'espace réservé suivant (l'espace réservé suivant avec le nombre plus élevé).

Insérer une balise lien HTML [<a]

Les balises lien HTML partagent toutes une structure commune. Elles ont un attribut `href` et un certain texte entre les balises ouvrantes et fermantes. Un fragment pour rendre la frappe plus efficiente nous introduira certaines fonctionnalités supplémentaires des espaces réservés. Pour utiliser ce fragment, tapez simplement `<a` et appuyez `Control+J`. L'éditeur développera ceci en

```
<a href="filename|"></a>
```

Pas seulement ça, le mot `filename` sera sélectionné, avec le curseur placé sur lui, ainsi vous pouvez facilement taper le vrai nom de fichier, en utilisant la petite fonctionnalité de l'éditeur *Auto-complétion* (page 116). Et une fois que vous avez fini de taper le nom de fichier, pressez `Control+J` à nouveau et le curseur sautera vers la position entre les balises `<a>` comme cela vous pouvez facilement y taper le texte pour le lien. Après que vous ayez fini de taper le texte, appuyez `Control+J` encore une fois pour sauter au point après la balise fermante. Le fragment est défini comme

```
Trigger: <a
Template: <a href="${1:filename}">${2*}</a>$3
```

Il y a ici deux nouvelles fonctionnalités. Premièrement l'espace réservé `$1` est devenu plus complexe. Il inclut maintenant un certain *texte par défaut* (le mot `filename`). Si un espace réservé contient du texte par défaut , le texte par défaut est substitué pour l'espace réservé quand le fragment est développé. Aussi quand vous sautez vers un espace réservé avec du texte par défaut en utilisant `Control+J`, le texte par défaut est sélectionné. De cette manière, vous pouvez utiliser le texte par défaut comme un rappel pour vous pour remplir d'importantes parties du modèle. Vous pouvez définir le texte par défaut pour un espace réservé en utilisant la syntaxe : `${<number>:texte par défaut}`.

L'autre nouvelle fonctionnalité est que le second espace réservé a un astérisque qui le suit (`${2*}`). Cela signifie que tout texte qui est sélectionné avant de développer le modèle est substitué pour l'espace réservé. Pour voir ceci en action, sélectionner un certain texte dans l'éditeur, appuyer `Control+J`, tapez `<a` et pressez `Control+J` à nouveau, le modèle sera développé en

```
<a href="filename">whatever text you selected</a>
```

Insérer une balise image HTML [<i]

C'est très similaire à insérer un lien HTML, comme nous avons vu ci-dessus. Cela vous permet d'entrer rapidement une balise `<img src="filename" alt="description" />` et sauter entre les attributs `src` et `alt`

```
Trigger: <i
Template: <img src="${1:filename}" alt="${2*:description}" />$3
```

Insérer une balise HTML quelconque[<<]

Ceci vous permet d'insérer une balise HTML quelconque (ou d'envelopper le texte précédemment sélectionné dans la balise). Pour l'utiliser, tapez simplement << et pressez `Control+J`. L'éditeur développera cela en :

```
<|></>
```

Tapez le nom de la balise, par exemple : `span` et pressez `Control+J`, ce qui résultera en

```
<span>|</span>
```

Vous noterez que la balise fermante a été automatiquement remplie avec `span`. Ceci est accompli avec encore une autre fonctionnalité des espaces réservés, *mirroring*. Mirroring signifie simplement que si vous spécifiez plus d'une fois l'échantillon d'espace réservé dans le modèle, le second et et toutes les positions futures seront automatiquement remplies avec ce que vous avez tapé en première position, quand vous appuyez sur `Control+J`. La définition pour ce fragment est

```
Trigger: <<
Template: <$1>${2*}</$1>$3
```

Comme vous pouvez voir, le premier espace réservé ($1) a été spécifié deux fois, la deuxième fois dans la balise fermante, ce qui copiera simplement ce que vous aurez tapé dans la balise ouvrante.

Insérer une balise HTML quelconque avec un attribut de classe [<c]

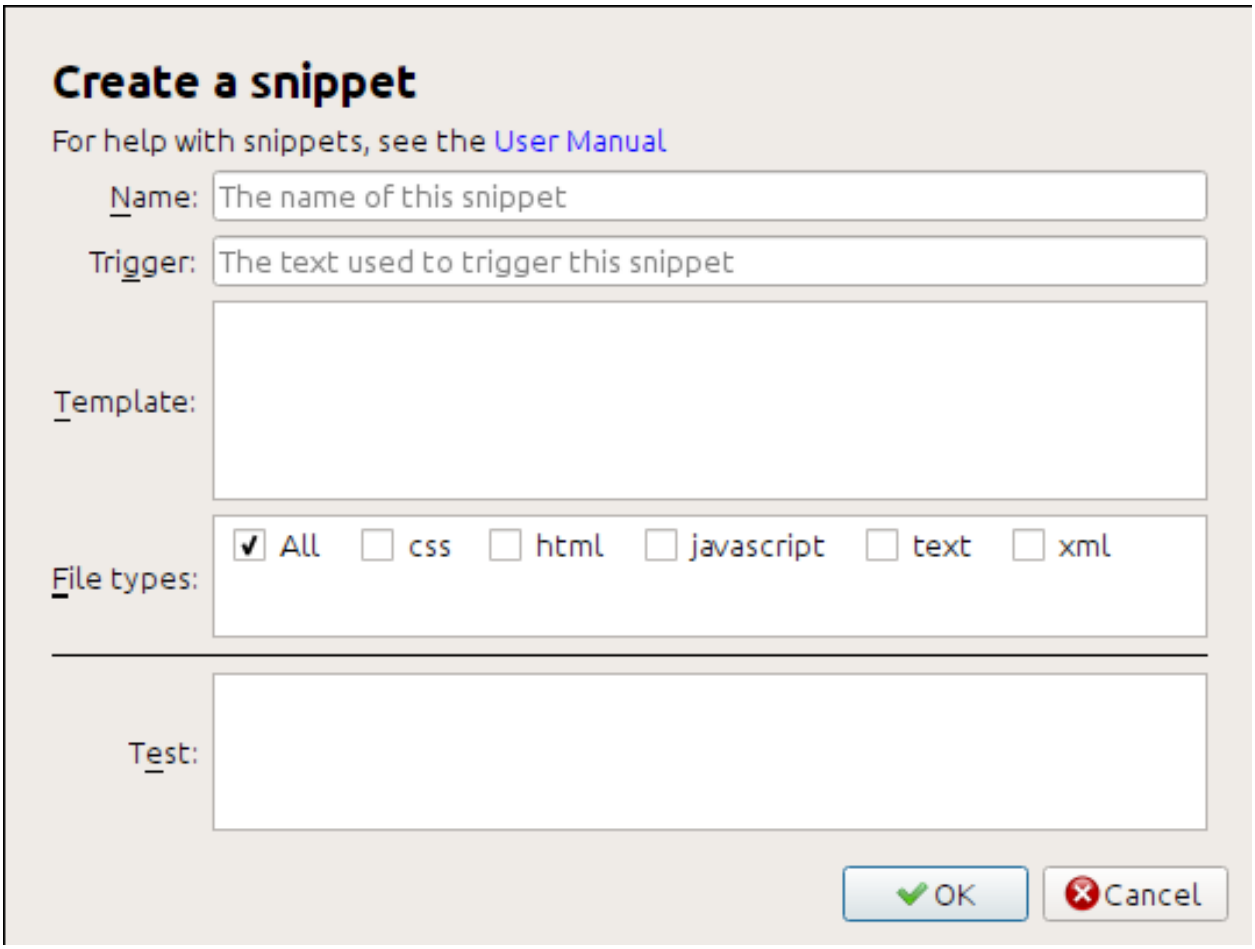
Ceci est très similaire à l'exemple d'insertion d'une balise quelconque ci-dessus, exceptez qu'il assume que vous voulez spécifier une classe pour la balise

```
Trigger: <c
Template: <$1 class="${2:classname}">${3*}</$1>$4
```

Ceci vous permettra de taper d'abord le nom de la balise, presser `Control+J`, taper le nom de la classe, presser `Control+J` tapez le contenu de la balise et pressez `Control+J` une dernière fois pour sortir de la balise. La balise de fermeture sera remplie automatiquement.

Créer vos propres fragments

Les fragments sont vraiment brillants car vous pouvez créer les vôtres pour suivre votre style d'édition. Pour créer vos propres fragments allez à *Éditer → Préférences → Propriétés de l'éditeur → Gérer les fragments* dans l'éditeur. Cela fera apparaître une boîte de dialogue facile à utiliser pour vous aider à créer vos propres fragments. Cliquez simplement le bouton :guilabel :`Ajouter un fragment`et vous verrez une boîte de dialogue qui ressemble à :



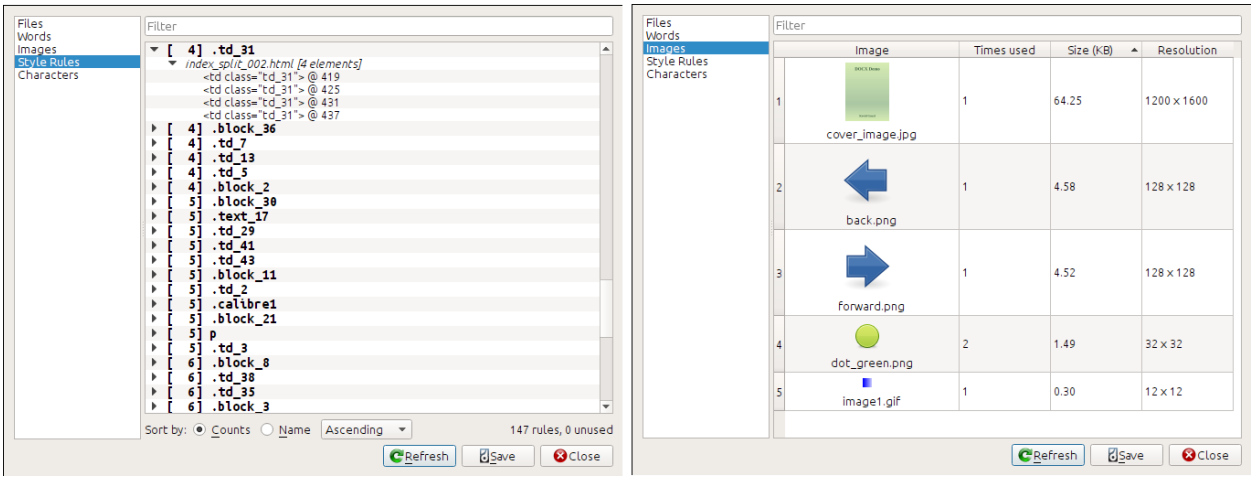
Tout d'abord donnez un nom à votre fragment, quelque chose de descriptif, pour aider à identifier le fragment par après. Spécifiez alors le *déclencheur*. Un déclencheur est simplement le texte que vous avez à taper avant de presser `Control+J` en vue de développer le fragment.

Spécifiez alors le modèle de fragment. Vous devriez commencer avec l'un des exemples ci-dessus et le modifier pour correspondre à vos besoins. Finalement, spécifier quels types de fichiers pour lesquels vous voulez que le fragment soit actif. De cette manière vous pouvez avoir de multiples fragments avec le même texte déclencheur qui fonctionne différemment dans différents types de fichiers.

L'étape suivante est de tester votre fragment nouvellement créé. Utilisez la boîte *Test* en bas. Entrez-y le texte déclencheur et appuyez sur `Control+J` pour développer le fragment et sauter entre les espaces réservés.

5.8.9 L'outil Rapports

L'éditeur inclut un outil sympa *Rapports* (via ;guilabel :*Outils->Rapports*) qui affiche des résumés des fichiers, images, mots, caractères et styles utilisés dans le livre. Chaque ligne dans le rapport est liée. Double cliquer une ligne envoie à l'endroit dans le livre où cet élément est utilisé ou défini (selon le cas). Par exemple, dans la vue *Liens*, vous pouvez double cliquer les entrées de la colonne *Source* pour sauter là où le lien est défini et les entrées dans la colonne *Cible* pour sauter là où le lien pointe.



5.9 Fonctionnalités spéciales dans l'éditeur de code

L'éditeur HTML calibre est très puissant. Il a beaucoup de fonctionnalités qui rendent l'édition de l'HTML (et du CSS) plus facile.

5.9.1 Mise en évidence de la syntaxe

L'éditeur HTML a des mises en évidence très sophistiquées de la syntaxe. Les fonctionnalités comprennent :

- Le texte inclut dans des balises gras, italique et titre est établi en gras/italique
- Au fur et à mesure que vous bougez votre curseur au travers de l'HTML, les balises HTML correspondantes sont mises en surbrillance, et vous pouvez sauter vers la balise ouvrante ou fermante avec les raccourcis clavier `Ctrl+{` et **:kdb :`Ctrl+}`**. De même, vous pouvez sélectionner le contenu d'une balise avec `Ctrl+Alt+T` ou `Ctrl+Shift+T`
- L'HTML invalide est mis en surbrillance avec un soulignement rouge
- Les erreurs d'orthographe dans le texte inclut dans les balises HTML et les attributs comme titre sont mis en surbrillance. La correction orthographique est fonction de la langue, basée sur la valeur de l'attribut `lang` de la balise courante et la langue globale du livre.
- Le CCS intégré dans des balises `<style>` est mis en surbrillance
- Les caractères spéciaux qui peuvent être difficiles à distinguer comme les espaces insécables, différents types de traits d'union, etc. sont mis en surbrillance.
- Les liens vers d'autres fichiers dans des balises `<a>`, des balises `<img>` et `<link>` ont tous leurs noms de fichiers mis en surbrillance. Si le nom de fichier vers lequel il pointe n'existe pas, le nom de fichier est marqué avec un soulignement rouge.

5.9.2 Aide sensible au contexte

Vous pouvez faire un clic droit sur une balise HTML ou une propriété CSS pour obtenir de l'aide sur cette balise ou propriété.

Vous pouvez également maintenir enfoncée la touche `Ctrl` et cliquer sur n'importe quel nom de fichier inclut dans une balise link pour ouvrir automatiquement ce fichier dans l'éditeur. De même, `Ctrl` en cliquant sur le nom d'une classe vous amènera à la première règle de style qui correspond à la balise et à la classe.

Un clic droit sur le nom d'une classe dans un fichier HTML vous permettra de renommer la classe, ce qui modifiera toutes les occurrences de la classe dans le livre et toutes ses feuilles de style.

5.9.3 Auto-complétion

Pendant l'édition d'un livre numérique, l'une des tâches des plus fastidieuses est la création de liens vers d'autres fichiers dans le livre, ou vers les feuilles de style CSS ou les images. Vous devez indiquer le nom du fichier correct et le chemin relatif au fichier. L'éditeur a une auto-complétion pour faciliter cela.

Au fur et à mesure que vous tapez un nom de fichier, l'éditeur fait apparaître automatiquement des suggestions. Utilisez simplement la touche `Tabulation` pour sélectionner le nom de fichier correct. L'éditeur offre même des suggestions pour des liens pointant vers une ancre dans un autre fichier HTML. Après que vous ayez tapé le caractère `#`, l'éditeur vous montrera une liste de toute les ancres dans le fichier cible, avec un petit extrait de texte pour vous aider à choisir la bonne ancre.

Notez que contrairement à la plupart des autres systèmes de complétion, la système de complétion de l'éditeur utilise la correspondance en sous-séquence. Cela signifie que vous pouvez taper seulement deux ou trois lettres de n'importe où dans le nom de ficher pour compléter le nom de fichier. Par exemple, disons que vous voulez le nom de fichier `../images/arrow1.png`, vous pouvez simplement taper `ia1` et appuyer sur la touche `Tabulation` pour compléter le nom de fichier. Lors de la recherche de correspondances, le système de complétion priorise les lettres qui sont au début d'un mot, ou immédiatement après un séparateur de chemin. Une fois que aurez pris l'habitude d'utiliser ce système, vous trouverez qu'il vous fait gagner beaucoup de temps et d'efforts.

5.9.4 Fragments

L'éditeur de livre numérique de calibre supporte les *fragments*. Un fragment est un morceau de texte qu est soit souvent réutilisé soit contient beaucoup de texte redondant. L'éditeur vous permet d'insérer un fragment avec seulement quelques séquences de touches. Les fragments sont très puissants, avec beaucoup de fonctionnalités, comme les emplacements d'où vous pouvez sauter de l'un à l'autre, refléter automatiquement le texte répété et ainsi de suite. Pour plus d'informations, voir fragments.

CHAPITRE 6

Le Serveur de contenu calibre

Le :guilabel :`Serveur de contenu`calibre vous permet d'accéder à vos bibliothèques calibre et de lire des livres directement dans un navigateur sur votre smartphone ou votre tablette. De ce fait, vous n'avez pas besoin d'installer une application de lecture ou de gestion de livres sur votre téléphone. Utilisez simplement le navigateur. Le serveur télécharge et conserve le livre que vous lisez dans un cache hors ligne afin que vous puissiez le lire même quand il n'y a pas de connexion internet.

Contenu

- *Accéder au Serveur de contenu depuis d'autres périphériques* (page 118)
  - *Accès au serveur à partir de périphériques sur votre réseau domestique* (page 118)
  - *Accès au serveur depuis n'importe où sur Internet* (page 119)
- *L'interface du serveur* (page 120)
  - *La liste de livres* (page 120)
  - *Le lecteur de livre* (page 120)
- *Compatibilité du navigateur* (page 120)
- *Activer le support hors ligne* (page 121)
- *Gérer les comptes utilisateurs uniquement à partir de la ligne de commande* (page 121)
- *Intégration du Serveur de contenu calibre dans d'autres serveurs* (page 121)
  - *Utilisation d'un hôte totalement virtuel* (page 121)
  - *Utilisation d'un préfixe d'URL* (page 122)
- *Créer un service pour le serveur calibre sur un système Linux moderne* (page 123)

Pour démarrer le serveur, cliquez le bouton *Connecter/partager`et choisissez :guilabel :`Démarrer le Serveur de contenu*. Vous pourriez avoir un message du pare-feu ou de l'antivirus de votre ordinateur demandant si vous voulez autoriser l'accès à `calibre.exe`. Cliquez le bouton `Autoriser` ou `OK`. Ouvrez alors un navigateur (de préférence Chrome ou Firefox) sur votre ordinateur et entrez l'adresse suivante :

http://127.0.0.1:8080

Ceci ouvrira une page dans dans votre navigateur vous affichant vos bibliothèques calibre, cliquez sur n'importe laquelle et parcourez les livres qu'elle contient. Cliquez sur un livre, et il vous montrera toutes les métadonnées à propos du livre, accompagnées des boutons *Lire* et *Télécharger* le livre. Cliquez sur le bouton *Lire* pour commencer la lecture du livre.

Note : L'adresse utilisée ci-dessus `http://127.0.0.1:8080` ne fonctionnera que sur l'ordinateur qui fait tourner calibre. Pour accéder au serveur à partir d'autres ordinateurs, téléphones, tablettes, etc… il va falloir un peu plus de travail, comme indiqué dans la section suivante.

6.1 Accéder au Serveur de contenu depuis d'autres périphériques

Il existe deux types d'accès depuis un périphérique distant dont vous aurez généralement besoin. Le premier type, plus simple, est l'accès depuis votre réseau domestique. Si vous exécutez calibre sur un ordinateur de votre réseau domestique et que vous avez également connecté vos autres périphériques au même réseau, vous devriez pouvoir accéder facilement au serveur sur ces appareils.

6.1.1 Accès au serveur à partir de périphériques sur votre réseau domestique

Après avoir démarré le serveur dans calibre comme décrit ci-dessus, cliquez à nouveau sur le bouton *Connecter/partager*. Au lieu de l'action *Démarrer le serveur de contenu*, vous devriez voir l'action *Arrêter le Serveur de contenu*. À droite de cette action il y a une adresse IP et un numéro de port. Cela ressemble à un tas de nombres séparés par deux points. Par exemple

```
Stop Content server [192.168.1.5, port 8080]
```

Ces chiffres indiquent l'adresse à utiliser pour se connecter au serveur depuis vos périphériques. En suivant l'exemple ci-dessus, l'adresse devient

```
http://192.168.1.5:8080
```

La première partie de l'adresse est toujours `http://` la partie suivante est l'adresse IP, c'est à dire les chiffres avant la virgule, et pour terminer le numéro de port qui doit être ajouté à l'adresse IP avec deux points (`:`). Si vous êtes chanceux, c'est tout ce dont vous devriez avoir besoin pour explorer vos bibliothèques calibre sur votre périphérique. Sinon, continuez la lecture.

Dépannage de la connexion sur un réseau domestique

Si vous ne parvenez pas à accéder au serveur depuis votre périphérique, essayez ce qui suit :

1. Vérifiez que le serveur fonctionne en ouvrant l'adresse `http://127.0.0.1:8080` dans un navigateur fonctionnant sur le même ordinateur que le serveur.
2. Vérifiez que votre pare-feu/anti-virus autorise les connexions à votre ordinateur sur le port `8080` et au programme calibre. La façon la plus simple de savoir si le pare-feu/anti-virus sont la source du problème est de les désactiver temporairement puis d'essayer de vous connecter. Avant d'éteindre le pare-feu, commencez par vous déconnecter d'Internet pour garder votre ordinateur en sécurité.
3. Vérifiez que votre périphérique et votre ordinateur sont sur le même réseau. Cela signifie qu'ils doivent être tous les deux connectés au même routeur sans fil. En particulier aucun ne doit être en train d'utiliser une connexion cellulaire ou une connexion directe WiFi fournie par votre FAI.

4. Si votre configuration réseau n'est pas standard, il se peut que l'adresse IP affichée dans le menu *Connecter/partager* soit incorrecte. Dans ce cas, vous devrez déterminer l'adresse IP correcte à utiliser par vous-même. Malheureusement, compte tenu de la grande diversité des configurations réseau possible, il est impossible de donner un mode d'emploi général.
5. Si vous avez configuré un nom d'utilisateur et un mot de passe, essayez tout d'abord sans pour voir s'il cela pause problème. Certaines périphériques e-ink ont des navigateurs qui ne prennent pas en charge l'authentification. Vous pouvez parfois contourner cela en incluant le nom d'utilisateur et le mot de passe dans l'URL, par exemple : `http://nomutilisateur:motdepasse@192.168.1.2:8080`.
6. Si vous êtes coincé, vous pouvez toujours demander de l'aide sur les forums utilisateurs de calibre[40].

6.1.2 Accès au serveur depuis n'importe où sur Internet

Avertissement : Avant de commencer, vous devriez activer la protection nom d'utilisateur/mot de passe dans le serveur, sinon n'importe qui dans le monde pourra accéder à vos livres. Allez dans *Préférences → Partager → Partager sur le net* et activez l'option *Exiger un nom d'utilisateur et un mot de passe pour accéder au serveur de contenu*.

Bien que les détails pour la configuration de l'accès via Internet varient en fonction de la configuration du réseau et du type d'ordinateur que vous utilisez, la méthode de base est la suivante.

1. Trouvez l'adresse IP externe de l'ordinateur sur lequel vous allez exécuter le serveur. Vous pouvez l'obtenir en visitant le site What is my IP address[41] dans un navigateur sur l'ordinateur.
2. Si l'ordinateur est derrière un routeur, activez le transfert de port (ou "port forwarding") sur le routeur pour transférer le port `8080` (ou n'importe quel port sur lequel vous choisissez d'exécuter le serveur de contenu calibre) sur l'ordinateur.
3. Assurez-vous que le serveur calibre est autorisé par tout programme pare-feu/anti-virus sur votre ordinateur.
4. Maintenant, vous devriez pouvoir accéder au serveur depuis n'importe quel périphérique connecté à Internet en utilisant l'adresse IP obtenue lors de la première étape. Par exemple, si l'adresse IP obtenue était `123.123.123.123` et que le port que vous utilisez pour le serveur calibre est `8080`, l'adresse à utiliser sur votre périphérique devient : `http://123.123.123.123:8080`.
5. Accessoirement, vous pouvez utilisez un service comme no-ip[42] pour mettre en place une adresse facile à retenir à utiliser à la place de l'adresse IP obtenue à la première étape.

Note : Pour une sécurité maximum, vous devriez aussi activer l'HTTPS sur le serveur de contenu. Vous pouvez soit le faire directement dans le serveur en fournissant le chemin vers le certificat HTTPS à utiliser dans les configurations avancées du serveur, ou vous pouvez configurer un proxy inversé (reverse proxy) comme décris ci-dessous, pour utiliser une configuration HTTP déjà existante.

[40] https://www.mobileread.com/forums/forumdisplay.php?f=166
[41] https://www.whatismyip.com/
[42] https://www.noip.com/free

6.2 L'interface du serveur

L'interface du serveur est une version simplifiée de l'interface principale de calibre, optimisée pour l'utilisation d'écrans tactiles. L'écran d'accueil vous affiche les livres que vous êtes en train de lire aussi bien que de vous permettre de choisir une bibliothèque calibre que vous voulez parcourir. Le serveur, dans calibre, vous donne accès à toutes vos bibliothèques, pas seulement à une seule, comme précédemment.

6.2.1 La liste de livres

La liste de livres du serveur est une simple grille de couvertures. Appuyez sur une couverture pour voir les métadonnées détaillées pour un livre, ou pour lire un livre. Si vous préférez une liste plus détaillée, vous pouvez changer la vue par défaut en cliquant sur les trois points verticaux dans le coin supérieur droit.

Trier et rechercher la liste de livres devrait être familier aux utilisateurs de calibre. Ils peuvent être accédés en cliquant sur leurs icônes dans la partie supérieure droite. Ils fonctionnent tous deux exactement de la même manière que dans le programme principal calibre. La page de recherche vous permet même de construire des requêtes en cliquant sur auteurs/étiquettes/etc., exactement comme vous le faites en utilisant le Navigateur d'étiquettes du programme principal.

Une fonctionnalité très appréciée du programme principal, *Bibliothèques virtuelles* est aussi présente dans l'interface du serveur. Cliquez sur les trois points verticaux dans le coin supérieur doit pour choisir une Bibliothèque virtuelle.

6.2.2 Le lecteur de livre

Vous pouvez lire n'importe quel livre dans votre bibliothèque calibre en appuyant simplement dessus et sélectionnant ensuite le bouton *Lire*. Le lecteur de livres est très simple de fonctionnement. Vous pouvez à la fois appuyer et glisser pour tourner les pages. Glisser vers le haut/bas bascule entre les chapitres. Appuyer le quart supérieur de l'écran vous donne des contrôles détaillés et des préférences de visionneuse.

Si vous laissez le Serveur de contenu en fonctionnement, vous pouvez même ouvrir le même livre sur de multiples appareils et il se souviendra de votre dernière position de lecture. S'il ne le fait pas vous pouvez forcer une synchronisation en appuyant dans le quart supérieur et en choisissant *Sync*.

6.3 Compatibilité du navigateur

Le nouveau serveur calibre fait un usage intensif des fonctionnalités avancées de HTML 5 et CSS 3. En tant que tel, il nécessite un navigateur à jour pour être utilisé. Il a été testé sur Android Chrome et iOS Safari ainsi que sur Chrome et Firefox sur le bureau.

Le serveur est attentif à utiliser les fonctionnalités qui ont soit déjà été standardisées soit sur les voies de la standardisation. En tant que tel s'il ne fonctionne pas actuellement avec votre navigateur, il le fera probablement une fois que ce navigateur aura rattrapé son retard.

Si vous utilisez un navigateur particulièrement ancien ou limité ou que vous n'aimez pas exécuter JavaScript, vous pouvez utiliser la vue « mobile », en ajoutant simplement `/mobile` à l'adresse du serveur.

Note : Sous IOS, Apple permet un seul moteur de navigation, donc Firefox, Chrome et Safari sont actuellement le même navigateur sous le capot. La nouvelle interface serveur requiert IOS 10.3.2 ou plus récent. Sous Android, le serveur a été testé avec Chrome 58 et plus récent.

6.4 Activer le support hors ligne

Les fabricants de navigateurs ont essayé de forcer les gens à utiliser SSL en désactivant les fonctions avancées de leur navigateur pour les connexions HTTP simples. L'une de ces victimes est ApplicationCache, qui a été utilisé dans calibre pour l'assistance hors ligne. Par conséquent, aujourd'hui, malheureusement, le mode hors ligne ne fonctionne que si vous gardez l'onglet du navigateur ouvert. De plus, dans Firefox sous Android, vous devrez taper `about:config` et rechercher `browser.tabs.useCache` et le définir sur `true`.

6.5 Gérer les comptes utilisateurs uniquement à partir de la ligne de commande

Le programme calibre a une belle section dans *Préférences* pour vous permettre de gérer les comptes utilisateurs pour le serveur. Si vous voulez exécuter le serveur autonome et ne pouvez exécuter le programme principal calibre sur le même ordinateur/compte utilisateur, vous pouvez également gérer les utilisateurs en utilisant uniquement la ligne de commande.

Vous pouvez gérer les comptes utilisateur en utilisant l'option `--manage-users` sur le programme `calibre-server` autonome. Supposez que vous voulez stocker la base de données utilisateur dans le répertoire `/srv/calibre`, alors créez-le en exécutant

```
calibre-server --userdb /srv/calibre/users.sqlite --manage-users
```

Suivez juste les instructions pour créer les comptes utilisateur, définir leurs permissions, etc. Une fois que vous avez fini, vous pouvez exécuter le serveur en tant que

```
calibre-server --userdb /srv/calibre/users.sqlite --enable-auth
```

Il utilisera les comptes utilisateur que vous avez créé à l'étape précédente.

6.6 Intégration du Serveur de contenu calibre dans d'autres serveurs

Ici, nous allons vous montrer comment intégrer le serveur de contenu calibre dans un autre serveur. La raison la plus fréquente de faire cela est d'utiliser SSL ou de desservir la bibliothèque calibre comme partie d'un site plus grand. La technique de base consiste à exécuter le serveur calibre et à configurer un proxy inversé à partir du serveur principal.

Un proxy inversé est quand votre serveur normal accepte les requêtes entrantes et les transmet au serveur calibre. Il lit alors la réponse du serveur calibre et la transmet au client. Ceci signifie que vous pouvez simplement exécuter le serveur calibre comme à la normale sans essayer de l'intégrer étroitement avec votre serveur principal.

6.6.1 Utilisation d'un hôte totalement virtuel

La configuration la plus simple est de dédier un hôte totalement virtuel au serveur calibre. Dans ce cas, exécuter le serveur calibre comme ceci :

```
calibre-server
```

Maintenant configurer l'hôte virtuel dans votre serveur principal, par exemple pour nginx :

```
http {
    client_max_body_size 64M;  # needed to upload large books
}
```

(suite sur la page suivante)

(suite de la page précédente)

```
server {
    listen [::]:80;
    server_name myserver.example.com;

    location / {
        proxy_pass http://127.0.0.1:8080;
    }
}
```

Ou, pour Apache :

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so

<VirtualHost *:80>
    ServerName myserver.example.com
    AllowEncodedSlashes On
    ProxyPreserveHost On
    ProxyPass "/"  "http://localhost:8080/"
</VirtualHost>
```

6.6.2 Utilisation d'un préfixe d'URL

Si vous ne voulez pas dédier un hôte totalement virtuel à calibre. vous pouvez l'avoir pour utiliser un préfixe d'URL. Démarrer le serveur calibre comme ceci :

```
calibre-server --url-prefix /calibre --port 8080
```

Ici le paramètre clé est `--url-prefix/calibre`. Ceci entraîne le Serveur de contenu à servir toutes les URLs préfixées par `/calibre`. Pour voir ceci en action, visitez `http://localhost:8080/calibre` dans votre navigateur. Vous devriez voir le site web normal du Serveur de contenu, mais maintenant il s'exécutera sous `/calibre`.

Avec nginx, la configuration nécessaire est

```
http {
    client_max_body_size 64M;  # needed to upload large books
}

proxy_set_header X-Forwarded-For $remote_addr;
location /calibre/ {
    proxy_buffering off;
    proxy_pass http://127.0.0.1:8080$request_uri;
}
location /calibre {
    # we need a trailing slash for the Application Cache to work
    rewrite /calibre /calibre/ permanent;
}
```

Pour Apache, activer tout d'abord les modules proxy dans Apache, en ajoutant ce qui suit à `httpd.conf` :

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
```

La technique exacte pour activer les modules proxy variera selon votre installation Apache. Une fois que vous avez activé les modules proxy, ajouter les règles suivantes à `httpd.conf` (ou si vous utilisez des hôtes virtuels au fichier conf pour l'hôte virtuel en question) :

```
AllowEncodedSlashes On
RewriteEngine on
RewriteRule ^/calibre/(.*) http://127.0.0.1:8080/calibre/$1 [proxy]
RedirectMatch permanent ^/calibre$ /calibre/
```

C'est tout, vous serez maintenant capable d'accéder au Serveur de contenu calibre sous l'URL `/calibre` dans votre serveur principal. Les règles ci-dessus transitent toutes les requêtes sous `/calibre` vers le serveur calibre s'exécutant sur le port 8080 et grâce à l'option `--url-prefix` ci-dessus, le serveur calibre les traite de manière transparente.

Note : Lors de l'utilisation d'un proxy inversé, vous devrez dire au Serveur de contenu calibre d'écouter uniquement le localhost, en utilisant `--listen-on 127.0.0.1`. De cette manière, le serveur écoutera uniquement les connexions venant du même ordinateur, càd du proxy inversé.

Note : Si vous avez configuré SSL pour votre serveur principal, vous devrez dire au serveur calibre d'utiliser l'authentification basique au lieu de l'authentification digest, plus rapide. Pour faire cela, transmettez l'option `--auth-mode=basic` à `calibre-server`.

6.7 Créer un service pour le serveur calibre sur un système Linux moderne

Vous pouvez aisément créer un service pour exécuter le serveur calibre sur sur un système moderne (systemd[43]) basé sur Linux. Il faut seulement créer le fichier `/etc/systemd/system/calibre-server.service` avec le contenu afficher ci-dessous

```
[Unit]
Description=calibre Content server
After=network.target

[Service]
Type=simple
User=mylinuxuser
Group=mylinuxgroup
ExecStart=/opt/calibre/calibre-server "/path/to/calibre library folder"

[Install]
WantedBy=multi-user.target
```

Changer `mylinuxuser` et `mylinuxgroup` en n'importe quel utilisateur et groupe sous lesquels vous désirez que le serveur tourne. Ceci devrait être le même utilisateur et groupe que ceux qui possèdent les fichiers dans le répertoire de la bibliothèque calibre. Notez que ce n'est généralement pas une bonne idée d'exécuter le serveur en tant que root. Changer également le chemin vers le répertoire de la bibliothèque calibre pour qu'il convienne à votre système. Vous pouvez ajouter de multiples bibliothèques si nécessaire. Voir l'aide pour la commande `calibre-server`.

Maintenant exécutez

[43] https://www.freedesktop.org/wiki/Software/systemd/

```
sudo systemctl start calibre-server
```

pour démarrer le serveur. Vérifiez son statut avec

```
sudo systemctl status calibre-server
```

Pour le faire démarrer au boot, exécutez

```
sudo systemctl enable calibre-server
```

Note : Le serveur calibre ne nécessite pas un serveur X en cours d'exécution, mais il nécessite que les librairies X soient installées.

Note : Le serveur calibre supporte également l'activation du socket systemd, vous pouvez donc l'utiliser, si nécessaire, bien sûr.

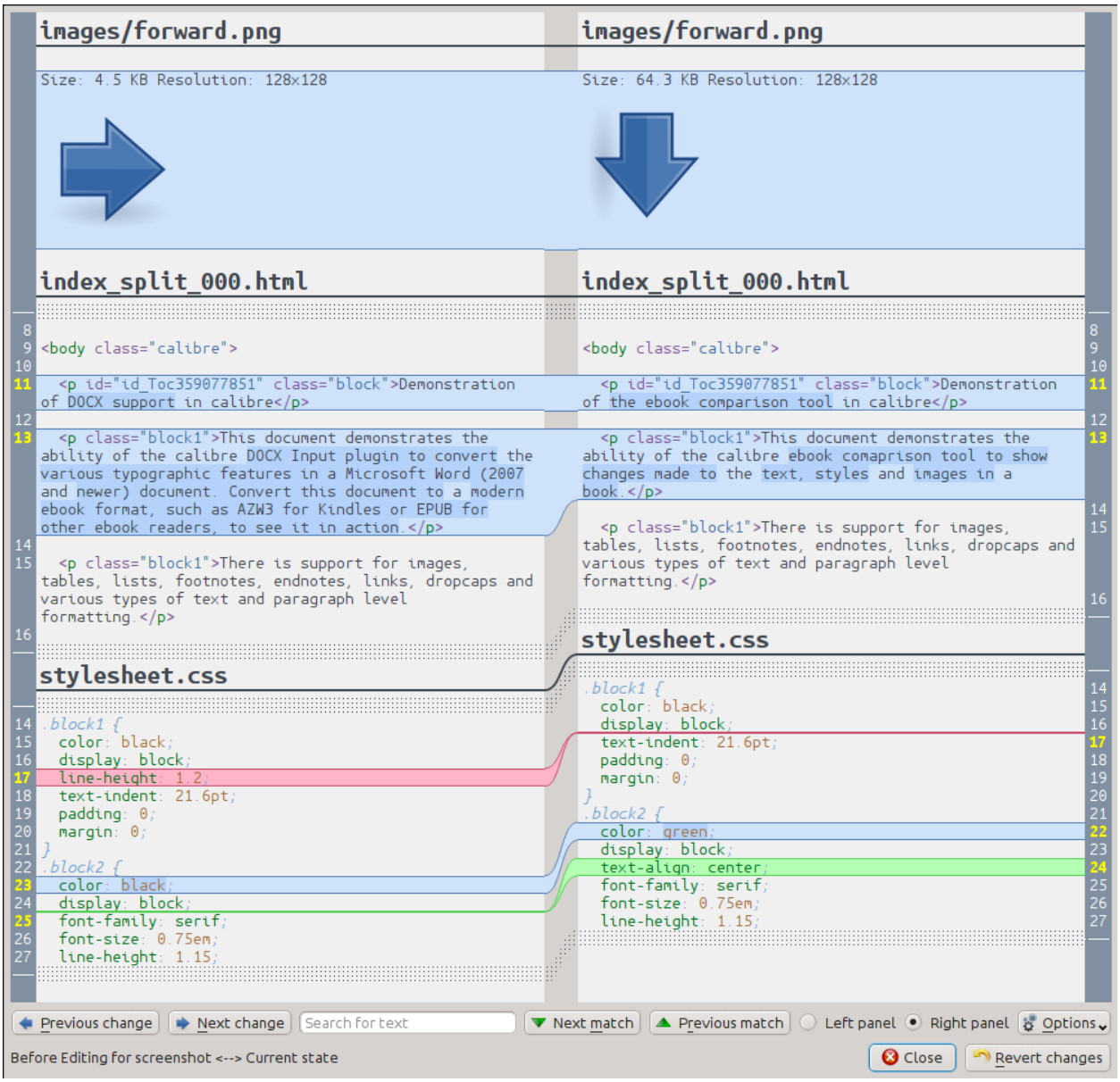
CHAPITRE 7

Comparer des livres numériques

calibre inclut un outil intégré de comparaison de livres numériques qui peut être utilisé pour voir ce qui a été changé à l'intérieur d'un livre numérique après son édition ou sa conversion. Il peut comparer des livres aux formats EPUB et AZW3.

Pour l'utiliser, soit ouvrez le livre dans l'outil Éditer et cliquez alors *Fichier → Comparer à un autre livre* ou utilisez le panneau *Détails du livre* (page 21). Si vous faites une conversion d'EPUB à EPUB, l'EPUB original sera sauvegardé comme ORIGINAL_EPUB. Faites simplement un clic droit sur l'entrée ORIGINAL_EPUB dans le panneau Détails du livre et choisissez *Comparer au format EPUB*.

L'outil de comparaison qui s'ouvre ressemblera à la capture d'écran ci-dessous. Il vous montre les différences dans le texte, les styles et le images dans les livres choisis.



7.1 Comprendre la vue de comparaison

Comme cela peut être vu dans la capture d'écran plus haut, la vue de comparaison montre les différences entre les deux livres côte à côte. Seules les différences, avec quelques lignes de contexte autour d'elles sont montrées. Ceci rend facile de voir en un coup d'œil seulement ce qui a été changé à l'intérieur d'un grand document comme un livre.

Le texte ajouté est montré avec un fond vert, le texte supprimé avec un fond rouge et le texte changé avec un fond bleu.

Les numéros de ligne de tout texte changé sont montrés sur les côtés, rendant facile le déplacement à un changement particulier dans l'éditeur. Quand vous ouvrez l'outil de comparaison depuis l'éditeur, vous pouvez aussi double-cliquer sur une ligne dans le panneau de droite pour aller automatiquement à cette ligne dans l'éditeur.

Une technique utile en comparant des livres est de dire à l'outil de comparaison d'embellir les fichiers de texte et de style avant de calculer les différences. Ceci peut souvent avoir comme conséquence d'obtenir des résultats plus propres et plus de facilité à suivre les différences. Pour faire ceci, cliquez sur le bouton *Options* en bas à droite et choisir *Embellir les*

fichiers avant de comparer. Notez que l'embellissement peut parfois avoir des effets non désirés, comme altéré un balisage invalide pour le rendre valide. Vous pouvez également changer le nombre de lignes de contexte autour des différences via le bouton guilabel :*Options*.

Vous pouvez chercher après n'importe quel texte dans les différences par l'intermédiaire de la Barre de recherche en bas. Vous aurez besoin de spécifier dans quel panneau rechercher, le *Gauche* ou le *Droit*.

7.2 Lancement de l'outil de comparaison

L'outil de comparaison est le plus utile quand vous avez deux versions du mêmes livre et que vous voulez voir ce qui est différent entre elles. À cet effet, il y a plusieurs manières de lancer l'outil.

7.2.1 Comparer deux fichiers de livre numérique

Ouvrez le premier fichier dans l'outil *Éditer des livres numériques* (page 81). Maintenant cliquez *Fichier → Comparer à un autre livre* et choisissez le deuxième livre. (il doit être dans le même format que le premier). La vue comparaison s'ouvrira avec le fichier en cours d'édition sur la droite et le second fichier sur la gauche.

7.2.2 Comparaison de ORIGINAL_FMT à FMT

Quand vous faites une conversion dans calibre à partir d'un FMT vers lui-même, le fichier original est sauvegardé comme ORIGINAL_FMT. Vous pouvez voir ce qui a été changé par la conversion, en faisant un clic droit sur l'entrée ORIGINAL_FMT dans le panneau ref :*book_details* dans la fenêtre principale de calibre et en sélectionnant *Comparer à FMT*. La vue comparaison s'ouvrira avec le ORIGINAL_FMT sur la gauche et le FMT sur la droite.

7.2.3 Comparer un point de contrôle à l'état actuel du livre tout en l'éditant

L'outil edition possède une caractéristique très utile points de contrôle. Ceci vous permet de sauver l'état actuel du livre comme un *point de contrôle* nommé, auquel vous pouvez retourner si vous n'aimez pas les modifications que vous avez apportées depuis la création du point de contrôle. Des points de contrôle sont également créés automatiquement quand vous effectuez diverses actions automatisées dans l'éditeur. Vous pouvez voir la liste de points de contrôle en allant dans *Aperçu → Points de contrôle* et employez alors le bouton *Comparer* pour comparer le livre au point de contrôle sélectionné à l'état actuel. L'outil de comparaison montrera le point de contrôle sur la gauche et la version actuelle sur la droite.

CHAPITRE 8

Éditer les métadonnées d'un livre numérique

Contenu

- *Édition des métadonnées d'un livre à la fois* (page 129)
  - *Télécharger les métadonnées* (page 130)
  - *Gérer les formats de livre* (page 130)
  - *Tout à propos des couvertures* (page 130)
- *Édition des métadonnées de plusieurs livres à la fois* (page 130)
  - *Rechercher Et Remplacer* (page 131)
  - *Téléchargement de métadonnées par lot* (page 132)
- *Ajouter des fichiers de données supplémentaires à un livre* (page 132)

Les livres numériques nous parviennent sous toutes les formes et tailles, et le plus souvent leurs métadonnées (éléments comme le titre/auteur/séries/éditeur) sont incomplètes ou incorrectes. La manière la plus simple de changer des métadonnées dans calibre est de simplement double cliquer sur une entrée et de saisir la correction. Plus sophistiquée, « édition puissante » utilise les outils d'édition de métadonnées discutés ci-dessous.

8.1 Édition des métadonnées d'un livre à la fois

Cliquez sur le livre que vous voulez éditer et cliquez alors sur le bouton *Modifier les métadonnées* ou appuyez sur la touche `E`. Une boîte de dialogue s'ouvre vous permettant d'éditer tous les aspects des métadonnées. Il y a différentes fonctionnalités pour rendre l'édition plus rapide et plus efficiente. Une liste des astuces habituellement utilisées :

- Vous pouvez cliquer le bouton entre titre et auteurs pour les permuter automatiquement.
- Vous pouvez cliquer le bouton à coté de Tri par auteur pour faire en sorte que calibre le remplisse automatiquement en utilisant le tri des valeurs stockées avec chaque auteur. Utilisez la boîte de dialogue *Gérer les auteurs* pour voir

et changer les valeurs de tri des auteurs. Cette boîte de dialogue peut être ouverte en cliquant longuement sur le bouton à coté de Tri par auteur.

- Vous pouvez cliquer sur le bouton qui précède les étiquettes pour utiliser **:quilabel :`Gestion des étiquettes`** pour gérer les étiquettes associées avec le livre.
- Les cases « ids » peuvent être utilisées pour entrer un numéro d'ISBN (et beaucoup d'autres types d'id), elles auront un arrière plan rouge si vous entrez un ISBN invalide. Il sera vert pour les ISBN valides.
- La case Tri par auteur sera rouge si la valeur de tri par auteur diffère de ce que calibre pense qu'elle devrait être.

8.1.1 Télécharger les métadonnées

La plus agréable fonctionnalité de la boîte de dialogue d'édition des métadonnées est sa capacité à remplir automatiquement beaucoup de champs de métadonnées en obtenant les métadonnées de divers sites web. Actuellement, calibre utilise Google Books et Amazon. Le téléchargement de métadonnées peut compléter Titre, Auteur, Série, Étiquettes, Notation, Description et ISBN pour vous.

Pour utiliser le téléchargement, complétez les champs titre et auteur et cliquez sur le bouton *Télécharger les métadonnées*. calibre vous présentera une liste des livres qui correspondent le plus précisément au titre et à l'auteur. Si vous complétez l'ISBN en premier, il sera utilisé en priorité sur le titre et l'auteur. Si aucune correspondance n'est trouvée, essayez de rendre votre recherche un peu moins spécifique en incluant seulement quelques mots clés dans le titre et seulement le nom de l'auteur.

8.1.2 Gérer les formats de livre

Dans calibre, une seule entrée de livre peut avoir beaucoup de *formats* associés. Par exemple vous avez pu obtenir les Œuvres de Shakespeare au format EPUB et l'avoir converti plus tard en MOBI pour lire sur votre Kindle. calibre gère automatiquement les multiples formats pour vous. Dans la section *Formats disponibles* de la boîte de dialogue Éditer les métadonnées, vous pouvez gérer ces formats. Vous pouvez ajouter un nouveau format, supprimer un format existant et aussi demander à calibre de régler les métadonnées et la couverture pour l'entrée de livre depuis les métadonnées dans l'un des formats.

8.1.3 Tout à propos des couvertures

Vous pouvez demander à calibre de télécharger les couvertures de livre pour vous, si le livre a un ISBN connu. Alternativement, vous pouvez spécifier un fichier sur votre ordinateur à utiliser comme couverture. calibre peut même générer une couverture par défaut pourvue de métadonnées basiques pour vous. Vous pouvez glisser et déposer des images sur la couverture pour la changer et aussi faire un clic droit pour copier/coller des images de couverture.

En complément, il y a un bouton pour ajuster automatiquement les bords de la couverture, au cas où votre image de couverture a un vilain bord.

8.2 Édition des métadonnées de plusieurs livres à la fois

Sélectionnez d'abord les livres que vous voulez éditer en maintenant `Ctrl` ou `Shift` et en les cliquant. si vous sélectionnez plus d'un livre, cliquez le bouton *Éditer les metadonées* entraînera l'ouverture de la boîte de dialogue d'édition de métadonnées par *lot*. En utilisant cette boîte de dialogue, vous pouvez rapidement régler l'auteur/l'éditeur/la notation/les étiquettes/ les séries etc. d'un tas de livres à la même valeur. Ceci est particulièrement utile si vous venez juste d'importer un nombre de livres qui ont des métadonnées en commun. Cette boîte de dialogue est très puissante, par exemple, elle a un onglet **:guilable :`Rechercher et remplacer`** que vous pouvez utiliser pour effectuer des opérations par lot sur les métadonnées et même copier des métadonnées d'une colonne à un autre.

La boite de dialogue normale d'édition des métadonnées a aussi des boutons *Suivant* et *Précédent* que vous pouvez utiliser pour éditer les métadonnées de plusieurs livres l'un après l'autre.

8.2.1 Rechercher Et Remplacer

La boîte de dialogue *Modifier les métadonnées de nombreux livres* vous permet d'effectuer des opérations de recherche et de remplacement arbitrairement puissante sur les livres sélectionnés. Elle utilise par défaut un rechercher et remplacer en texte simple, mais elle supporte aussi *les expressions régulières*. Pour plus d'informations sur les expressions régulières, voir *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225).

Comme noté plus haut, il y a deux modes de rechercher et remplacer : la correspondance de caractère et l'expression régulière. La correspondance de caractère regardera dans *le champ de recherche* que vous choisissez après les caractères que vous avez entré dans la case *Rechercher* et remplace ces caractères par ceux que vous avez entré dans la case *Remplacer par*. Chaque occurrence des caractères recherchés dans le champ sera remplacée. Par exemple, supposons que le chant faisant l'objet de la recherche contient *a bad cat*, si vous recherchez après un *a* à remplacer par HELLO, alors le résultat sera *HELLO bHELLOd cHELLOt*.

Si le champ qui fait l'objet de la recherche est un champ multiple comme étiquettes, alors chaque étiquette est traitée séparément. Par exemple, si vos étiquettes contiennent *Horreur, Effrayant, l'expression de recherche `r*, ne correspondra pas à quelque chose parce que l'expression sera d'abord appliquée à *Horreur* et ensuite à *Effrayant*.

Si vous voulez que la recherche ignore les différences majuscules/minuscules, décochez la case *Sensible à la casse*.

Vous pouvez faire en sorte que calibre change la casse du résultat (l'information après le remplacement s'est produite) en choisissant l'une des fonctions de la case *Casse à appliquer après le remplacement*. Les opérations disponibles sont :

- *Minuscule* – changer tous les caractères dans le champ en minuscule
- *Majuscule* – changer tous les caractères dans le champ en majuscule
- *Casse de titre* – met une majuscule à chaque mot du résultat.

La case *Votre test* vous est procurée pour entrer du texte pour vérifier que rechercher/remplacer fait ce que vous voulez. Dans la majorité des cas la case de test livre sera suffisante, mais il est possible y a un cas que vous voulez vérifier qui n'apparaît pas dans ces cases. Entrer le cas dans *Votre test*.

Le mode expression régulière a quelques différences avec le mode caractère, au delà (bien sûr) de l'utilisation des expressions régulières. La première est que ces fonctions sont appliquées aux parties de la chaîne qui correspondent à la chaîne de recherche, pas au champ entier. La seconde est que ces fonctions s'appliquent à la chaîne de remplacement, pas au champ entier.

La troisième et plus importante est que la chaîne de remplacement peut faire référence à des parties la chaîne de recherche en utilisant des références arrières. Une référence arrière is `\\n` où n est un entier qui se réfère aux n groupes mis entre parenthèses dans l'expression de recherche. Par exemple, prenons le même exemple que plus haut, *a bad cat*, une expression de recherche *a (…) (…)* et une expression de remplacement *a \2 \1*, le résultat sera *a cat bad* Merci de regardez le *Tout à propos de l'utilisation des expressions régulières dans calibre* (page 225) pour plus d'informations sur les références arrières.

Un modèle utile : supposez que vous voulez changer la casse d'un champ entier. La manière la plus facile de faire ceci est d'employer le mode caractère, mais supposons que vous voulez aller plus loin en employant le mode expression régulière. L'expression de recherche devrait être *(^.*$)*, l'expression de remplacement devrait être *\1*, et la fonction désirée sensible à la casse devra être sélectionnée.

Finalement, en mode expression régulière vous pouvez copier des valeurs d'un champ à un autre. Rendez simplement la source et le champ de destination différente. La copie peut remplacer le champ de destination, ajouter au champ (ajouté au début), ou l'apposer au champ (ajoutez à l'extrémité). La case à cocher *usecomma* indique à calibre (ou pas) d'ajoutez une virgule entre le texte et le champ de destination dans les modes ajoutez au début et adjonction. Si la destination est multiple (par exemple, des étiquettes), alors vous ne pouvez pas décocher cette case.

Recherchez et remplacez est fait après que tous les autres changements de métadonnées dans les autres onglets sont appliqués. Ceci peut mener à une certaine confusion, parce que les cases de test montreront l'information avant que les autres changements, mais l'opération sera appliquée après les autres changements. Si vous avez n'importe quels doutes au sujet de ce qui va se produire, ne mélangez pas rechercher/remplacer avec d'autres changements.

8.2.2 Téléchargement de métadonnées par lot

Si vous voulez télécharger les métadonnées pour de multiples livres en une fois, faites un clic droit sur le bouton *Editer les métadonnées* et sélectionnez *Télécharger les métadonnées et les couvertures*. Vous pouvez choisir de télécharger seulement des métadonnées, seulement des couvertures, ou les deux.

8.3 Ajouter des fichiers de données supplémentaires à un livre

calibre peut stocker un nombre quelconque de fichiers de données supplémentaires associés à un livre. Il peut s'agir de couvertures alternatives, de matériel supplémentaire, etc. Ils ne peuvent pas être consultés directement ou utilisés comme sources de conversion. Ils ne sont pas non plus indexés par le moteur de recherche en texte intégral de calibre. Pour les voir, les ajouter ou les supprimer, sélectionnez le livre et faites un clic droit sur le bouton *Editer les métadonnées* et choisissez *Gérer les fichiers de données*. Cela ouvrira une fenêtre dans laquelle vous pourrez effectuer des opérations sur ces fichiers. Vous pouvez également faire un clic droit sur le bouton *Ajouter des livres* et choisir *Ajouter des fichiers de données aux enregistrements des livres sélectionnés* pour ajouter plus rapidement des fichiers de données.

CHAPITRE 9

Questions Fréquemment Posées

Contenu

- *Conversion de format de livre numérique* (page 133)
- *Intégration d'un périphérique* (page 137)
- *Gestion de la bibliothèque* (page 144)
- *Divers* (page 150)

9.1 Conversion de format de livre numérique

Contenu

- *Quels sont les formats que supporte calibre en entrée/sortie pour la conversion ?* (page 134)
- *Quels sont les meilleurs formats source pour la conversion ?* (page 134)
- *J'ai converti un fichier PDF, mais le résultat présente divers problèmes ?* (page 134)
- *Comment puis-je convertir mon fichier contenant des caractères non-anglais, ou des guillemets intelligents ?* (page 134)
- *Quel est le problème avec la Table des Matières dans les fichiers MOBI ?* (page 135)
- *Comment puis-je convertir une collection de fichiers HTML dans un ordre spécifique ?* (page 136)
- *L'EPUB que j'ai produit avec calibre n'est pas valide ?* (page 136)
- *Comment puis-je utiliser certaines les fonctionnalités avancées des outils de conversion ?* (page 136)

9.1.1 Quels sont les formats que supporte calibre en entrée/sortie pour la conversion ?

calibre supporte la conversion de nombreux formats en entrée vers de nombreux formats en sortie. Il peut convertir tous les formats d'entrée dans la liste ci-dessous, vers chaque format de sortie.

Formats d'entrée : AZW, AZW3, AZW4, CBZ, CBR, CB7, CBC, CHM, DJVU, DOCX, EPUB, FB2, FBZ, HTML, HTMLZ, LIT, LRF, MOBI, ODT, PDF, PRC, PDB, PML, RB, RTF, SNB, TCR, TXT, TXTZ

Formats de sortie : AZW3, EPUB, DOCX, FB2, HTMLZ, OEB, LIT, LRF, MOBI, PDB, PMLZ, RB, PDF, RTF, SNB, TCR, TXT, TXTZ, ZIP

Note : PRC est un format générique, calibre supporte les fichiers PRC avec des entêtes TextRead et MOBIBook. PDB est aussi un format générique. calibre supporte les fichiers eReader, Plucker (format d'entrée uniquement), PML et zTxt PDB. Le support de DJVU est seulement pour la conversion de fichiers DJVU qui contiennent du texte intégré. Ceux-ci sont typiquement produits par un logiciel OCR. Les livres MOBI peuvent être de deux types Mobi6 et KF8. calibre supporte pleinement chacun des deux. Les fichiers MOBI ont souvent des extensions de fichier .azw ou .azw3. Les fichiers DOCX de Microsoft Word 2007 et ultérieurs sont supportés.

9.1.2 Quels sont les meilleurs formats source pour la conversion ?

Par ordre décroissant de préférence : LIT, MOBI, AZW, EPUB, AZW3, FB2, FBZ, DOCX, HTML, PRC, ODT, RTF, PDB, TXT, PDF

9.1.3 J'ai converti un fichier PDF, mais le résultat présente divers problèmes ?

PDF est un terrible format à partir duquel convertir. Pour une liste des différents problèmes que vous pouvez rencontrer lors de la conversion PDF, voir : *Conversion de documents PDF* (page 76).

9.1.4 Comment puis-je convertir mon fichier contenant des caractères non-anglais, ou des guillemets intelligents ?

Il y a deux aspects à ce problème :

1. Connaitre l'encodage du fichier source : calibre essaie de deviner quel encodage de caractères vos fichiers sources utilisent, mais souvent, c'est impossible, aussi devez-vous lui préciser l'encodage à utiliser. Cela peut être fait dans l'interface graphique utilisateur par l'intermédiaire du champ *Encodage des caractères d'entrée* dans la section *Apparence → Texte* de la boite de dialogue de conversion. Les outils en lignes de commande ont tous une option `ebook-convert-txt-input --input-encoding`.
2. Lors de l'ajout de fichiers HTML dans calibre, vous devrez peut-être indiquer à calibre quel est l'encodage des fichiers. Pour ce faire, allez dans *Préférences → Avancé → Extensions → extensions → Type de fichier*, et personnalisez l'extension *HTML to ZIP* en lui indiquant quel encodage de caractère vous avez dans vos fichiers HTML Maintenant lorsque vous ajoutez des fichiers HTML dans calibre, ils seront traités correctement. Les fichiers HTML provenant de différentes sources ont souvent des encodages différents, de sorte que vous pouvez avoir à modifier ce paramètre à plusieurs reprises. Un codage courant pour beaucoup de fichiers sur le Web est `cp1252` et je vous suggère de commencer par celui-là. Notez que, lors de la conversion des fichiers HTML, laissez le paramètre d'encodage d'entrée mentionnée ci-dessus à blanc. Car l'extension *HTML to ZIP* convertit automatiquement les fichiers HTML selon une norme d'encodage (utf-8).

9.1.5 Quel est le problème avec la Table des Matières dans les fichiers MOBI ?

La première chose à comprendre est que la plupart des livres numériques ont deux tables des matières. L'une est la Table des Matières traditionnelle, comme la TdM que vous trouvez dans les livres papier. Cette Table des Matières est une partie du flux du document principal et peut être mise en page, comme bon vous semble. Cette TdM est appelée *TdM de contenu*.

Ensuite, il y a la *TdM des métadonnées*. Cette TdM ne fait pas partie du texte du livre et elle est généralement accessible par un bouton spécial sur une liseuse. Par exemple, dans la visionneuse de livre numérique de calibre, vous utilisez le bouton Afficher la Table des Matières pour y accéder. Cette TdM ne peut être mise en forme par le créateur du livre. Sa représentation dépend du logiciel de visualisation.

Dans le format MOBI, la situation est un peu confuse. C'est parce que le format MOBI, le seul parmi les formats de livre numérique principaux, *ne gère pas* correctement les TdM de métadonnées. Un livre MOBI simule la présence d'une TdM de métadonnées en ajoutant une TdM de contenu *supplémentaire* à la fin du livre. Lorsque vous cliquez sur Aller à la Table des Matières sur votre Kindle, c'est vers cette TdM de contenu supplémentaire que le Kindle vous emmène.

Maintenant, il pourrait bien vous sembler que le livre MOBI a deux TdM identiques. Rappelez-vous que, sémantiquement parlant, il y a une TdM de contenu et une TdM de métadonnées, même si elles ont le même contenu et le même aspect. L'une est accessible directement depuis le menu Kindle, l'autre ne peut pas l'être.

Lors de la conversion vers MOBI, calibre détecte la *TdM de métadonnées* dans le document d'entrée et génère une TdM de fin de fichier dans le fichier MOBI de sortie. Vous pouvez désactiver cette fonction par une option dans les paramètres de sortie MOBI. Vous pouvez aussi dire à calibre de la mettre au début ou la fin de l'ouvrage via une option dans les paramètres de sortie MOBI. Rappelez-vous que cette TdM est sémantiquement une *TdM de métadonnées*, dans n'importe quel autre format que MOBI elle *ne peut pas faire partie du texte*. Le fait qu'il s'agisse d'une partie du texte dans MOBI est un accident causé par les limitations de MOBI. Si vous voulez une TdM à un endroit particulier dans le document texte, créez-la à la main. Nous vous recommandons fortement de laisser la valeur par défaut, c'est à dire avec la TdM de métadonnées à la fin du livre. Notez également que si vous désactivez la génération d'une TdM de fin de fichier, le fichier MOBI résultant pourrait ne pas fonctionner sur Kindle, vu que Kindle utilise la TdM de métadonnées pour beaucoup de choses, incluant la fonctionnalité Tourner la Page.

Si vous avez une TdM créée à la main dans le document d'entrée, vous pouvez utiliser les options de détection de TdM dans calibre pour générer automatiquement une TdM de métadonnées à partir de celle-ci. Voir la section sur la conversion du Manuel de l'utilisateur pour plus de détails sur la façon d'utiliser ces options.

Enfin, je vous encourage à abandonner la création d'une TdM de contenu et de n'avoir qu'une TdM de métadonnées dans vos lvres numériques. La TdM de métadonnées fournira aux personnes qui lisent vos livres numériques une expérience de navigation très supérieure (sauf sur le Kindle, où elle est essentiellement la même qu'une TdM de contenu).

Note : Le nouveau format AZW3 a sa propre prise en charge d'une TdM de métadonnées. Cependant, le micrologiciel de la Kindle a tendance à mal fonctionner si vous désactivez la génération de la TdM interne de fin de fichier. Aussi il est recommandé de laisser la TdM générée seule. Si vous créez un fichier AZW3 avec une TdM de métadonnées et pas de TdM générée de fin de fichier, certaines fonctionnalités sur la Kindle ne fonctionneront pas, telle que la fonctionnalité Tourner la Page.

9.1.6 Comment puis-je convertir une collection de fichiers HTML dans un ordre spécifique ?

Pour convertir une collection de fichiers HTML dans un ordre spécifique, vous devez créer un fichier table des matières. Celui-ci est, un autre fichier HTML qui contient des liens vers tous les autres fichiers dans l'ordre souhaité. Un tel fichier ressemble à

```
<html>
   <body>
     <h1>Table of Contents</h1>
     <p style="text-indent:0pt">
        <a href="file1.html">First File</a><br/>
        <a href="file2.html">Second File</a><br/>
        .
        .
        .
     </p>
   </body>
</html>
```

Ensuite, il suffit d'ajouter ce fichier HTML au GUI et d'utiliser le bouton *Convertir* pour créer votre livre numérique. Vous pouvez utiliser l'option dans la section Table des Matières de la boîte de dialogue Convertir pour contrôler la façon dont la Table des Matières est générée.

Note : Par défaut, lorsque vous ajoutez des fichiers HTML, calibre suit les liens dans les fichiers en adoptant un parcours en profondeur *depth first order*. Cela signifie que si A.html pointe vers B.html, C.html et D.html, mais que B.html a également un lien vers D.html, les fichiers seront dans l'ordre suivant : A.html, B.html, D.html, C.html. Si vous voulez l'ordre A.html, B.html, C.html, D.html alors vous devez dire à calibre d'ajouter vos fichiers dans un parcours en largeur *breadth first order*. Pour ce faire, allez dans *Préférences* → *Avancé* → *Extensions* et personnalisez le module *HTML to ZIP*.

9.1.7 L'EPUB que j'ai produit avec calibre n'est pas valide ?

calibre ne garantit pas qu'un EPUB qu'il produit soit valide. La seule garantie qu'il donne est que si vous lui donnez du XHTML 1.1 + CSS 2.1 valide, il va vous afficher un EPUB valide. calibre est conçu pour les consommateurs de livres numériques, non pour les producteurs. Il s'efforce de s'assurer que les EPUBs qu'il produit fonctionnent réellement comme prévu sur une grande variété de périphériques, objectif qui est incompatible avec la production d'EPUBs valides, et qui est beaucoup plus important pour la grande majorité de ses utilisateurs. Si vous avez besoin d'un outil qui produit des EPUBs toujours valides, calibre n'est pas pour vous. Cela signifie, que si vous voulez envoyer un EPUB produit par calibre à une boutique en ligne qui utilise un vérificateur de validité d'EPUB, vous avez à assurer que l'EPUB est valide par vous même, calibre ne le fait pas pour vous – en d'autres mots vous devez alimenter calibre en XHTML + CSS valides comme documents d'entrée.

9.1.8 Comment puis-je utiliser certaines les fonctionnalités avancées des outils de conversion ?

Vous pouvez obtenir de l'aide sur n'importe quelle caractéristique particulière des convertisseurs en passant la souris au-dessus de l'interface graphique ou en exécutant `ebook-convert dummy.html .epub -h` dans un terminal. Un bon endroit pour démarrer est de regarder le fichier de démonstration suivant qui illustre quelques-unes des fonctionnalités avancées html-demo.zip[44].

[44] https://calibre-ebook.com/downloads/html-demo.zip

9.2 Intégration d'un périphérique

Contenu

- *Quels périphériques calibre supporte-t'il ?* (page 137)
- *Comment puis-je aider à ce que mon périphérique soit pris en charge dans calibre ?* (page 137)
- *Mon périphérique n'est pas détecté par calibre ?* (page 138)
- *Mon périphérique est non standard ou inhabituel. Comment faire pour le connecter ?* (page 138)
- *Comment utiliser calibre avec mon iPad/iPhone/iPod touch ?* (page 139)
- *Comment utiliser calibre avec mon téléphone/ma tablette Android ou ma Kindle Fire ?* (page 139)
- *Puis-je accéder à mes livres calibre en utilisant le navigateur web sur mon Kindle ou sur un autre périphérique de lecture ?* (page 141)
- *Je ne peux pas envoyer des courriels en utilisant calibre ?* (page 141)
- *Mon appareil est monté en lecture seule dans Linux, donc calibre ne peut pas s'y connecter ?* (page 142)
- *Pourquoi calibre ne prend pas en charge les collections sur Kindle ou les étagères sur Nook ?* (page 142)
- *J'obtiens un message d'erreur lorsque j'essaie d'utiliser calibre avec mon Kobo Touch/Glo/etc. ?* (page 143)
- *Les couvertures des livres que j'envoie à ma Kindle apparaissent momentanément et sont ensuite remplacées par une couverture générique ?* (page 143)
- *Les couvertures de mes fichiers MOBI n'apparaissent plus sur mon Kindle pour PC/Kindle pour Android/iPad etc.* (page 144)
- *J'ai transféré des livres vers ma Kindle en utilisant calibre et ils n'y apparaissent pas ?* (page 144)

9.2.1 Quels périphériques calibre supporte-t'il ?

calibre peut se connecter directement aux principaux (et la plupart des moins connus) périphériques de lecture de livre numérique, smartphones, tablettes, etc. En outre, en utilisant la fonction *Connecter au dossier*, vous pouvez l'utiliser avec n'importe quel lecteur de livre numérique qui se comporte comme un disque USB. Pour finir, vous pouvez vous connecter sans fil à tout périphérique qui a un navigateur web en utilisant le Serveur de contenu calibre.

9.2.2 Comment puis-je aider à ce que mon périphérique soit pris en charge dans calibre ?

Si votre périphérique apparaît comme un disque USB dans le système d'exploitation, ajouter la prise en charge de celui-ci à calibre est très facile. Nous avons juste besoin de connaître quelques informations de votre part :

- La liste complète des formats de livre numérique que votre périphérique prend en charge.
- Il y a-t'il un dossier spécial dans lequel tous les fichiers livre numérique devront être placés ? Est-ce que le périphérique détecte les fichiers placés dans les sous-dossiers ?
- Nous avons également besoin d'informations sur votre périphérique que calibre recueillera automatiquement. Tout d'abord, si votre périphérique prend en charge les cartes SD, insérez-en une. Ensuite, connectez votre périphérique à l'ordinateur. Dans calibre allez dans *Préférences → Avancé → Divers* et cliquez sur le bouton « Déboguez la

détection du périphérique ». Cela va créer un journal de débogage. Copiez-le dans un fichier et répétez le processus, mais cette fois avec votre périphérique déconnecté de votre ordinateur.

- Envoyez-nous les deux journaux de débogage ainsi que les autres informations recueillies et nous écrirons un pilote pour votre périphérique.

Une fois que vous nous envoyez les informations sur un système d'exploitation particulier, la prise en charge pour ce périphérique dans ce système d'exploitation apparaîtra dans la prochaine version de calibre. Pour nous envoyer les fichiers, ouvrez un rapport de bug et attachez vos informations en pièces jointes. Voir comment rapporter des bugs[45].

9.2.3 Mon périphérique n'est pas détecté par calibre ?

Suivez ces étapes pour trouver le problème :

- Assurez-vous que vous connectez un seul périphérique à la fois à votre ordinateur. Ne pas avoir un autre périphérique pris en charge par calibre comme un iPhone/iPad etc. en même temps.
- Si vous connectez un iDevice Apple (iPad, iPod Touch, iPhone), Apple ne permet plus à des applications tierces de se connecter à leurs périphériques en utilisant un câble USB. A la place veuillez utiliser une connexion sans fil via le Serveur de contenu calibre.
- Si vous connectez un Kindle Fire ou un autre appareil Android, lisez la note sous *En utilisant un cable USB* (page 140).
- Sous macOS si vous obtenez des erreurs d'autorisation lors de la connexion d'un périphérique à calibre, vous pouvez corriger cela en regardant sous :guilabel :`Préférences système > Sécurité et confidentialité > Confidentialité > Fichiers et répertoires".
- Assurez-vous que vous utilisez la dernière version de calibre (Actuellement 7.9.0. La dernière version peut être téléchargée à partir du site web de calibre[46]. Vous pouvez voir quelle version de calibre vous utilisez actuellement en regardant la ligne du bas de la fenêtre principale de calibre.
- Vérifiez que votre système d'exploitation peut voir le périphérique. Autrement dit, le périphérique devrait apparaître dans l'Explorateur Windows (sous Windows) ou le Finder (sous macOS).
- Dans calibre, allez dans *Préférences → Périphérique à ignorer* et vérifiez que votre périphérique n'est pas ignoré.
- Si toutes les étapes précédentes ont échoué, allez dans *Préférences → Divers* et cliquez sur *Déboguer la détection de périphérique* avec votre périphérique connecté et postez la sortie comme un ticket sur le traqueur de bug calibre[47].

9.2.4 Mon périphérique est non standard ou inhabituel. Comment faire pour le connecter ?

En plus de la fonction *Se connecter au dossier* qui se trouve dans le menu contextuel du bouton *Connecter/partager*, calibre fournit une extension de périphérique `Définie par l'utilisateur` qui peut être utilisée pour se connecter à n'importe quel périphérique USB considéré comme un lecteur de disque par votre système d'exploitation. Remarque : Sous Windows, le périphérique doit avoir une lettre de lecteur pour pouvoir être utilisé par calibre. Voir l'extension de périphérique `Préférences -> Extensions -> Extensions interface du périphérique -> User defined` et `Préférences -> Divers-> Récupérer les informations pour définir les préférences de l'utilisateur pour les périphériques connectés` pour plus d'informations. Notez que si vous utilisez un module d'extension personnalisé sur un périphérique normalement détecté par une extension de périphérique calibre, vous devez désactivez tout d'abord l'extension intégrée, de ce fait l'extension de périphérique définie par l'utilisateur est utilisée à la place.

[45] https://calibre-ebook.com/fr/bugs
[46] https://calibre-ebook.com/fr/download
[47] https://bugs.launchpad.net/calibre

9.2.5 Comment utiliser calibre avec mon iPad/iPhone/iPod touch ?

Une autre manière aisée de naviguer dans votre collection calibre sur votre périphérique Apple est d'utiliser le le Serveur, qui rend votre collection disponible sur le net. Effectuez d'abord les étapes suivantes dans calibre

- Définir le Format de Sortie Préféré de calibre à ePub (le format de sortie peut être paramétré dans *Préférences → Interface → Comportement*)
- Définissez le profil de sortie pour iPad (cela fonctionnera pour iPhone/iPod aussi), sous *Préférences → Conversion → Options communes → Mise en page*
- Convertir les livres que vous voulez lire sur votre iDevice au format EPUB en les sélectionnant et en cliquant sur le bouton *Convertir*.
- Lancez le Serveur de contenu en cliquant sur le bouton *Connecter/partager* et laissez calibre ouvert. Vous pouvez aussi dire à calibre de démarrer automatiquement le serveur de contenu via *Préférences → Partager → Partager via internet*.

Le Serveur de contenu calibre vous permet de lire des livres directement dans Safari. En complément, il y a beaucoup d'autres applications pour votre iDevice qui peuvent se connecter au Serveur de contenu calibre. Par exemple : Marvin, Mapleread et iBooks lui-même.

Utilisation du serveur de contenu

Lancez le navigateur Safari et saisissez l'adresse IP et le port de l'ordinateur exécutant le serveur calibre, comme ceci :

```
http://192.168.1.2:8080/
```

Remplacez `192.168.1.2` par l'adresse IP locale de l'ordinateur qui fait tourner le serveur calibre. Voir *Le Serveur de contenu calibre* (page 117) pour plus d'informations sur l'exécution du serveur et comment trouver la bonne adresse IP à utiliser.

Vous verrez une liste de livres dans Safari, cliquez sur n'importe quel livre et il vous sera donné l'opportunité de soit le télécharger soit de le lire dans le navigateur. Si vous choisissez de le télécharger, Safari vous invite ensuite à l'ouvrir avec iBooks.

De nombreuses applications de lecture permettent de parcourir la bibliothèque calibre directement via son support OPDS. Dans ces apps, vous pouvez accéder à l'écran du catalogue en ligne et ajouter l'adresse IP du serveur calibre pour parcourir et télécharger les livres de votre bibliothèque calibre dans l'app.

9.2.6 Comment utiliser calibre avec mon téléphone/ma tablette Android ou ma Kindle Fire ?

Il existe deux solutions pour connecter votre périphérique Android à calibre. En utilisant un câble USB – ou sans fil, grâce au wifi. La première étape pour un périphérique Android est d'installer une application de lecture de livre numérique. Il existe de nombreuses applications de lecture de livre numérique gratuites ou payantes pour Android : quelques exemples (sans ordre particulier) : FBReader[48], Moon+[49], Mantano[50], Aldiko[51], Kindle[52].

[48] https://play.google.com/store/apps/details?id=org.geometerplus.zlibrary.ui.android&hl=en
[49] https://play.google.com/store/apps/details?id=com.flyersoft.moonreader&hl=en
[50] https://play.google.com/store/apps/details?id=com.mantano.reader.android.lite&hl=en
[51] https://play.google.com/store/apps/details?id=com.aldiko.android&hl=en
[52] https://play.google.com/store/apps/details?id=com.amazon.kindle&feature=related_apps

En utilisant un cable USB

Il suffit de brancher votre périphérique à l'ordinateur avec un câble USB. calibre devrait détecter automatiquement le périphérique et alors vous pourrez transférer des livres en cliquant sur le bouton guilabel :*Envoyer au périphérique*. Notez que sous MacOS et Linux, un seul programme peut se connecter à un appareil Android à la fois. Assurez-vous donc que l'appareil n'est pas ouvert dans le gestionnaire de fichiers de l'OS, ou dans l'utilitaire de transfert de fichiers Android, etc.

Note : Avec les nouveaux appareils Android, vous pourriez avoir à faire quelques pirouettes pour réussir à faire fonctionner la connexion, puisque Google ne veut vraiment pas que vous soyez indépendant de son nuage. Tout d'abord, déverrouillez l'écran avant de brancher le câble USB. Lorsque vous branchez le câble USB, vous recevrez un popup de notification. Assurez-vous qu'il indique quelque chose comme « Transfert de fichiers multimédias » ou « MTP (mode de transfert de média) ». Si ce n'est pas le cas, tapez sur la notification et changez le mode en transfert multimédia (MTP). Vous devrez peut-être redémarrer calibre à ce stade pour que votre périphérique soit reconnu. Enfin, vous pouvez obtenir un pop-up sur l'appareil à chaque fois que calibre ou le système d'exploitation essaie réellement de s'y connecter, en demandant la permission, appuyez sur OK.

Note : Avec le Kindle Fire 8 ou plus récent, une icône s'affiche lorsque le câble USB est branché, indiquant que l'appareil est en cours de chargement. Touchez cette icône et mettez l'appareil en mode de transfert de données, puis lancez calibre, il devrait alors être détecté.

Sans fil

calibre possède un serveur web intégré, le *Le Serveur de contenu calibre* (page 117). Il rend votre collection calibre disponible sur le net. Vous pouvez la consulter sur votre appareil en utilisant un simple navigateur ou une application dédiée. Effectuez d'abord les étapes suivantes dans calibre

- Paramétrez le *Format Préféré de Sortie* de calibre à EPUB pour les périphériques Android, ou MOBI pour Kindle (Le format de sortie peut être réglé dans *Préférences → Interface → Comportement*)
- Convertir les livres que vous voulez lire sur votre périphérique au format EPUB en les sélectionnant et en cliquant sur le bouton *Convertir*.
- Activez le *Serveur de contenu* dans les préférences de calibre et laissez calibre en exécution.

Maintenant, sur votre périphérique Androïd, ouvrez le navigateur et naviguez vers

> http://192.168.1.2:8080/

Remplacez `192.168.1.2` par l'adresse IP locale de l'ordinateur qui fait tourner le serveur calibre. Voir *Le Serveur de contenu calibre* (page 117) pour plus d'informations sur l'exécution du serveur et comment trouver la bonne adresse IP à utiliser.

Vous pouvez maintenant naviguer dans votre bibliothèque de livres et télécharger les livres depuis calibre sur votre périphérique pour les ouvrir avec n'importe quel logiciel de lecture de livre numérique que vous avez préalablement installé sur votre périphérique Android.

De nombreuses applications de lecture permettent de parcourir la bibliothèque calibre directement via son support OPDS. Dans ces apps, vous pouvez accéder à l'écran du catalogue en ligne et ajouter l'adresse IP du serveur calibre pour parcourir et télécharger les livres de votre bibliothèque calibre dans l'app.

9.2.7 Puis-je accéder à mes livres calibre en utilisant le navigateur web sur mon Kindle ou sur un autre périphérique de lecture ?

Calibre dispose d'un *serveur de contenu* qui exporte les livres dans calibre en tant que page Web. Voir *Le Serveur de contenu calibre* (page 117) pour plus de détails.

Certains périphériques, comme le Kindle (1/2/DX), ne permettent pas d'accéder au port 8080 (Le port par défaut sur lequel le serveur de contenu fonctionne). Dans ce cas, remplacer le port dans les Préférences de calibre par 80. (Sur certains systèmes d'exploitation, vous pouvez ne pas être en mesure de lancer le serveur sur un numéro de port inférieur à 1024 en raison des paramètres de sécurité. Dans ce cas, la solution la plus simple est de régler votre routeur pour rediriger les requêtes sur leport 80 vers le port 8080).

De plus, le navigateur de certains périphériques n'est pas assez perfectionné pour faire fonctionner l'interface utilisée par le Serveur de contenu. Pour de tels appareils, vous pouvez simplement ajouter `/mobile` à l'URL du serveur pour obtenir une interface simplifiée et non Javascript.

9.2.8 Je ne peux pas envoyer des courriels en utilisant calibre ?

En raison de la grande quantité de spams dans les courriers électroniques, l'envoi de courriels peut être délicat, car les différents serveurs de messagerie utilisent différentes stratégies pour bloquer les courriels. Le problème est particulièrement courant si vous envoyez le courriel directement (sans relais de messagerie) par calibre. Beaucoup de serveurs (par exemple, Amazon) bloquent les courriels qui ne proviennent pas d'un relais connu. Pour configurer l'envoi de courriels dans calibre faites ce qui suit :

- Créer un compte GMX gratuit sur GMX[53].
- Aller dans *Préférences → Partager → Partager des livres par courriel*, cliquez sur le bouton *Utiliser GMX* et remplissez les informations demandées.
- Identifiez-vous sur votre compte GMX sur le site web et activez l'envoi SMTP (`Paramètres Mail->Fonctions POP3 & IMAP-> Envoyer et recevoir des emails via des programmes externes)
- calibre sera alors capable d'utiliser GMX pour envoyer le mail
- Si vous envoyez à votre Kindle, n'oubliez pas de mettre à jour les préférences de messagerie sur votre page Amazon Kindle afin d'autoriser les courriers électroniques envoyés depuis votre adresse électronique GMX. Notez également qu'Amazon n'autorise pas l'envoi par e-mail des fichiers MOBI AZW3 et du nouveau style (KF8). Enfin, Amazon a récemment commencé à envoyer des e-mails de confirmation sur lesquels vous devez cliquer pour revenir à votre compte GMX avant que le livre ne soit effectivement livré.

Même après avoir fait cela, vous pouvez avoir des problèmes. Une source fréquente de problèmes est que certains programmes antivirus, mal conçus, bloquent calibre à l'ouverture de la connexion pour envoyer les courriels. Essayez d'ajouter une exclusion pour calibre dans votre programme antivirus.

Note : Microsoft/GMX peuvent désactiver votre compte si vous l'utilisez pour envoyer de grandes quantités de courriels. Aussi, lors de l'utilisation de ces services calibre se limite automatiquement à l'envoi d'un livre toutes les cinq minutes. Si vous n'avez pas peur de risquer que votre compte soit bloqué, vous pouvez réduire cet intervalle d'attente en allant dans *Préférences → Avancé → Ajustements* dans calibre.

Note : Récemment, Google a délibérément interrompu la prise en charge de son protocole d'envoi d'e-mails (SMTP) dans le but de forcer tout le monde à utiliser son interface Web afin de pouvoir vous montrer davantage de publicités. Ils

[53] https://www.gmx.com

essaient de prétendre que le SMTP n'est pas sûr, ce qui est faux et n'est qu'une excuse. Utilisez plutôt un autre fournisseur de messagerie.

Note : Si vous êtes inquiet de donner à calibre accès à votre compte courriel, créez simplement un nouveau compte courriel gratuit avec GMX ou Outlook et employez-le seulement pour calibre.

9.2.9 Mon appareil est monté en lecture seule dans Linux, donc calibre ne peut pas s'y connecter ?

Les noyaux Linux montent les périphériques en lecture seule lorsque qu'il y a des erreurs dans leur système de fichiers. Vous pouvez les réparer en faisant :

```
sudo fsck.vfat -y /dev/sdc
```

Remplacez /dev/sdc avec le chemin pointant vers le fichier représentant votre périphérique. Vous pouvez trouver quel est le nom de fichier représentant votre périphérique, qui sera toujours dans /dev en examinant le résultat de la commande :

```
mount
```

9.2.10 Pourquoi calibre ne prend pas en charge les collections sur Kindle ou les étagères sur Nook ?

Ni Kindle ni le Nook ne fournissent de manière de manipuler des collections sur une connexion USB. Si vous vous inquiétez vraiment d'employer des collections, je vous inviterais à vendre votre Kindle/Nook et à obtenir une Kobo. Seulement Kobo semble comprendre que la vie est trop courte pour écrire des collections une par une sur un écran e-ink .

Notez que, dans le cas du Kindle, il existe un moyen pour manipuler les collections via USB, mais il faut que le Kindle soit réinitialisé à *chaque fois* qu'il est déconnecté de l'ordinateur, pour que les changements apportés aux collections soient reconnus. En tant que tel, il est peu probable que les développeurs de calibre soient assez motivés pour faire les développements nécessaires. Il y a cependant une extension calibre qui vous permet de créer des collections sur votre Kindle à partir des métadonnées calibre. Elle est disponible ici[54].

Note : Amazon a complètement supprimé la possibilité de manipuler les collections sur leurs modèles plus récents, comme le Kindle Touch et Kindle Fire, si bien que même l'extension ci-dessus serait inutile, à moins que vous ne rootez votre Kindle et installez un micrologiciel personnalisé.

[54] https://www.mobileread.com/forums/showthread.php?t=244202

9.2.11 J'obtiens un message d'erreur lorsque j'essaie d'utiliser calibre avec mon Kobo Touch/Glo/etc. ?

La Kobo a un firmware très bugué. S'y connecter est connu pour échouer aléatoirement. Certaines combinaisons de carte mère, ports USB/câbles/hubs peuvent exacerber cette tendance. Si vous obtenez une erreur lors de la connexion de votre Touch à votre bibliothèque calibre essayez ce qui suit, chacune a résolu le problème pour *certains* utilisateurs de calibre.

- Connectez la Kobo directement à votre ordinateur, et non via une Hub USB
- Essayez avec un câble USB et un port USB différents sur votre ordinateur
- Se déconnecter de la Kobo et s'y connecter à nouveau, ceci entraîne la reconstruction de la base de données, solutionnant les erreurs de base de données corrompues.
- Essayez de mettre à jour le firmware de votre Kobo Touch avec la dernière version
- Essayez de réinitialiser la Kobo (parfois cela résout le problème pendant un petit moment, mais il réapparaît. Dans ce cas, vous devez réinitialiser à plusieurs reprises)
- Essayez seulement de mettre un ou deux livres à la fois sur la Kobo et de ne pas garder de grandes quantités de livres sur la Kobo

9.2.12 Les couvertures des livres que j'envoie à ma Kindle apparaissent momentanément et sont ensuite remplacées par une couverture générique ?

Cela se produit à cause d'un bug d'Amazon. Ils essaient de télécharger une couverture pour le livre à partir de leurs serveurs et lorsque cela échoue, ils remplacent la couverture existante créée par calibre par une couverture générique. Pour plus de détails, voir ce fil de discussion du forum[55]. A partir de la version 4.17, calibre a une solution de contournement, où si vous connectez le Kindle à calibre après que les couvertures aient été détruites par Amazon, calibre les restaurera automatiquement. Donc pour voir les couvertures sur votre Kindle, vous devez :

1) Envoyez le livre à la Kindle avec calibre
2) Déconnecter la Kindle et attendre qu'Amazon détruise la couverture
3) Reconnecter la Kindle à calibre

Notez que ce contournement ne fonctionne que pour les livres envoyés avec calibre 4.17 ou ultérieur. Sinon, gardez simplement votre Kindle en mode avion, vous ne voulez pas vraiment qu'Amazon connaisse chaque livre que vous lisez de toute façon. Je vous encourage à contacter le service clientèle d'Amazon et à vous plaindre bruyamment de ce bogue. Peut-être qu'Amazon vous écoutera.

Note : Si le contournement ne fonctionne pas pour vous, assurez-vous que votre micrologiciel Kindle est au moins de la version 5.12.5, sortie en avril 2020.

---

[55] https://www.mobileread.com/forums/showthread.php?t=329945

9.2.13 Les couvertures de mes fichiers MOBI n'apparaissent plus sur mon Kindle pour PC/Kindle pour Android/iPad etc.

Ceci est causé par un bug du logiciel Amazon. Vous pouvez contourner le problème en allant dans *Préférences → Converion → Options de sortie → sortie MOBI* et mettre à oui l'option *Activer le partage du contenu du livre*. Si vous reconvertissez un livre précédemment converti, vous devrez également activer cette option dans la boîte de dialogue de conversion pour ce livre individuel (vu que par livre les paramètres sont enregistrés et ont la priorité).

Notez que cela aura pour conséquence que le MOBI généré sera affiché dans les documents personnels à la place de Books sur le Kindle Fire et que l'Amazon whispersync ne fonctionnera pas. Mais les couvertures seront présentes. C'est à vous de voir quelle fonctionnalité est la plus importante pour vous. Je vous encourage à contacter Amazon et leur demander de corriger ce bug.

Le bogue dans le logiciel d'Amazon consiste à supposer que lorsque vous mettez un fichier MOBI dans un Kindle, à moins que celui-ci ne soit marqué comme Document Personnel, vous avez acheté celui-ci chez Amazon et essaye de télécharger une imagette de la couverture à partir de ses serveurs. Quand le téléchargement échoue, il refuse d'utiliser la couverture définie dans le fichier MOBI comme solution de rechange. Ceci est probablement un comportement délibéré de la part d'Amazon afin d'essayer de forcer les auteurs à vendre par leur intermédiaire. En d'autres termes, le Kindle affiche seulement les couvertures pour les livres marqués comme Documents Personnels ou les livres achetés directement chez Amazon.

Si vous envoyez un fichier MOBI à une Kindle e-ink avec Calibre en utilisant une connexion USB, calibre pallie à ce bogue d'Amazon en téléchargeant de lui-même une vignette de la couverture. Cependant, cette solution n'est possible que lorsque une connexion USB est utilisée et le fichier envoyé avec calibre. Veuillez noter que si vous envoyez le fichier MOBI en utilisant un email, Amazon marquera automatiquement celui-ci comme Document Personnel et la couverture fonctionnera, mais le livre apparaîtra dans les Documents Personnels.

9.2.14 J'ai transféré des livres vers ma Kindle en utilisant calibre et ils n'y apparaissent pas ?

Les livres envoyés à la Kindle n'apparaissent qu'une fois qu'ils ont été *indexés* par celle-ci. Cela peut prendre un peu de temps. Si le livre n'apparait pas après un certain temps, alors il est possible que l'indexeur de la Kindle ait crashé. Parfois, un livre particulier peut causer ce crash . Malheureusement, Amazon n'a pas encore fourni de moyen de déduire quel livre causait le crash sur la Kindle. Vos seuls recours sont de soit remettre votre Kindle aux paramètres d'usine, soit de vider tout les fichiers de sa mémoire en utilisant Windows Explorer (ou n'importe quel gestionnaire de fichier que vous utilisez) et ensuite envoyer les livres à nouveaux, un par un, jusqu'à ce que vous trouviez le livre problématique. Une fois trouvé, supprimez le de la Kindle et faites une conversion MOBI vers MOBi ou MOBI vers AZW3 dans calibre et renvoyez le. Ceci résoudra probablement le problème.

9.3 Gestion de la bibliothèque

Contenu

- *Où sont stockés les fichiers des livres ?* (page 145)
- *Comment calibre gère-t-il le nom des auteurs et les tris ?* (page 145)
- *Pourquoi Calibre ne me laisse-t-il pas stocker mes livres dans ma propre structure de dossiers ?* (page 147)
- *Pourquoi calibre n'a-t-il pas telle ou telle colonne ?* (page 147)
- *Puis-je avoir une colonne indiquant le format ou le numéro ISBN ?* (page 148)

- *Comment puis-je déplacer mes données calibre d'un ordinateur à un autre ?* (page 148)
- *La liste des livres dans calibre est vide !* (page 149)
- *J'obtiens des erreurs avec ma bibliothèque calibre sur un disque réseau/NAS ?* (page 149)

9.3.1 Où sont stockés les fichiers des livres ?

Lorsque vous lancez calibre pour la première fois, il va vous être demandé de préciser un dossier dans lequel stocker vos livres. Chaque fois que vous ajouterez un livre à calibre, il va copier le livre dans ce répertoire. Les livres seront disposés dans des sous-répertories par auteur et titre dans ce répertoire général. Notez que le contenu de ce dossier est automatiquement géré par calibre. **Ne pas** ajouter de fichiers/répertoires manuellement dans ce répertoire, car ils pourraient être supprimés automatiquement. Si vous souhaitez ajouter un fichier associé à un livre précis, utilisez la flèche en haut à droite de la boîte de dialogue *Modifier les métadonnées*. calibre mettra automatiquement ce fichier dans le dossier et le déplacera lorsqu'il y aura des changements sur le titre ou l'auteur.

Les métadonnées des livres sont stockées dans le fichier `metadata.db` au plus haut niveau du répertoire de la bibliothèque. Ce fichier est une base de données SQLite. Lorsque vous sauvegardez votre bibliothèque assurez-vous de copier l'intégralité du répertoire et tous ses sous-répertoires.

Le dossier de la bibliothèque et tout son contenu constituent ce qu'on appelle une bibliothèque calibre. Vous pouvez avoir plusieurs de ces bibliothèques. Pour gérer les bibliothèques, cliquez sur l'icône calibre sur la barre d'outils. Vous pouvez créer de nouvelles bibliothèques, supprimer/renommer les bibliothèques et basculer facilement entre les bibliothèques existantes.

Vous pouvez copier ou déplacer des livres entre les différentes bibliothèques (une fois que vous avez installé plus d'une bibliothèque) en faisant un clic-droit sur le livre et en sélectionnant l'action *Copier dans la bibliothèque*.

9.3.2 Comment calibre gère-t-il le nom des auteurs et les tris ?

Les noms d'auteur sont complexes, particulièrement au travers des cultures voir la note[56] pour certaines de ces complexités. calibre a une stratégie très flexible pour gérer les noms d'auteur. La première chose à comprendre est que livres et auteurs sont des entités différentes dans calibre. Un livre peut avoir plus d'un auteur, et un auteur peut peut avoir plus d'un livre. Vous pouvez gérer les auteurs d'un livre par la boite de dialogue d'édition des métadonnées. Vous pouvez gérer les auteurs individuellement en faisant un clic-droit sur un auteur dans le Navigateur d'étiquettes sur la droite de la fenêtre principale de calibre et en sélectionnant *Gérer les auteurs*. En utilisant cette boîte de dialogue vous pouvez changer le nom d'un auteur et également comment ce nom est trié. Ceci changera automatiquement le nom d'un auteur dans tous les livres de cet auteur. Quand un livre a de multiples auteurs, séparer leurs noms en utilisant le caractère &.

Maintenant passons au tri par nom d'auteur :

- Quand un nouvel auteur est ajouté à calibre (ce qui arrive quand un livre écrit par un nouvel auteur est ajouté), calibre calcule automatiquement une chaîne de tri pour le livre et pour l'auteur.
- Les auteurs dans le Navigateur d'étiquettes sont triés selon la valeur de tri pour les **auteurs**. Rappelez-vous que ceci est différent du champ Tri par Auteur d'un livre.
- Par défaut, cet algorithme de tri suppose que le nom de l'auteur est au format `Prénom Nom` et génère un valeur de tri `Nom, Prénom`.
- Vous pouvez modifier cet algorithme en allant dans *Préférences → Avancé → Ajustements* et en paramétrant l'ajustement *author_sort_copy_method*.

[56] https://www.w3.org/International/questions/qa-personal-names.fr

- Vous pouvez forcer calibre à recalculer les valeurs de tri d'auteur pour chaque auteur en cliquant droit sur n'importe quel auteur et en sélectionnant *Gérer Auteurs*, puis en cliquant sur le bouton *Recalculer toutes les valeurs de tri par auteur*. Faites cette action, après avoir défini l'ajustement author_sort_copy_method à ce que vous souhaitez .
- Vous pouvez forcer calibre à recalculer les valeurs de tri d'auteur pour tous les livres en utilisant la boîte de dialogue Modifier les métadonnées par lot (sélectionner tous les livres et cliquez sur Modifier les métadonnées. Ensuite, cochez la case *Effectuer un tri automatique par auteur*, puis cliquez sur OK.)
- Quand les valeurs de tri de l'auteur sont recalculées pour les livres , calibre utilise les valeurs de tri pour chaque auteur individuel. Par conséquent, assurez-vous que les valeurs de tri individuelles par auteur sont correctes avant de recalculer les valeurs de tri de l'auteur pour les livres.
- Vous pouvez contrôler de quelle manière le Navigateur d'étiquettes va afficher les auteurs en utilisant leur nom ou en se basant sur leurs valeurs de tri en paramétrant l'ajustement *categories_use_field_for_author_name* dans *Préférences → Avancé → Ajustements*.

Notez que vous pouvez définir la valeur de tri d'un auteur individuellement de la manière que vous désirez en utilisant *Gérer auteurs*. Cela peut être utile lorsque calibre ne le fait pas correctement,sur les noms composés complexes comme Miguel de Cervantes Saavedra ou lorsqu'il s'agit de noms asiatiques comme Sun Tzu.

Avec toute cette flexibilité, il est possible de voir calibre gérer vos noms d'auteurs exactement comme vous le souhaitez. Prenons comme exemple, une demande habituelle : que calibre trie les auteurs par Nom, Prénom. Pour ce faire, et si la note qui se trouve ci-dessous ne s'applique pas à vous, alors :

- Paramétrez l'ajustement `author_sort_copy_method` à `copy` comme décrit ci-dessus.
- Redémarrez calibre. Ne pas modifier les métadonnées de livre avant de faire les étapes restantes.
- Changer tous les noms d'auteurs à Nom, Prénom en utilisant la boîte de dialogue Gérer auteurs.
- Après avoir changé tous les auteurs, cliquez sur le bouton *Recalculer toutes les valeurs de tri par auteur*.
- Appuyez sur OK, à ce moment calibre va modifier les auteurs dans tous vos livres. Cela peut prendre un certain temps.

Note :

Lors du changement depuis Prénom Nom à Nom, Prénom, il est souvent le cas que les valeurs author_sort sont déjà au format Nom, Prénom. Si c'est le cas, procédez comme suit :

- Paramétrez l'ajustement `author_sort_copy_method` à `copy` comme décrit ci-dessus.
- Redémarrez calibre. Ne pas modifier les métadonnées de livre avant de faire les étapes restantes.
- Ouvrir la boîte de dialogue Gérer Auteurs. Cliquer sur le bouton `Copier toutes les valeurs de tri d'auteur dans auteur`.
- Vérifiez les auteurs pour être sûr du résultat. Vous pouvez toujours cliquer sur Annuler pour abandonner les modifications. Une fois que vous aurez cliqué sur OK, il n'y aura plus d'annulation possible.
- Appuyez sur OK, à ce moment calibre va modifier les auteurs dans tous vos livres. Cela peut prendre un certain temps.

9.3.3 Pourquoi Calibre ne me laisse-t-il pas stocker mes livres dans ma propre structure de dossiers ?

Le point essentiel de la gestion de la bibliothèque calibre, c'est qu'il fournit une interface de recherche et de tri pour localiser les livres beaucoup plus efficace que n'importe lequel des système de répertoire que vous pourriez concevoir pour votre collection. En effet, une fois que vous serez à l'aise avec l'interface de calibre pour trouver, trier et parcourir votre bibliothèque, vous ne ressentirez plus le besoin de faire la chasse à vos livres sur votre disque dur pour les retrouver. En gérant les livres selon sa propre structure de dossier Auteur -> Titre -> Fichiers du livre, calibre est capable d'atteindre un haut niveau de fiabilité et de standardisation. Pour illustrer pourquoi une interface de recherche/étiquetage est supérieure à la gestion des dossiers, considérez ce qui suit. Supposons que votre collection de livre est bien classée dans des dossiers avec le schéma suivant :

```
Genre -> Author -> Series -> ReadStatus
```

Dans ce schéma, il est très facile de trouver, par exemple, tous les livres de science-fiction écrits par Isaac Asimov dans la série Fondation. Mais si vous voulez trouver tous les livres de science-fiction - non lus, il n'y a aucun moyen simple de le faire avec ce schéma de dossier. Il aurait fallu choisir un schéma qui ressemble à :

```
ReadStatus -> Genre -> Author -> Series
```

Dans calibre, vous pouvez utiliser des étiquettes pour indiquer le genre et le statut de lecture et ensuite simplement utiliser une requête de recherche basique comme `tag:scifi and not tag:read`. calibre a même une interface graphique agréable : vous n'avez pas à apprendre son langage de recherche mais simplement à cliquer sur les étiquettes pour les inclure ou exclure de la recherche.

Pour ceux d'entre vous qui réclament avoir besoin d'accéder au système de fichiers pour récupérer vos livres à partir du réseau, calibre a un excellent Serveur de contenu qui vous donne accès à votre bibliothèque calibre depuis le net.

Si vous craignez que calibre cesse un jour d'être développé et laisse tous vos livres en rade, coincés dans sa structure de dossiers, explorez la puissante fonctionnalité : **:quilabel :`Enregistrer sur le disque`** dans calibre qui vous permet d'exporter tous vos fichiers dans une arborescence de dossiers d'une complexité quelconque fondée sur leurs métadonnées.

Enfin, la raison pour laquelle il y a des chiffres à la fin de chaque dossier de titre, est la *robustesse*. Ce nombre est le numéro d'identification de l'enregistrement du livre dans la base de données calibre. La présence du numéro vous permet d'avoir plusieurs enregistrements avec le même titre et noms d'auteur. C'est aussi une partie de ce qui permet à calibre de régénérer magiquement la base de données avec toutes les métadonnées si le fichier de base de données est corrompu. Étant donné que la mission de calibre est de vous amener à ne plus stocker les métadonnées dans les noms de fichiers et d'arrêter d'utiliser le système de fichiers pour trouver des choses, la robustesse accrue offerte par les numéros d'identification vaut bien des noms de dossiers plus laids.

Si vous n'êtes toujours pas convaincu, alors j'ai bien peur que calibre ne soit pas pour vous. cherchez ailleurs pour vos besoins catalogage de livres. Pour que tout soit clair, nous ne changerons pas notre méthode. Inutile de nous contacter pour tenter de nous faire changer d'avis.

9.3.4 Pourquoi calibre n'a-t-il pas telle ou telle colonne ?

calibre est conçu pour avoir des colonnes pour les champs les plus fréquemment et largement utilisés. En outre, vous pouvez ajouter toutes les colonnes que vous désirez. Les colonnes peuvent être ajoutées via *Préférences → Interface → Ajout de colonnes personnalisées*. Regardez le tutoriel UI Power tips[57] pour apprendre à créer vos propres colonnes, ou lisez ce poste de blog[58].

Vous pouvez également créer des « colonnes virtuelles » qui contiennent des combinaisons de métadonnées à partir d'autres colonnes. Dans la boîte de dialogue ajouter une colonne utilisez les liens *Création rapide* pour créer facilement des colonnes

[57] https://calibre-ebook.com/fr/demo#tutorials
[58] https://blog.calibre-ebook.com/calibre-custom-columns/

pour montrer l'ISBN du livre ou les formats. Vous pouvez utiliser le puissant langage de modèle pour faire beaucoup plus avec les colonnes. Pour plus de détails, voir *Le langage de modèle calibre* (page 169).

9.3.5 Puis-je avoir une colonne indiquant le format ou le numéro ISBN ?

Oui, vous le pouvez. Suivez les instructions de la réponse ci-dessus pour ajouter des colonnes personnalisées.

9.3.6 Comment puis-je déplacer mes données calibre d'un ordinateur à un autre ?

Vous pouvez exporter toutes les données calibre (livres, paramètres et extensions) et les importer alors sur un autre ordinateur. Tout d'abord voyons comment exporter les données :

- Faites un clic droit sur l'icône calibre dans la barre d'outils principale et sélectionnez *Exporter/Importer toutes les données calibre*. Notez que si un périphérique est actuellement connecté, cette option du menu ne sera pas disponible - donc, déconnectez tout périphérique connecté. Cliquez alors sur le bouton étiqueté *Exporter toutes vos données calibre*. Vous verrez une liste de toutes vos bibliothèques calibre. Cliquez OK et choisissez un répertoire vide quelque part sur votre ordinateur. Les données exportées seront sauvegardées dans ce répertoire. Copiez simplement ce répertoire vers votre nouvel ordinateur et suivez les instructions ci-dessous pour importer les données.
- Installez calibre sur votre nouvel ordinateur et passez en revue *l'Assistant de bienvenue*, ce que vous faites là n'a pas d'importance, vu que vous importerez vos anciens paramètres à l'étape suivante. Vous aurez alors un calibre vide, avec uniquement le guide *Getting Started* dans votre bibliothèque. Encore une fois, faites un clic droit sur le bouton calibre et choisissez *Exporter/Importer toutes les données calibre*. Cliquez alors le bouton étiqueté *Importer les données précédemment exportées*. Sélectionnez le répertoire avec les données exportées que vous avez copiées plus tôt. Vous aurez alors une liste des bibliothèques que vous pouvez importer. Passer la liste en revue une par une, et sélectionnez le nouvel emplacement pour chaque bibliothèque (un emplacement est simplement un répertoire vide sur votre ordinateur). Cliquez OK. Après l'accomplissement de l'importation, calibre redémarrera, avec toutes vos anciennes bibliothèques, vos paramètres et vos extensions calibre.

Note : Cette fonctionnalité Importer/Exporter est seulement disponible à partir de la version 2.47 et ultérieures. Si vous avez une ancienne version de calibre, ou si vous rencontrez des problèmes avec l'importation/exportation vous pouvez simplement copier votre répertoire de bibliothèque calibre manuellement, comme décrit dans le paragraphe suivant.

Copier simplement le répertoire de la bibliothèque calibre depuis l'ancien vers le nouvel ordinateur. Vous pouvez trouver quel est le répertoire de la bibliothèque en cliquant sur l'icône calibre dans la barre d'outils et vous verrez le chemin vers la bibliothèque calibre actuelle en haut de la fenêtre.

Sur le nouvel ordinateur, démarrez calibre pour la première fois. Il va exécuter *l'Assistant de bienvenue* qui vous demandera de spécifier l'emplacement de la bibliothèque de calibre. Sélectionnez le dossier précédemment copié. Si l'ordinateur où vous avez transféré la bibliothèque a déjà une installation de calibre, *l'Assistant de bienvenue* ne se lancera pas. Dans ce cas, faites un clic droit sur l'icône calibre de la barre d'outils et sélectionnez le nouveau dossier copié. Vous aurez maintenant deux bibliothèques calibre sur votre ordinateur et vous pouvez basculer de l'une à l'autre en cliquant sur l'icône calibre dans la barre d'outils. Le transfert de votre bibliothèque de cette façon préservera toutes vos métadonnées, les étiquettes, les colonnes personnalisées, etc.

9.3.7 La liste des livres dans calibre est vide !

Afin de comprendre pourquoi cela s'est produit, il faut comprendre ce qu'est une bibliothèque calibre. Au niveau le plus élémentaire, une bibliothèque calibre est juste un répertoire. Chaque fois que vous ajoutez un livre à calibre, les fichiers de ce livre sont copiés dans ce répertoire (subdivisé en sous-répertoires par auteur et par titre). A la racine du répertoire contenant la bibliothèque calibre, vous verrez un fichier appelé metadata.db. Ce fichier stocke les métadonnées comme le titre/auteur/cote/étiquettes, etc pour *tous* les livres de votre bibliothèque calibre. La liste des livresque calibre affiche est crée par la lecture du contenu de ce fichier metadata.db.

Il peut y avoir deux raisons pour lesquelles calibre affiche une liste vide de livres :

- Votre dossier de bibliothèque calibre a changé d'emplacement. Cela peut arriver s'il était sur un disque externe et que la lettre de lecteur pour ce disque a changé. Ou si vous avez accidentellement déplacé le dossier. Dans ce cas, calibre ne peut plus trouver sa bibliothèque et démarre avec une bibliothèque vide à la place. Pour y remédier, faites un clic-droit sur l'icône calibre dans la barre d'outils et sélectionnez Changer/créer/déplacer une bibliothèque. Cliquez sur la petite icône bleue pour sélectionner le nouvel emplacement de votre bibliothèque calibre et cliquez sur OK. Si vous ne connaissez pas le nouvel emplacement rechercher le fichier `metadata.db` sur votre ordinateur.
- Votre fichier metadata.db a été supprimé/endommagé. Dans ce cas, vous pouvez demander à calibre de restaurer le fichier metadata.db à partir de ses sauvegardes. Faites un clic droit sur l'icône calibre dans la barre d'outils et sélectionnez Maintenance de la bibliothèque->Restaurer la base de données. Calibre va restaurer automatiquement metadata.db.

9.3.8 J'obtiens des erreurs avec ma bibliothèque calibre sur un disque réseau/NAS ?

Ne mettez pas votre bibliothèque calibre sur un disque réseau.

Un système de fichiers est une bête complexe. La plupart des systèmes de fichiers réseau ne comprennent pas les diverses caractéristiques du système de fichier utilisé par calibre. Certains ne supportent pas le verrouillage des fichiers, d'autres ne supportent pas le hardlinking. En outre, calibre est une application mono-utilisateur. Si vous exécutez accidentellement deux copies de calibre sur la même bibliothèque en réseau, des incidents peuvent se produire. Enfin, différents systèmes d'exploitation imposent des limitations sur le système de fichiers, donc si vous partagez votre lecteur réseau à travers plusieurs systèmes d'exploitation, une fois de plus, des incidents *se produiront*.

Pensez à utiliser le Serveur de contenu de calibre pour rendre les livres disponibles sur d'autres ordinateurs. Lancez calibre sur un seul ordinateur et accédez-y via le Serveur de contenu ou via une solution de Bureau à Distance.

Si vous devez partager la bibliothèque actuelle, utilisez un outil de synchronisation de fichiers comme Dropbox ou rsync plutôt qu'un disque réseau. Si vous utilisez un outil de synchronisation de fichiers, il est **essentiel** que vous vous assuriez que calibre et l'outil de synchronisation de fichiers ne cherchent pas à accéder à la bibliothèque en même temps. En d'autres termes, **n'exécuter pas** l'outil de synchronisation de fichiers et calibre au même moment.

Même avec ces outils il y a un risque de corruption/de perte de données. Ne le faire que si vous êtes prêt à vivre avec ce risque. En particulier, sachez que **Google Drive** est incompatible avec calibre : si vous mettez votre bibliothèque calibre dans Google Drive, **vous subirez une perte de données**. Voir ce poste[59] pour des détails.

[59] https://www.mobileread.com/forums/showthread.php?t=205581

9.4 Divers

Contenu

- *Amazon arrête la livraison par e-mail des fichiers MOBI ?* (page 151)
- *Je veux que calibre télécharge les actualités de mon site préféré.* (page 151)
- *Pourquoi le nom calibre ?* (page 151)
- *Pourquoi calibre ne montre qu'une partie de mes polices sous macOS ?* (page 152)
- *calibre ne démarre pas sous Windows ?* (page 152)
- *calibre se bloque/plante occasionnellement ?* (page 153)
- *Les outils Visionneuse de livres numériques et Édition de livres de calibre ne fonctionnent pas sous Windows ?* (page 153)
- *Utiliser la visionneuse ou faire n'importe quelle conversion cause une erreur de permission refusée dans Windows* (page 154)
- *Calibre crash/ ne démarre pas sous macOS ?* (page 154)
- *Je n'obtiens qu'un écran noir ou blanc lorsque je lance la visionneuse de livres numériques calibre ?* (page 154)
- *J'ai téléchargé le programme d'installation mais il ne fonctionne pas ?* (page 155)
- *Mon antivirus affirme que calibre est un virus/cheval de troie ?* (page 155)
- *Comment faire une sauvegarde de calibre ?* (page 156)
- *Comment puis-je mettre des livres numériques EPUB achetés avec calibre (ou que dois-je faire avec les fichiers .acsm) ?* (page 156)
- *J'ai une erreur "accès refusé" ?* (page 156)
- *Puis-je avoir la métadonnée commentaire apparaître sur ma liseuse ?* (page 157)
- *Comment obtenir que calibre utilise mon proxy HTTP ?* (page 157)
- *Je voudrais certaines fonctionnalités ajoutées à calibre. Que puis-je faire ?* (page 157)
- *Pourquoi calibre n'a pas de mise à jour automatique ?* (page 157)
- *Quelle est la licence de calibre ?* (page 158)
- *Comment exécuter calibre depuis ma clé USB ?* (page 158)
- *Comment faire fonctionner des éléments de calibre comme le téléchargement d'actualités et le serveur de contenu sur mon propre serveur Linux ?* (page 158)

9.4.1 Amazon arrête la livraison par e-mail des fichiers MOBI ?

Amazon a annoncé[60] qu'ils n'accepteront plus les fichiers MOBI envoyés par courriel aux adresses `@kindle.com`. Vous pouvez demander à calibre d'envoyer EPUB au lieu de MOBI en allant dans *Préférences → Partage de livres par e-mail* puis en supprimant MOBI de la liste des formats à envoyer à votre adresse e-mail `@kindle.com` et en ajoutant EPUB à la place.

Noter cependant que le système de prise en charge EPUB d'Amazon est très imparfait, et qu'il rejettera un certain nombre de fichiers EPUB qui fonctionnent partout ailleurs. Dans ce cas, vous pouvez essayer l'astuce suivante :

1. Convertir le fichier EPUB en MOBI
2. Convertir ensuite le fichier MOBI en EPUB et envoyer le fichier EPUB résultant

Cela supprimera tout le formatage avancé, les polices intégrées, etc., mais augmentera considérablement les chances qu'Amazon accepte l'EPUB.

Note : Si vous utilisiez auparavant la livraison par courriel des périodiques téléchargés par calibre, il est préférable de les envoyer par câble USB ou de les télécharger depuis le serveur de contenu de calibre via le navigateur intégré du Kindle. Toutefois, si vous souhaitez continuer à utiliser la livraison par courriel, vous pouvez essayer de changer le format de sortie dans Préférences->Comportement en EPUB, calibre téléchargera alors les nouvelles au format EPUB. Quant à savoir si Amazon acceptera ou non le format EPUB, c'est une toute autre question.

9.4.2 Je veux que calibre télécharge les actualités de mon site préféré.

Si vous êtes un minimum compétent en informatique, vous pouvez indiquer à calibre comment télécharger des actualités à partir de n'importe quel site de votre choix. Pour apprendre à le faire, voir *Ajouter votre site web favori d'actualités* (page 33).

Sinon, vous pouvez demander un site d'actualités en particulier en postant dans le calibre Recipes forum[61].

9.4.3 Pourquoi le nom calibre ?

Faites votre choix :

- Converter And LIBRary for E-books
- Un produit de haut *calibre*
- Un hommage au Librie de SONY qui a été le premier lecteur de livre numérique e-ink
- Ma femme l'a choisi ;-)

Calibre se prononce cal-i-ber *pas* ca-li-bre. Si ça vous interroge, calibre est l'orthographe britannique / du Commonwealth pour caliber. En tant qu'Indien, c'était l'orthographe la plus naturelle pour moi.

[60] https://blog.the-ebook-reader.com/2022/05/03/amazon-dropping-mobi-support-on-send-to-kindle-apps/
[61] https://www.mobileread.com/forums/forumdisplay.php?f=228

9.4.4 Pourquoi calibre ne montre qu'une partie de mes polices sous macOS ?

calibre incorpore les polices dans les fichiers de livre numérique qu'il crée. Les fichiers de livre numérique gèrent uniquement les polices TrueType et Open Type (.ttf et .otf) incorporées. La plupart des polices sur les systèmes macOS sont au format .dfont, donc elles ne peuvent pas être incorporées. calibre affiche uniquement les polices TrueType et Open Type trouvées sur votre système. Vous pouvez obtenir beaucoup de telles polices sur le web. Il vous suffit de télécharger les fichiers .ttf/.otf et de les ajouter au dossier Library/Fonts dans votre dossier principal.

9.4.5 calibre ne démarre pas sous Windows ?

Il peut y avoir plusieurs raisons à cela :

- Si vous n'obtenez aucune erreur mais que la fenêtre de calibre n'apparaît pas, elle est probablement apparue hors de l'écran. Vous pouvez rassembler toutes les fenêtres sur l'écran actuel à l'aide de l'une des techniques décrites ci-dessous ici[62].
- Si vous obtenez une erreur sur calibre indiquant que vous n'êtes pas en mesure d'ouvrir un fichier car ce fichier est utilisé par un autre programme, procédez comme suit :
  - Désinstallez calibre
  - Redémarrer votre ordinateur
  - Réinstallez calibre. Mais ne démarrez pas Calibre à partir de l'assistant d'installation.
  - Désactivez temporairement votre programme antivirus (déconnecter Internet au préalable, pour être sûr)
  - Regardez dans le dossier défini comme étant votre bibliothèque calibre. Si vous voyez un fichier nommé metadata.db, supprimez-le.
  - Démarrez calibre
  - A partir de là, vous devriez être en mesure de démarrer calibre normalement.
- Si vous obtenez une erreur sur une fonction Python s'interrompant soudainement après la mise à jour de calibre, tout d'abord désinstallez calibre puis supprimez les répertoires C :Program FilesCalibre et C :Program FilesCalibre2 (s'ils existent). Ensuite, ré-installez et ça devrait fonctionner.
- Si vous obtenez une erreur dans *l'Assistant de bienvenue* à l'ouverture initiale de calibre, essayez de choisir un dossier comme :file ;`C :bibliothèque` comme bibliothèque de calibre (calibre a parfois du mal avec les emplacements de bibliothèque si le chemin contient des caractères non-anglais, ou seulement des chiffres, etc).
- Essayez de l'exécuter en tant qu'administrateur (Cliquez avec le bouton droit de la souris sur l'icône et sélectionnez *Exécuter en tant qu'administrateur*)

Si Calibre ne démarre toujours pas, lancez une invite de commande (touche Windows et R, puis tapez *cmd.exe* dans la boîte de dialogue Exécuter qui apparaît). Dans l'invite de commande tapez la commande suivante et appuyez sur Entrée :

```
calibre-debug -g
```

Post any output you see in a help message on the Forum[63].

[62] https://www.wikihow.com/Bring-an-Off-Screen-Window-Back-on-Windows
[63] https://www.mobileread.com/forums/forumdisplay.php?f=166

9.4.6 calibre se bloque/plante occasionnellement ?

Il y a plusieurs choses possibles que je sais, qui peuvent causer cela :

- Vous avez récemment connecté un moniteur externe ou un téléviseur à votre ordinateur. Dans ce cas, chaque fois que calibre ouvre une nouvelle fenêtre comme la fenêtre Modifier les métadonnées ou la boîte de dialogue de Convertir, elle apparaît sur l'autre moniteur et vous ne le remarquez pas et vous pensez donc que calibre est bloqué. Déconnectez votre second moniteur et redémarrez calibre.
- Les programmes suivants ont été signalés comme causant des plantages dans calibre : Si vous exécutez l'un de ceux-là, fermez-le avant de démarrer calibre, ou désinstallez-le : *RoboForm*, *Logitech SetPoint Settings*, *Constant Guard Protection by Xfinity*, *Spybot*, *Killer Network Manager*, *Nahimic UI Interface*, *Acronis True Image*.
- Vous utilisez une souris USB/une tablette de la marque Wacom. Il existe une incompatibilité entre les pilotes Wacom et la boite à outils graphique utilisée par calibre. Essayez avec une souris d'une autre marque.
- Sur certaines versions 64 bit de Windows il existe des logiciels/paramètres de sécurité qui empêchent calibre 64 bit de fonctionner correctement. Si vous utilisez la version 64 bit de calibre essayez de passer à la version 32-bit.
- Si le plantage survient lorsque vous essayez de copier du texte à partir de la Visionneuse de livre numérique calibre, cela est du le plus souvent à une application surveillant/gérant votre presse papier que vous utilisez. Arrêtez-la et tout rentrera dans l'ordre.
- Si le plantage survient spécifiquement quand vous utilisez le dialogue de fichier, comme cliquer sur le bouton *Ajouter des livres* ou le bouton *Enregistrer sur le disque*, alors vous avez certains programmes qui ont installé des extensions Shell endommagées sur votre ordinateur. Les responsables connus comprennent : *SpiderOak*, *odrive sync* et *Dell Backup and Recovery* et *Netdrive*. Si vous avez l'un de ceux-là, désinstallez-les et tout ira bien. Vous pouvez également utiliser l'utilitaire NirSoft Shell Extension Viewer[64] pour voir quelles extensions Shell sont installées sur votre système et les désactiver individuellement, si vous ne voulez pas complètement désinstaller le programme. Rappelez-vous d'utiliser « Redémarrer Explorer » ou redémarrer votre ordinateur après avoir désactivé les extensions Shell.

Si aucun des points répertoriés ci-dessus ne s'appliquent à vous, alors il y a un autre programme sur votre ordinateur qui interfère avec calibre. Tout d'abord redémarrez votre ordinateur en mode sans échec, afin d'avoir aussi peu de programmes que possible en exécution, et voyez si le programme démarre. Si c'est le cas, alors vous savez que c'est un programme qui cause le problème. La cause la plus probable est un programme qui modifie le comportement d'autres programmes, tels qu'un antivirus, un pilote de périphérique, quelque chose comme RoboForm (une application de remplissage automatique de formulaires) ou une aide technique comme la Synthèse Vocale ou un Lecteur d'Ecran.

La seule façon de trouver le coupable est d'éliminer les programmes un par un et voir lequel est la cause du problème. Essentiellement, arrêtez un programme, exécutez calibre, vérifiez les plantages. S'ils se produisent encore, arrêtez un autre programme et ainsi de suite.

9.4.7 Les outils Visionneuse de livres numériques et Édition de livres de calibre ne fonctionnent pas sous Windows ?

Ces deux programmes utilisent l'accélération matérielle car ils intègrent une version du navigateur Chrome pour rendre le HTML. S'ils ne fonctionnent pas, ce sera en raison d'une incompatibilité avec les pilotes GPU (graphiques) de votre système. Essayez d'abord de les mettre à jour, puis redémarrez. Si cela ne résout pas le problème, vous pouvez définir la variable d'environnement « QTWEBENGINE_CHROMIUM_FLAGS » à la valeur « -disable-gpu » pour désactiver l'accélération matérielle. Voir cette page[65] pour plus de détails.

[64] https://www.nirsoft.net/utils/shexview.html
[65] https://doc.qt.io/qt-6/qtwebengine-debugging.html

9.4.8 Utiliser la visionneuse ou faire n'importe quelle conversion cause une erreur de permission refusée dans Windows

Quelque chose sur votre ordinateur empêche calibre d'accéder à ses propres fichiers temporaires. Le plus souvent les permissions sur le répertoire `Temp` sont incorrectes. Allez au répertoire file :*C :\Users\USERNAME\AppData\Local* dans l'Explorateur de Windows et faites alors un clic droit sur le répertoire file :*Temp*, sélectionnez *Propriétés* et allez à l'onglet :guilabel ;`Sécurité`. Assurez-vous que votre compte utilisateur à le contrôle complet sur ce répertoire.

Certains utilisateurs ont rapporté qu'exécuter la commande suivante dans une invite de commande Administrateur répare leurs permissions. Pour obtenir une Invite de Commande Administrateur, cherchez après cmd.exe dans le menu démarrer, faites alors un clic droit sur l'entrée de l'invite de commande et sélectionner *Exécuter en tant qu'administrateur*. Dans l'Invite de commande entrer la commande suivante et appuyer Enter :

```
icacls "%appdata%\..\Local\Temp" /reset /T
```

Alternativement, vous pouvez exécuter calibre en tant qu'Administrateur, mais faire cela entraînera que certaines fonctionnalités, telle que le glisser et déposer de, de ne pas fonctionner.

Finalement, certains utilisateurs ont rapporté que désactiver l'UAC résout le problème.

9.4.9 Calibre crash/ ne démarre pas sous macOS ?

Une cause fréquente de pannes sur macOS est l'utilisation de technologies d'accessibilité qui sont incompatibles avec la boite à outils graphique utilisée par calibre. Essayez de désactiver VoiceOver si vous l'avez. Puis allez sur Préférences Système->Système->Accès Universel et désactivez-le paramètre pour autoriser l'accès aux périphériques assistés dans tous les onglets. Une autre raison peut être une application tierce modifiant l'apparence du système, comme Smart Scroll

Vous pouvez obtenir des informations de débogage sur les incidents calibre qui ne démarre pas en lançant *Console.app*. Les informations de débogage y seront transcrites. Si ces informations contiennent une ligne ressemblant à :

```
Qt: internal: -108: Error ATSUMeasureTextImage text/qfontengine_mac.mm
```

alors le problème est probablement dû à un cache de police endommagé. Vous pouvez effacer le cache en suivant ces instructions[66]. Si ça ne le résout pas, cherchez un fichier de police corrompu sur votre système, dans ~/Library/Fonts ou quelque chose qui y ressemble. Un moyen facile de vérifier les polices corrompues dans macOS est de lancer l'application « Font Book », sélectionnez toutes les polices, puis dans le menu Fichier, choisissez « Valider les polices ».

9.4.10 Je n'obtiens qu'un écran noir ou blanc lorsque je lance la visionneuse de livres numériques calibre ?

Cela est dû à une incompatibilité entre Qt WebEngine, que la visionneuse utilise pour effectuer le rendu, et les pilotes de GPU de votre système. Essayez d'abord de mettre à niveau les pilotes de GPU. Si cela n'aide pas, vous pouvez essayer de désactiver l'accélération matérielle dans Qt WebEngine en définissant la variable d'environnement `QTWEBENGINE_CHROMIUM_FLAGS` à la valeur `--disable-gpu`. Voir *Variables d'environnement* (page 301) pour savoir comment modifier les variables d'environnement.

[66] https://www.macworld.com/article/1139383/fontcacheclear.html

9.4.11 J'ai téléchargé le programme d'installation mais il ne fonctionne pas ?

Le téléchargement depuis Internet peut parfois aboutir à un résultat corrompu. Si le programme d'installation de calibre que vous avez téléchargé ne s'ouvre pas, essayez de le télécharger à nouveau. Si le nouveau téléchargement ne fonctionne pas, téléchargez-le à partir d'un autre site <https://github.com/kovidgoyal/calibre/releases/latest>`_. Si le programme d'installation ne fonctionne toujours pas, alors quelque chose sur votre ordinateur empêche son exécution.

- Essayez de désactiver temporairement votre programme antivirus (Microsoft Security Essentials, ou Kaspersky ou McAfee ou Norton ou autre). C'est probablement le coupable si le processus de mise à jour est suspendu en cours de lancement.
- De même, si le programme d'installation ne fonctionne pas/revient en arrière et que vous avez Microsoft PowerToys en cours d'exécution, quittez-le.
- Essayez de redémarrer votre ordinateur et exécutez un nettoyeur de registre comme Wise registry cleaner[67].
- Essayez une installation propre, C'est-à-dire, désinstallez calibre, supprimez `C:\Program Files\Calibre2` (ou quelque soit l'endroit où vous avec choisi d'installer calibre précdemment) Réinstallez alors calibre. Notez que la désinstallation ne touche pas à vos livres ou à vos paramètres.
- Essayez de télécharger le programme d'installation avec un autre navigateur. Par exemple, si vous utilisez Microsoft Edge, essayez d'utiliser Firefox ou Chrome à la place.
- Si vous obtenez une erreur sur une DLL manquante sur Windows alors, très probablement, les autorisations sur le dossier temporaire sont incorrectes. Allez dans le dossier `C:\Users\USERNAME\AppData\Local` dans votre Explorateur Windows, puis faites un clic droit sur le dossier `Temp`, puis sélectionnez *Propriétés* et cliquez sur l'onglet *Sécurité*. Assurez-vous que votre compte utilisateur a le contrôle total sur ce dossier.

Si vous ne parvenez toujours pas à démarrer le programme d'installation et que vous êtes sous Windows, vous pouvez utiliser l'installateur de calibre portable[68], qui n'a pas besoin d'un programme d'installation (c'est juste un fichier ZIP).

9.4.12 Mon antivirus affirme que calibre est un virus/cheval de troie ?

La première chose à vérifier est que vous avez téléchargé calibre sur le site officiel[69]. Assurez-vous que vous cliquez les liens de téléchargement sur la gauche, pas les publicités sur la droite. calibre est un programme très populaire et des gens peu scrupuleux peuvent offrir la possibilité de le télécharger sur des sites Web non recommandables.

Si vous avez le téléchargement officiel et que votre antivirus prétend toujours que calibre est un virus, alors, votre programme antivirus fait une erreur. Les programmes antivirus utilisent des technologies heuristiques, les modèles de code qui semblent suspicieux pour détecter les virus. C'est un peu comme le profilage racial. calibre est un produit entièrement open source. Vous pouvez réellement parcourir le code source vous-même (ou le faire faire par quelqu'un d'autre) pour vérifier que ce n'est pas un virus. Merci de signaler la faute d'identification à la companie où vous avez acheté votre logiciel antivirus. Si le programme antivirus vous empêche de télécharger/installer calibre, désactivez-le temporairement, installez calibre, puis réactivez-le.

[67] https://www.wisecleaner.com
[68] https://calibre-ebook.com/fr/download_portable
[69] https://calibre-ebook.com/fr/download

9.4.13 Comment faire une sauvegarde de calibre ?

L'élément le plus important à sauvegarder est le dossier bibliothèque de calibre, qui contient tous vos livres et métadonnées. C'est le dossier que vous avez choisi pour votre bibliothèque calibre lors de l'exécution de calibre pour la première fois. Vous pouvez obtenir le chemin d'accès au dossier de la bibliothèque en cliquant sur l'icône calibre sur la barre d'outils principale. Vous devez sauvegarder ce dossier complet avec tous ses fichiers et sous-dossiers.

Vous pouvez basculer calibre pour utiliser une bibliothèque sauvegardée en cliquant simplement sur l'icône calibre sur la barre d'outils et en sélectionnant le répertoire de sauvegarde de votre bibliothèque. Un dossier de sauvegarde de la bibliothèque conserve vos colonnes personnalisées, les recherches enregistrées ainsi que tous vos livres et métadonnées.

Si vous souhaitez sauvegarder les extensions de configuration calibre, vous devez sauvegarder le dossier config. Vous pouvez trouver ce dossier config via *Préférences → Divers*. Notez que la restauration des dossiers de configuration n'est pas officiellement supportée, mais devrait fonctionner dans la plupart des cas. Il suffit de copier le contenu du dossier de sauvegarde dans le répertoire de configuration à restaurer.

9.4.14 Comment puis-je mettre des livres numériques EPUB achetés avec calibre (ou que dois-je faire avec les fichiers .acsm) ?

La plupart des livres EPUB achetés ont un *DRM* (page 389). Ceci empêche calibre de les ouvrir. Vous avez toujours la possibilité d'utiliser calibre pour stocker et transférer les livres sur votre liseuse. Tout d'abord, vous devez autoriser votre liseuse sur Windows avec le logiciel Adobe Digital Editions. Une fois cela fait, les livres EPUB transférés avec calibre fonctionneront sur votre liseuse. Lorsque vous achetez un livre EPUB sur un site Web, vous obtiendrez un fichier *.acsm*. Ce fichier doit être ouvert avec Adobe Digital Editions, qui téléchargera ensuite le livre sous format *.ePub*. Le fichier de livre numérique sera stocké dans le dossier « My Digital Editions », d'où vous pourrez l'ajouter à calibre.

9.4.15 J'ai une erreur "accès refusé" ?

Une erreur d'accès refusé peut se produire pour de nombreuses raisons possibles, aucune n'ayant un rapport avec calibre.

- Vous pouvez obtenir des erreurs accès refusé si vous utilisez une carte SD avec protection en écriture activée.
- Sous macOS si vous obtenez des erreurs d'autorisation lors de la connexion d'un périphérique à calibre, vous pouvez corriger cela en regardant sous :guilabel :`Préférences système > Sécurité et confidentialité > Confidentialité > Fichiers et répertoires".
- Si vous, ou un certain programme que vous utilisez a changé les permissions en lecture seule sur les fichiers en question.
- Si une erreur de système de fichier sur votre périphérique a amené votre système d'exploitation à monter le système de fichier en lecture seule ou marqué un fichier particulier comme étant en lecture seule pendant une restauration.
- Si les fichiers ont un utilisateur propriétaire autre que vous.
- Si votre fichier est ouvert dans un autre programme.
- Si le fichier se trouve sur un périphérique, vous avez atteint la limite d'un maximum de 256 fichiers à la racine du périphérique. Dans ce cas, vous devez reformater le périphérique/la carte SD référencé avec un système de fichiers FAT32, ou supprimer des fichiers de la mémoire de la carte SD/du périphérique.

Vous devez corriger la cause sous-jacente à l'accès refusé avant de reprendre l'utilisation de calibre. Lisez le message d'erreur avec soin, repérez le fichier en cause et rétablissez les permissions d'accès sur ce fichier ou sur les répertoires qui le contienne

9.4.16 Puis-je avoir la métadonnée commentaire apparaître sur ma liseuse ?

La plupart des liseuses ne supportent pas cela. Vous devriez vous plaindre auprès du fabricant à ce sujet et j'espère que si suffisamment de gens se plaignent, les choses vont changer. En attendant, vous pouvez insérer des métadonnées, y compris des commentaires dans une « Page jaquette » au début du livre numérique, en utilisant l'option « Insérer les métadonnées comme page au début du livre » durant la conversion. L'option se trouve dans la section *Détection de la structure* des paramètres de conversion. Pour que cela ait un effet, vous devez *convertir* le livre. Si votre livre est déjà dans un format qui ne nécessite pas de conversion, vous pouvez convertir de ce format sur le même format.

Une autre alternative est de créer un catalogue sous forme de livre numérique contenant une liste de tous les livres dans votre bibliothèque calibre, avec leurs métadonnées. Cliquez et maintenez le bouton *Convertir* pour accéder à l'outil de création de catalogues. Et avant que vous demandiez, non, vous ne pouvez pas avoir de *lien direct* du catalogue aux livres sur votre liseuse.

9.4.17 Comment obtenir que calibre utilise mon proxy HTTP ?

Par défaut, calibre utilise tous les paramètres de proxy définis dans votre OS. Parfois, ceux-ci sont incorrects, par exemple, sous Windows, si vous n'utilisez pas Microsoft Edge alors les paramètres de proxy peuvent ne pas être à jour. Vous pouvez dire à calibre pour utiliser un serveur proxy notamment en réglant les variables d'environnement `` http_proxy`` et `` https_proxy``. Le format de la variable est : `http://username:password@servername` vous devriez demander à votre administrateur réseau de vous donner la valeur correcte pour cette variable. Notez que calibre ne prend en charge que les proxies HTTP et non les proxies SOCKS. Vous pouvez voir les proxies actuellement utilisés par calibre dans Préférences->Divers.

9.4.18 Je voudrais certaines fonctionnalités ajoutées à calibre. Que puis-je faire ?

Vous avez deux choix :

1. Créer un patch en bidouillant calibre et me l'envoyer pour examen et inclusion. Voir Développement[70].
2. Ouvrir un bug demandant la fonctionnalité[71]. Rappelez-vous que même si vous pensez que votre demande de fonctionnalité est extrêmement importante/essentielle, les développeurs de calibre peuvent ne pas être d'accord. Heureusement, calibre est open source, ce qui signifie que vous avez toujours la possibilité de mettre en œuvre votre fonction vous-même, ou d'embaucher quelqu'un pour le faire pour vous. En outre, calibre a une architecture d'extension compréhensive, vous devriez donc être en mesure de développer votre fonctionnalité en tant qu'extension, voir *Écrire vos propres extensions pour étendre les fonctionnalités de calibre* (page 236).

9.4.19 Pourquoi calibre n'a pas de mise à jour automatique ?

Pour de nombreuses raisons :

- *Il n'est pas nécessaire de mettre à jour chaque semaine*. Si vous êtes satisfait du fonctionnement de calibre, désactivez la notification de mise à jour. Revenez vérifier pour voir si vous voulez mettre à jour une fois par an ou plus. Il y a une case à cocher pour désactiver la notification de mise à jour, sur la notification de mise à jour elle-même.
- Les téléchargements de calibre utilisent actuellement environ 150 To de bande passante par mois[72]. L'implémentation des mises à jour automatiques augmenterait considérablement cela et cela finirait par coûter des milliers de dollars par mois, que quelqu'un devrait payer.

[70] https://calibre-ebook.com/fr/get-involved
[71] https://calibre-ebook.com/fr/bugs
[72] https://calibre-ebook.com/fr/dynamic/downloads

- Si j'implémentais une boîte de dialogue permettant de télécharger la mise à jour et de la lancer, au lieu d'aller sur le site comme il faut le faire actuellement, cela permettrait d'économiser *tout au plus cinq clics par semaine*. Il y a de bien plus hautes priorités à mettre en place pour le développement de calibre.
- Si vous détestez vraiment télécharger la mise à jour calibre toutes les semaines mais tenez à avoir la dernière version, je vous encourage à l'exécuter depuis la source qui rend la mise à jour insignifiante. Les instructions sont *disponibles ici* (page 357).
- Il y a des installateurs de mise à jour automatiques faits par des utilisateurs calibre dans le forum calibre[73].

9.4.20 Quelle est la licence de calibre ?

calibre est sous licence General Public License v3 (une licence open source). Cela signifie que vous êtes libre de redistribuer calibre tant que vous rendez le code source disponible. Donc, si vous voulez mettre calibre sur un CD avec votre produit, vous devez également mettre le code source de calibre sur le CD. Le code source est disponible en téléchargement[74]. Vous êtes libre d'utiliser le résultat des conversions de calibre comme vous le souhaitez. Vous ne pouvez pas utiliser le code et les bibliothèques de calibre dans votre logiciel sans rendre votre logiciel open source. Pour plus de détails, voir The GNU GPL v3[75].

9.4.21 Comment exécuter calibre depuis ma clé USB ?

Une version portable de calibre est disponible ici[76].

9.4.22 Comment faire fonctionner des éléments de calibre comme le téléchargement d'actualités et le serveur de contenu sur mon propre serveur Linux ?

Tout d'abord, vous devez installer calibre sur votre serveur linux. Si votre serveur utilise une distribution Linux récente, vous ne devriez avoir aucun problème pour y installer calibre.

Note : calibre nécessite GLIBC >= 2.31 et libstdc++ >= 6.0.28. Si vous disposez d'un serveur plus ancien, vous devrez soit les compiler à partir des sources, soit utiliser calibre 3.48 qui nécessite GLIBC >= 2.17 ou 2.85.1 qui nécessite le GLIBC >= 2.13 ou calibre 1.48 qui ne nécessite que GLIBC >= 2.10. En outre, bien que les utilitaires en ligne de commande de calibre n'aient pas besoin d'un serveur X en fonctionnement, certains d'entre eux exigent que les bibliothèques du serveur X soient installées sur votre système. Cela est dû à Qt, qui est utilisé pour diverses tâches de traitement d'images, et aux liens avec ces bibliothèques. Si vous obtenez une ImportError à propos de certains modules Qt, il est probable qu'il vous manque certaines bibliothèques X. Les candidats typiques sont : `libxcb-cursor0`, `libxcb-xinerama0`, `libegl1`, `libopengl0`.

Vous pouvez exécuter le serveur de calibre via la commande :

```
/opt/calibre/calibre-server /path/to/the/library/you/want/to/share
```

Vous pouvez télécharger les actualités et les convertir en livre numérique avec la commande :

```
/opt/calibre/ebook-convert "Title of news source.recipe" outputfile.epub
```

[73] https://www.mobileread.com/forums/forumdisplay.php?f=238
[74] https://download.calibre-ebook.com
[75] https://www.gnu.org/licenses/gpl.html
[76] https://calibre-ebook.com/fr/download_portable

Si vous voulez générer un MOBI, utilisez `outputfile.mobi` à la place et utiliser `--output-profile kindle`.

Vous pouvez envoyer par courriel les actualités téléchargées avec la commande :

```
/opt/calibre/calibre-smtp
```

Je vous laisse trouver la ligne de commande exacte comme un exercice pour le lecteur.

Enfin, vous pouvez ajouter les nouvelles téléchargées vers la bibliothèque calibre avec :

```
/opt/calibre/calibredb add --with-library /path/to/library outfile.epub
```

N'oubliez pas de lire la section *Interface en ligne de commande* (page 315) du Manuel de l'Utilisateur calibre pour en savoir sur celles-ci et sur d'autres commandes.

CHAPITRE 10

Tutoriels

Vous trouverez ici des tutoriels pour vous guider dans des fonctionnalités plus avancées de calibre, telles que XPath et les modèles.

10.1 Gérer les sous-groupes de livres, par exemple « genre »

Certaines personnes souhaitent organiser les livres dans leur bibliothèque en sous-groupes, similaires à des sous-répertoires. La raison la plus habituellement donnée est de créer des hiérarchies de genre, mais il y en a beaucoup d'autres. Un utilisateur demandais une manière d'organiser des manuels par sujet et numéro de cours. Un autre voulait garder une trace de cadeaux par sujet et destinataires. Ce tutoriel utilisera l'exemple genre pour le reste de ce poste.

Avant de commencer, merci de noter que nous ne parlons pas de répertoires sur le disque dur. Les sous-groupes ne sont pas des répertoires de fichiers. Les livres ne seront copiés nul part. La structure de fichier de la bibliothèque calibre n'est pas affectée. Au lieu de cela, nous faisons une présentation d'une manière d'organiser et d'afficher des sous-groupes de livres dans une bibliothèque calibre.

- *Configuration* (page 163)
- *Recherche* (page 165)
- *Restrictions* (page 166)
- *Modèles de fonctions utiles* (page 166)

Les requis habituellement donnés pour les sous-groupes comme les genres sont :

- Un sous-groupe (par ex., un genre) doit contenir (pointer vers) des livres, pas des catégories de livres. C'est ce qui distingue les sous-groupes des catégories utilisateur calibre.
- Un livre peut être dans de multiples sous-groupes (genres). C'est ce qui distingue les sous groupes des répertoires de fichier physiques.
- Les sous-groupes (genres) doivent former une hiérarchie ; les sous-groupes peuvent contenir des sous-groupes.

Les étiquettes vous donnent les deux premières. Si vous étiquetez un livre avec le genre alors vous pouvez utiliser le Navigateur d'étiquettes (ou la recherche) pour trouver les livres avec ce genre, vous donnant la première. Beaucoup de livres peuvent avoir la/les même(s) étiquette(s), vous donnant la deuxième. Le problème est que les étiquettes ne satisfont pas le troisième prérequis. Elles ne procurent pas une hiérarchie.

La fonctionnalité hiérarchie de calibre vous donne la troisième – la capacité de voir les genres dans un "arbre" et la capacité de facilement rechercher des livres dans genre ou sous-genre. Par exemple, supposons que votre structure de genre est similaire à ceci

```
Genre
    . History
    .. Japanese
    .. Military
    .. Roman
    . Mysteries
    .. English
    .. Vampire
    . Science Fiction
    .. Alternate History
    .. Military
    .. Space Opera
    . Thrillers
    .. Crime
    .. Horror
    etc.
```

En utilisant la caractéristique hiérarchie, vous pouvez voir ces genres dans le navigateur d'étiquettes sous la forme d'un arbre, comme montré dans la capture d'écran. Dans cet exemple, le niveau intrinsèque (Genre) est une colonne personnalisée qui contient les genres. Les genres contenant des sous genres apparaissent avec un petit triangle à leur suite. Cliquer sur ce triangle ouvrira l'élément et affichera les sous genres, comme vous pouvez le voir avec Histoire et Science Fiction.

Cliquer sur un genre peut rechercher tous les livres avec ce genre ou les descendants de ce genre. Par exemple, en cliquant sur Science Fiction vous aurez accès aux trois genres enfants de Alternate History, Military et Space Opera. Cliquer sur Alternate History donnera accès aux livres correspondant à ce genre précis, ignorant ceux qui sont dans Military et Space

Opera. Bien sûr, un livre peut avoir plusieurs genres. Si un livre est à la fois Space Opera et Militaire, alors vous verrez ce livre en cliquant sur l'un ou l'autre genre. « Rechercher » est abordé plus en détail ci dessous.

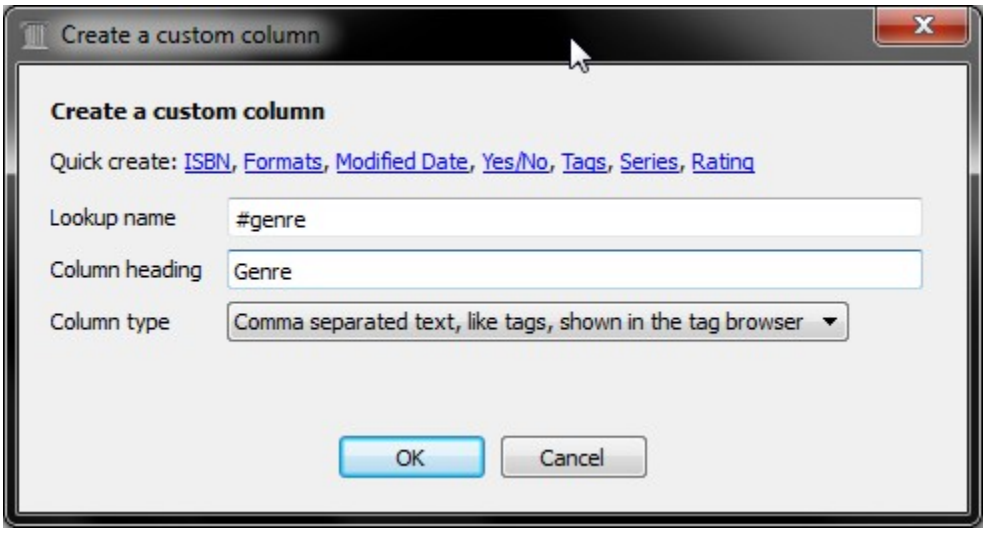
Une autre chose que vous pouvez voir dans l'image est que le genre Military apparaît deux fois, une fois sous History et une fois sous Science Fiction. Parce que les genres sont dans une hiérarchie, ceux-ci sont deux genres séparés. Un livre peut être dans un, l'autre ou les deux (discutable dans ce cas). Par exemple, les livres « The Second World War » de Winston Churchill pourraient être dans « History.Military ». Les Honor Harrington de David Weber pourraient être dans « Science Fiction.Military », et pour cette matière également dans « Science Fiction.Space Opera. »

Une fois qu'un genre existe, c'est qu'il y a au moins un livre de ce genre, vous pouvez facilement l'appliquer à d'autres livres en faisant glisser les livres de la liste de la bibliothèque sur le genre que vous voulez leur donner. Vous pouvez également inclure les livres dans des genres par l'intermédiaire de l'Éditeur de métadonnées ; plus sur ceci ci dessous.

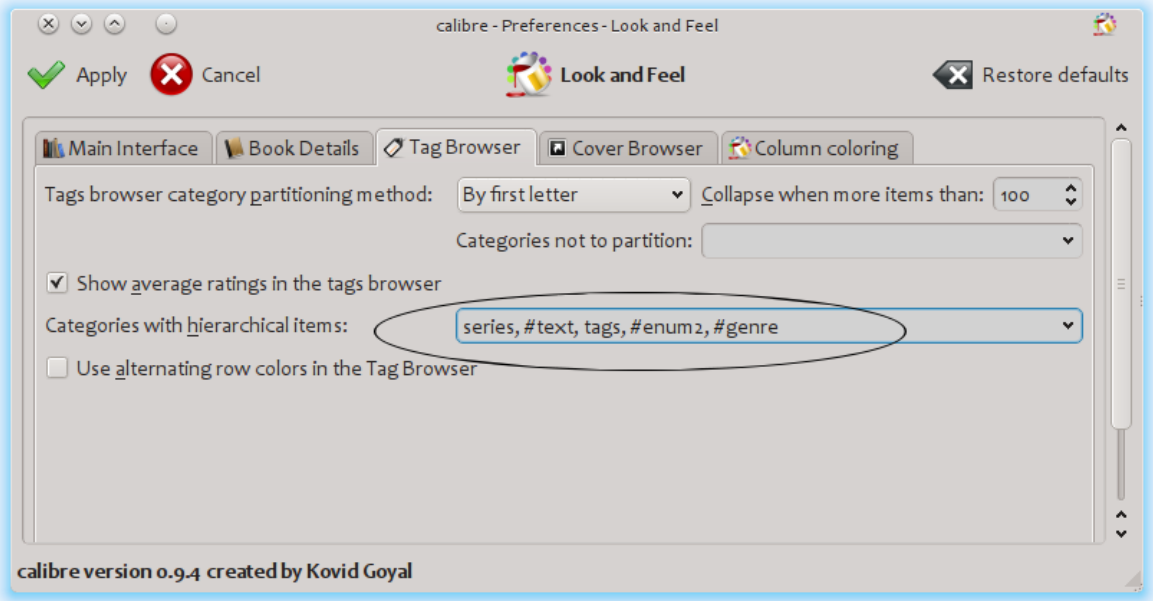
10.1.1 Configuration

A présent, votre question peut être : « Comment mettre tout cela en place ? ». Il y a trois étapes : 1) Créer la colonne personnalisée, 2) Dire à calibre que la nouvelle colonne doit être traitée comme une hiérarchie, et 3) ajouter les genres.

Vous créez la colonne personnalisée par la voie habituelle, en utilisant Préférences–>Ajout de colonnes personnalisées. Cet exemple utilise « #genre » comme nom de recherche et « Genre » comme en-tête de colonne. Le type de colonne est « Texte séparé par des virgules, comme étiquettes, affichée dans le navigateur d'étiquettes. »

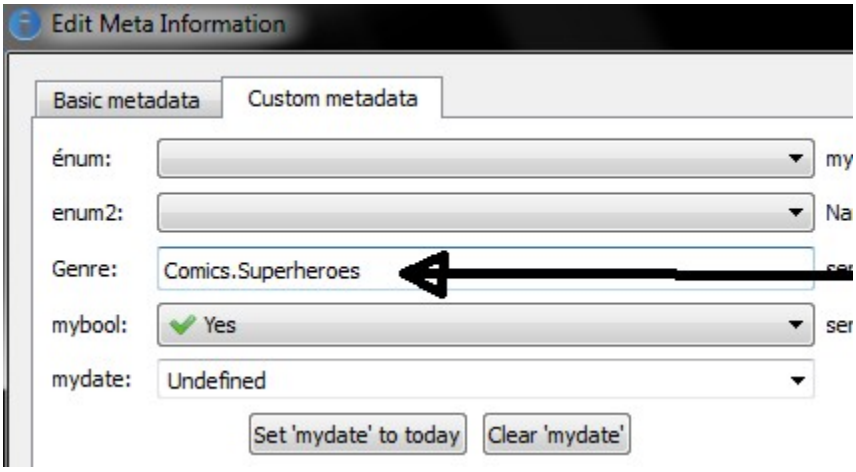


Après avoir redémarrer calibre, vous devez dire à calibre que la colonne est à être traitée comme une hiérarchie. Allez dans *Préférences → Apparence → Navigateur d'étiquettes* et entrez le nom de recherche « #genre » dans la case « Catégories avec des articles hiérarchiques ». Cliquez sur Appliquer, et vous en avez fini de mettre en place.



En ce moment, il n'y a aucun genre dans la colonne. Nous nous retrouvons avec la dernière étape : comment appliquer un genre à un livre. Un genre n'existe pas dans calibre jusqu'à ce qu'il apparaisse sur au moins un livre. Pour apprendre comment appliquer un genre pour la première fois, nous devons entrer dans certains détails au sujet de à quoi ressemble un genre dans les métadonnées d'un livre.

Une hiérarchie de "choses" est construite en créant un élément consistant en expressions séparées par des points. En continuant l'exemple genre, ces éléments seraient « History.Military », « Mysteries.Vampire », « Science Fiction.Space Opera », etc. Donc pour créer un nouveau genre, vous prenez un livre qui devrait avoir ce genre, éditez ses métadonnées, et entrez le nouveau genre dans la colonne que vous avez créée. En continuant notre exemple, si vous voulez attribuer un nouveau genre « Comics » avec un sous genre « Superheroes » à un livre, vous devrez "éditer les métadonnées" pour ce livre(bande dessinée), choisir l'onglet Métadonnées personnalisées, et entrer « Comics.Superheroes » comme affiché ici (ignorez les autres colonnes personnalisées) :

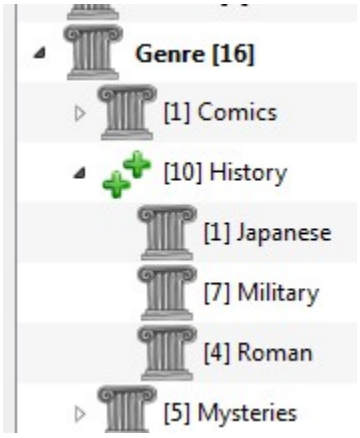


Après avoir fait ce qui est indiqué plus haut, vous voyez dans le Navigateur d'étiquettes :

A partir d'ici, pour appliquer ce nouveau genre à un livre (une bande dessinée, vraisemblablement), vous pouvez soit déposer le livre sur le genre, ou l'ajouter au livre en utilisant éditer les métadonnées exactement de la même manière que plus haut.

Note : L'affichage hiérarchique fonctionne uniquement si le Navigateur d'étiquettes est paramétré pour trier les éléments par nom. C'est l'option par défaut et elle peut être vérifiée en cliquant le bouton *Configurer* en bas du Navigateur d'étiquettes.

10.1.2 Recherche



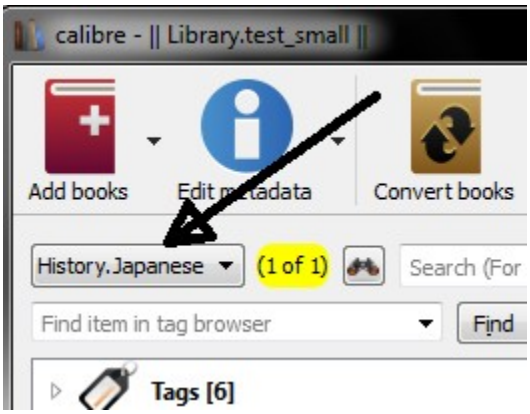
La meilleure façon de chercher des genres est d'utiliser le navigateur d'étiquettes, de cliquer sur le genre que vous souhaitez voir. En cliquant sur un genre qui a des descendants, vous verrez tous les livres contenus dans ce genre et dans tous les descendants de ce genre. Cependant, ceci peut évoquer une question. Juste parce qu'un genre a des descendants ne signifie pas que ce n'est pas un genre de son propre chef. Par exemple, un livre peut avoir le genre « History » mais pas « History.Military ». Comment recherchez-vous des livres avec seulement « History » ?

Le mécanisme de recherche du Navigateur d'étiquettes sait si un élément a des sous éléments. S'il le fait, cliquer sur l'élément permet d'accéder à 5 variations de recherche au lieu de trois. Le premier est le plus vert normal, qui vous affiche les livres avec ce genre uniquement (par ex. History). Le second est le double plus (montré plus haut), qui vous affiche les livres avec ce genre et tous les sous genres (par ex. History et History.Military). Le troisième est le moins rouge normal, qui vous affiche les livres qui n'ont pas ce genre avec exactitude, Le quatrième un double moins, qui montre les livres sans ce genre ou les sous genres. Le cinquième est revenir au début, pas de marque, signifie pas de recherche.

10.1.3 Restrictions

Si vous recherchez un genre, créez une recherche sauvegardée pour celui-ci, vous pouvez utiliser la fonction Restriction pour créer une bibliothèque virtuelle contenant les livres de ce genre. C'est très utile si vous voulez faire d'autres recherches à l'intérieur du genre ou gérer/mettre à jour les métadonnées des livres dans le genre. Poursuivant notre exemple, vous pouvez créer une recherche sauvegardée nommée "History.Japanese" en cliquant d'abord sur le sous-genre Japanese dans le Navigateur d'étiquettes pour obtenir une recherche dans la barre de recherche, puis en entrant History.Japanese dans la boîte de Recherches sauvegardées, puis en appuyant sur le bouton « Enregistrer la recherche » (le carré vert avec un signe plus blanc du côté droit).

Après avoir créé la recherche sauvegardée, vous pouvez l'utiliser comme restriction.



10.1.4 Modèles de fonctions utiles

Vous pourriez vouloir employer l'information de genre dans un modèle, comme avec Enregistrer sous ou Envoyer au périphérique . La question pourrait alors être « Comment est-ce que j'obtiens le nom ou les noms périphériques de ce genre ? » Un modèle de fonction calibre, sous-articles, est fournie pour faire cela facilement.

Par exemple, supposons que vous voulez ajouter le niveau de genre le plus périphérique au modèle save-to-disk pour faire des répertoires genre, comme dans « History/The Gathering Storm - Churchill, Winston ». Pour faire cela, vous devez extraire le premier niveau de la hiérarchie et l'ajouter au début avec une barre oblique pour indiquer qu'il devra en faire un répertoire. Le modéle ci-dessous accomplis cela :

```
{#genre:subitems(0,1)||/}{title} - {authors}
```

Voir *Le langage des modèles* (page 169) pour plus d'informations sur les modèles et la fonction `subitems()`.

10.2 Tutoriel XPath

Dans ce tutoriel, il vous sera donné une légère introduction à XPath[77], un langage de requête qui peut être utilisé pour sélectionner des parties quelconques de documents HTML[78] dans calibre. XPath est un standard utilisé largement, et faire une recherche sur Google produira une tonne d'information. Ce tutoriel, toutefois, se concentrera sur l'utilisation d'XPath pour des tâches relatives aux livres numériques comme trouver les entêtes de chapitre dans un document HTML non structuré.

Contenu

- *Sélection par nom de balise* (page 167)
- *Sélection par attributs* (page 168)
- *Sélection par contenu de balise* (page 168)
- *Exemple de livre numérique* (page 168)
- *Les fonctions XPath intégrées* (page 169)

10.2.1 Sélection par nom de balise

La forme la plus simple de sélection est de sélectionner les balises par nom. Par exemple, supposons que vous voulez sélectionner toutes les balises `<h2>` dans un document. La requête XPath pour cela est simplement

```
//h:h2        (Selects all <h2> tags)
```

Le préfixe *//signifie *rechercher à n'importe quel niveau du document*. Maintenant supposons que vous voulez rechercher les balises ``<span>`` qui sont contenues dans des balises `<a>`. Ceci peut être accompli avec

```
//h:a/h:span    (Selects <span> tags inside <a> tags)
```

Si vous voulez rechercher des balises à un niveau particulier dans le document, changez le préfixe

```
/h:body/h:div/h:p (Selects <p> tags that are children of <div> tags that are
             children of the <body> tag)
```

Ceci correspondra uniquement à `<p>Un livre numérique très court pour démontrer l'utilisation d'XPath.</p>` dans le *Exemple de livre numérique* (page 168) mais dans aucune des autres balises `<p>`. Le préfixe `h:` dans l'exemple plus haut est nécessaire pour correspondre aux balises XHTML. Ceci car, en interne, calibre représente tout le contenu comme du XHTML. En XHTML les balises ont un *espace de nom*, et `h:` est le préfixe d'espace de nom pour les balises HTML.

Maintenant supposons que vous voulez sélectionner conjointement les balises `<h1>` et `<h2>`. Pour faire cela, nous avons besoin d'une construction XPath appelée *prédicat*. Un *prédicat* est simplement un test qui est utilisé pour sélectionner les balises. Les tests peuvent être arbitrairement puissants et au long de la progression de ce tutoriel, vous verrez des exemples plus puissants. Un prédicat est créé en mettant l'expression test entre crochets.

```
//*[name()='h1' or name()='h2']
```

Il y a plusieurs nouvelles caractéristiques dans cette expression XPath. La première est l'utilisation du joker `*`. Il signifie *correspond à n'importe quelle balise*. Maintenant regardez l'expression test `name()='h1' or name()='h2'`. *name()*

[77] https://fr.wikipedia.org/wiki/XPath
[78] https://fr.wikipedia.org/wiki/HTML

est un exemple d'une *fonction intégrée*. Il évalue simplement le nom de la balise. Donc en l'utilisant, nous pouvons sélectionner des balises dont les noms sont soit *h1* ou *h2*. Notez que la fonction *name()* ignore les espaces de nom de sorte qu'il n'est d'aucune nécessité du préfixe `h:`. XPath a plusieurs fonctions intégrées utiles. Quelques unes de plus seront introduites dans ce tutoriel.

10.2.2 Sélection par attributs

Pour sélectionner des balises sur base de leurs attributs, l'utilisation des prédicats est requise //

```
//*[@style]              (Select all tags that have a style attribute)
//*[@class="chapter"]    (Select all tags that have class="chapter")
//h:h1[@class="bookTitle"] (Select all h1 tags that have class="bookTitle")
```

Ici l'opérateur `@` réfère aux attributs de la balise. Vous pouvez utiliser certaines des *Fonctions intégrées XPath* (page 169) pour effectuer des correspondances plus sophistiquées sur les valeurs attribut.

10.2.3 Sélection par contenu de balise

En utilisant XPath, vous pouvez même sélectionner des balises basées sur le texte qu'elles contiennent. La meilleure manière de faire cela est d'utiliser la puissance des *expressions régulières* par l'intermédiaire de la fonction intégrée *re :test()* :

```
//h:h2[re:test(., 'chapter|section', 'i')] (Selects <h2> tags that contain the words␣
↪chapter or
                                          section)
```

Ici l'opérateur `.` réfère au contenu de la balise, tout comme l'opérateur `@` réfère à ses attributs.

10.2.4 Exemple de livre numérique

```
<html>
    <head>
        <title>A very short e-book</title>
        <meta name="charset" value="utf-8" />
    </head>
    <body>
        <h1 class="bookTitle">A very short e-book</h1>
        <p style="text-align:right">Written by Kovid Goyal</p>
        <div class="introduction">
            <p>A very short e-book to demonstrate the use of XPath.</p>
        </div>

        <h2 class="chapter">Chapter One</h2>
        <p>This is a truly fascinating chapter.</p>

        <h2 class="chapter">Chapter Two</h2>
        <p>A worthy continuation of a fine tradition.</p>
    </body>
</html>
```

10.2.5 Les fonctions XPath intégrées

name()
Le nom de la balise actuelle.

contains()
`contains(s1, s2)` renvoie *true* si s1 contient s2.

re :test()
`re:test(src, pattern, flags)` renvoie *true* si la chaîne *src* correspond au *pattern* de l'expression régulière. un indicateur particulièrement utile est `i`, il rend les correspondances sensibles à la casse. Une bonne amorce sur la syntaxe pour les expressions régulières peut être trouvée sur regexp syntax[79]

10.3 Le langage de modèle calibre

Le langage de modèle de calibre est un langage spécifique à calibre utilisé dans l'ensemble de calibre pour des tâches telles que la spécification des chemins de fichiers, le formatage des valeurs et le calcul de la valeur des colonnes spécifiées par l'utilisateur. Exemples :

- Spécifier la structure du dossier et les noms de fichiers lors de l'enregistrement des fichiers de la bibliothèque calibre sur le disque ou le lecteur de livres numériques.
- Définir des règles pour l'ajout d'icônes et de couleurs à la liste de livres de calibre.
- Définir les *colonnes virtuelles* qui contiennent des données provenant d'autres colonnes.
- Recherche avancée dans la bibliothèque.
- Recherche et remplacement avancé des métadonnées.

Le langage est construit autour de la notion de *modèle*, qui spécifie les métadonnées du livre à utiliser, les calculs sur ces métadonnées, et la façon dont elles doivent être formatées.

10.3.1 Modèles de base

Un modèle de base est constitué d'une ou plusieurs `expression modèle`. Une `expression modèle` consiste en du texte et des noms entre crochets (`{}`) qui sont remplacés par les métadonnées correspondantes du livre en cours de traitement. Par exemple, le modèle par défaut de calibre utilisé pour enregistrer des livres sur un périphérique comporte 4 `expressions modèle` :

```
{author_sort}/{title}/{title} - {authors}
```

Pour le livre « The Foundation » par « Isaac Asimov » cela donnera :

```
Asimov, Isaac/The Foundation/The Foundation - Isaac Asimov
```

Les barres obliques ne sont pas des `expressions modèle` car elles se trouvent entre les `{}`. Un tel texte est laissé là où il apparaît. Par exemple, si le modèle est :

```
{author_sort} Some Important Text {title}/{title} - {authors}
```

dès lors, pour « The Foundation », le modèle produit :

[79] https://docs.python.org/library/re.html

```
Asimov, Isaac Some Important Text The Foundation/The Foundation - Isaac Asimov
```

Une `expression modèle` peut accéder à toutes les métadonnées disponibles dans calibre, y compris les colonnes personnalisées (colonnes que vous créez vous-même), en utilisant le `nom de recherche` d'une *colonne*. Pour trouver le nom de recherche d'une *colonne* (parfois appelée *champs*), passez votre souris sur l'en-tête de la colonne dans la liste des livres de calibre. Les noms de recherche pour les colonnes personnalisées commencent toujours par `#`. Pour les colonnes de type série, il y a un champ supplémentaire appelé `#lookup name_index` qui est l'index de la série pour ce livre dans la série. Par exemple, si vous avez une colonne de série personnalisée nommée `#messéries`, il y aura également une colonne nommée `#messéries_index`. L'index de la colonne de série standard est nommé `series_index`.

En plus des champs standard basés sur des colonnes, vous pouvez également utiliser :

- `{formats}` - Une liste des formats disponibles dans la bibliothèque calibre pour un livre
- `{identifiers:select(isbn)}` - L'ISBN du livre

Si les métadonnées du champ pour un livre donné ne sont pas définies, le champ dans le modèle est remplacé par la chaîne vide (`''`). Par exemple, considérons le modèle suivant

```
{author_sort}/{series}/{title} {series_index}
```

Si le livre d'Asimov « Seconde Fondation » fait partie de la série « Fondation », alors le modèle produit

```
Asimov, Isaac/Foundation/Second Foundation 3
```

Si aucune série n'a été saisie pour le livre, le modèle produit

```
Asimov, Isaac/Second Foundation
```

Le processeur de modèles supprime automatiquement les barres obliques multiples et les espaces avant ou arrière.

10.3.2 Mise en forme avancée

Outre la substitution de métadonnées, les modèles peuvent inclure du texte supplémentaire de manière conditionnelle et contrôler le formatage des données substituées.

Comprenant conditionnellement le texte

Parfois, vous voulez que du texte n'apparaisse dans la sortie que si un champ n'est pas vide. Un cas courant est celui de `series` et `series_index` où vous voulez soit rien, soit les deux valeurs séparées par un trait d'union. calibre gère ce cas en utilisant une syntaxe spéciale de `expression modèle`.

Par exemple, et en utilisant l'exemple de Fondation ci-dessus, supposons que vous voulez que le modèle produise *Fondation - 3 - Deuxième Fondation*. Ce modèle produit ce résultat :

```
{series} - {series_index} - {title}
```

Cependant, si un livre n'a pas de série, le modèle produira - - *le titre*, ce qui n'est probablement pas ce que vous voulez. En général, les gens veulent que le résultat soit le titre sans les traits d'union superflus. Vous pouvez y parvenir en utilisant la syntaxe de modèle suivante :

```
{field:|prefix_text|suffix_text}
```

Cette `expression modèle` indique que si `champ` a la valeur `XXXX`, le résultat sera `prefix_textXXXXXsuffix_text`. Si `champ` est vide (n'a pas de valeur) alors le résultat sera la chaîne vide (rien) car le préfixe et le suffixe sont ignorés. Le préfixe et le suffixe peuvent contenir des blancs.

Ne pas utiliser de sous-modèles (`{ … }`) ou de fonctions (voir ci-dessous) dans le préfixe ou le suffixe.

En utilisant cette syntaxe, nous pouvons résoudre le problème de non-série ci-dessus avec le modèle

```
{series}{series_index:| - | - }{title}
```

Les traits d'union ne seront inclus que si le livre possède un index de série, ce qui n'est le cas que si le livre possède une série. Si l'on reprend l'exemple de Fondation, le modèle produira *Fondation - 1 - Deuxième Fondation*.

Notes :

- Vous devez inclure les deux points après le `nom de la recherche` si vous utilisez un préfixe ou un suffixe.
- Vous devez utiliser soit aucun des caractères `|`, soit les deux. L'utilisation d'un seul caractère, comme dans `{field:| - }`, n'est pas autorisée.
- Il est possible de ne fournir aucun texte pour le préfixe ou le suffixe, comme dans `{série:|| - }`. Le modèle `{title:||}` est le même que `{title}`.

Formatage

Supposons que vous vouliez que `series_index` soit formaté en trois chiffres avec des zéros en tête. Ceci fait l'affaire :

> `{series_index:0>3s}` - Trois chiffres avec des zéros en tête

Pour des zéros qui suivent, utilisez :

> `{series_index:0>3s}` - Trois chiffres avec des zéros qui suivent

Si vous utilisez des indices de série avec des valeurs fractionnées, par exemple 1.1, vous pouvez souhaiter que les points décimaux soient alignés. Par exemple, vous pouvez vouloir que les indices 1 et 2.5 apparaissent sous la forme 01.00 et 02.50 afin qu'ils soient triés correctement sur un périphérique qui effectue un tri lexical. Pour ce faire, utilisez :

> `{series_index:0>5.2f}` - Cinq caractères composés de deux chiffres avec des zéros en tête, un point décimal, puis deux chiffres après le point décimal.

Si vous ne voulez que les deux premières lettres de la donnée, utilisez :

> `{author_sort :.2}` - Seulement les deux premières lettres du nom de tri de l'auteur

Une grande partie du formatage du langage de modèle de calibre provient de Python. Pour plus de détails sur la syntaxe de ces opérations de formatage avancées, voir la *documentation Python <https ://docs.python.org/3/library/string.html#formatstrings>*

10.3.3 Utilisation de modèles pour définir des colonnes personnalisées

Les modèles peuvent être utilisés pour afficher des informations qui ne sont pas dans les métadonnées de calibre, ou pour afficher les métadonnées différemment du format normal de calibre. Par exemple, vous pourriez vouloir afficher le `ISBN`, un champ que calibre n'affiche pas. Pour cela, vous pouvez créer une colonne personnalisée de type *Colonne construite à partir d'autres colonnes* (appelée ci-après *colonnes composites*) et fournir un modèle pour générer le texte affiché. La colonne affichera le résultat de l'évaluation du modèle. Par exemple, pour afficher l'ISBN, créez la colonne et entrez « identifiers :select(isbn)} » dans la champ du modèle. Pour afficher une colonne contenant les valeurs de deux colonnes personnalisées de séries, séparées par une virgule, utilisez `{#series1:||,}{#series2}`.

Les colonnes composites peuvent utiliser n'importe quelle option modèle, y compris le formatage.

Remarque : Vous ne pouvez pas modifier les données affichées dans une colonne composite. Vous devez plutôt modifier les colonnes sources. Si vous modifiez une colonne composite, par exemple en double-cliquant dessus, calibre ouvrira le modèle à modifier, et non les données sous-jacentes.

10.3.4 Modèles et tableaux de connexions

Les tableaux de connexions sont utilisés pour modifier les métadonnées écrites dans les livres pendant l'envoi à une liseuse et les opérations de sauvegarde sur le disque. Le tableau de connexion vous permet de spécifier pour un modèle de lecteur précis quelles données à insérer dans les métadonnées du livre. Vous pouvez utiliser les tableaux de connexions pour modifier les champs suivants : authors, author_sort, language, publisher, tags, title, title_sort. Cette fonctionnalité aide ceux qui veulent utiliser les différentes métadonnées des livres pour contrôler le tri ou l'affichage des livres sur leurs périphériques.

Lorsque vous créez un tableau de connexion, vous spécifiez le format et le périphérique pour lequel le tableau de connexion doit être utilisé. Un périphérique spécial est procuré, `save_to_disk` qui est utilisé lors de la sauvegarde de formats (en opposition à les envoyer vers le périphérique). Une fois que vous avez choisi le format et le périphérique, vous choisissez les champs de métadonnées à modifier, indiquez le modèle à appliquer pour fournir les nouvelles valeurs. Ces modèles sont *reliés* à leurs champs de destination, d'où le nom de *tableau de connexions*. Vous pouvez bien sûr utiliser des colonnes composites dans ces modèles.

Les tableaux de connexions sont assez flexibles et peuvent être écrits en mode fonction unique, en mode programme modèle, en mode programme général ou en mode modèle Python.

Quand un tableau de connexion doit s'appliquer (Serveur de contenu, sauvegarde sur disque ou envoyer au périphérique), calibre recherche les tableau de connexions définis et celui qui est correct pour le format donné et le périphérique. Par exemple, pour trouver le tableau de connexion approprié pour un livre EPUB envoyé à un périphérique ANDROID, calibre recherche dans les tableaux de connexion dans l'ordre de recherche suivant :

- un tableau de connexion avec une correspondance exacte sur le format et le périphérique, par ex., `EPUB` et `ANDROID`
- un tableau de connexion avec une correspondance exacte sur le format et le choix spécial `n'importe quel périphérique`, par ex., `EPUB` et `n'importe quel périphérique`
- un tableau de connexion avec le choix spécial `n'importe quel format` et une correspondance exacte sur le périphérique, par ex., `n'importe quel format` et `ANDROID`
- un tableau de conversion avec `n'importe quel format` et `n'importe quel périphérique`

Les étiquettes et les champs auteur ont un traitement spécial, parce que ces deux champs peuvent contenir plusieurs données. Un livre peut avoir plusieurs étiquettes et plusieurs auteurs. Lorsque vous spécifiez que l'un de ces deux champs doit être modifié, le résultat du modèle est examiné pour voir s'il y a plus d'un élément dedans. Pour les étiquettes, le résultat est découpé partout où calibre trouve une virgule. Par exemple, si le modèle produit les valeurs `Thriller, Horreur`, alors ke résultat sera deux étiquettes, `Thriller` et `Horreur`. Il n'est pas possible de mettre une virgule au milieu d'une étiquette.

La même chose se produit pour les auteurs, mais en utilisant un caractère différent comme séparateur, le *&* (esperluette) au lieu d'une virgule. Par exemple, si le modèle produit la valeur `Blogs, Joe&Posts, Susan`, alors le livre finira avec deux auteurs `Blogs, Joe` et `Posts, Susan`. Si le modèle produit la valeur `Blogs, Joe;Posts, Susan`, alors ce livre aura un auteur avec un nom assez étrange.

Les tableaux de connexions affectent les métadonnées écrites dans le livre quand il est sauvegardé sur le disque ou copié sur le périphérique. Les tableaux de connexions n'affecte pas les métadonnées utilisées par `Enregistrer sur le disque` et `Envoyer au périphérique`. A la place, les noms de fichiers sont construits en utilisant les modèles entrés dans la fenêtre de préférence appropriée.

10.3.5 Utilisation des fonctions dans les modèles - Mode de fonctionnement unique

Supposons que vous vouliez afficher la valeur d'un champ en casse majuscules alors que ce champ est normalement en casse titre. Vous pouvez le faire en utilisant les fonctions de modèle. Par exemple, pour afficher le titre en casse majuscules, utilisez la fonction `uppercase`, comme dans `{title:uppercase()}`. Pour l'afficher en casse titre, utilisez la fonction `{title:titlecase()}`.

Les fonctions sont placées dans la partie format du modèle, après le `:` et avant le premier `|` ou le `}` de fermeture si aucun préfixe/suffixe n'est utilisé. Si vous avez à la fois une référence de format et de fonction, la fonction vient après un deuxième `:`. Les fonctions renvoient la valeur de la colonne spécifiée dans le modèle, convenablement modifiée.

La syntaxe pour utiliser les fonctions est l'une des suivantes

```
{lookup_name:function(arguments)}
{lookup_name:format:function(arguments)}
{lookup_name:function(arguments)|prefix|suffix}
{lookup_name:format:function(arguments)|prefix|suffix}
```

Les noms de fonctions doivent toujours être suivis de parenthèses ouvrantes et fermantes. Certaines fonctions nécessitent des valeurs supplémentaires (arguments), qui sont placées à l'intérieur des parenthèses. Les arguments sont séparés par des virgules. Les virgules Littérales (les virgules en tant que texte, et non en tant que séparateurs d'arguments) doivent être précédées d'une barre oblique inversée (`\`) . Le dernier (ou le seul) argument ne peut pas contenir de parenthèse fermante textuelle.

Les fonctions sont évaluées avant les spécifications de format et le préfixe/suffixe. Voir plus bas pour un exemple d'utilisation à la fois d'un format et d'une fonction.

Important : Si vous avez de l'expérience en programmation, veuillez noter que la syntaxe du *Mode Fonction Unique* n'est pas celle à laquelle vous vous attendez. Les chaînes de caractères ne sont pas citées et les espaces sont significatifs. Tous les arguments sont considérés comme des constantes ; il n'y a pas d'expressions.

N'utilisez pas les sous-modèles (`{ … }`) comme arguments de fonction. A la place, utilisez plutôt Mode Programme Modèle et Mode Programme Général.

Certaines fonctions nécessitent des expressions régulières. Dans le langage des modèles, la correspondance des expressions régulières est insensible à la casse.

Dans la documentation des fonctions ci-dessous, la notation `[quelque chose]*` signifie que `quelque chose` peut être répété zéro fois ou plus. La notation `[quelque chose]+` signifie que `quelque chose` est répété une ou plusieurs fois (doit exister au moins une fois).

Les fonctions destinées à être utilisées en Mode Fonction Unique sont les suivantes :

- `capitalize()` – renvoie la valeur avec la première lettre en majuscule et le reste en minuscule.
- `contains(pattern, text if match, text if not match)` – vérifie si la valeur est couverte par l'expression régulière `pattern`. Renvoie `text if match` si le motif correspond à la valeur, sinon renvoie `text if no match`.
- `count(separator)` – interprète la valeur comme une liste d'éléments séparés par le `separator` et retourne le nombre d'éléments dans la liste. La plupart des listes utilisent une virgule comme séparateur, mais `authors` utilise une esperluette (&). Exemples : `{tags:count(,)}`, `{authors:count(&)}`. Alias : `count()`, `list_count()`
- `format_number(template)` – interprète la valeur comme un nombre et formate ce nombre en utilisant un modèle de formatage Python tel que `{0:5.2f}` ou *{0 :,d}`* ou `${0:5,.2f}`. Le modèle de formatage doit commencer par `{0:` et se terminer par `}` comme dans les exemples ci-dessus. Exception : vous pouvez omettre

le début « {0 :» et la fin « } » si le modèle de format ne contient qu'un format. Voir le langage des modèles et la documentation Python[80] pour plus d'exemples. Renvoie la chaîne vide si le formatage échoue.

- `human_readable()` – s'attend que la valeur soit un nombre et renvoie une chaîne représentant ce nombre en KB, MB, GB, etc.
- `ifempty(text)` – si le champ n'est pas vide, renvoie la valeur du champ. Autrement renvoie *text if empty*.
- `in_list(separator, [ pattern, found_val, ]* not_found_val)` – interprète la valeur comme une liste d'éléments séparés par le `separator`, en vérifiant le `pattern` pour chaque élément de la liste. Si le `pattern` correspond à un élément, il renvoie `found_val`, sinon il renvoie `not_found_val`. La paire `pattern``et ``found_value` peut être répétée autant de fois que souhaité, ce qui permet de renvoyer des valeurs différentes selon la valeur de l'élément. Les motifs sont vérifiés dans l'ordre, et la première correspondance est retournée.
- `language_strings(localize)` – retourne les noms de langue[81] pour les codes de langue[82] passés en tant que valeur. Exemple : `{languages:language_strings()}`. Si `localize` est zéro, retourne les chaînes de caractères en anglais. Si `localize` est différent de zéro, retourne les chaînes de caractères dans la langue de la locale courante. `Lang_codes` est une liste séparée par des virgules.
- `list_item(index, separator)` – interprète la valeur comme une liste d'éléments séparés par un `separator`, en retournant `index'`. Le premier élément est le numéro zéro. Le dernier élément a l'indice `-1` comme dans `list_item(-1,separator)`. Si l'élément n'est pas dans la liste, alors la chaîne vide est retournée.
- `lookup([ pattern, key, ]* else_key)` – Les motifs seront comparés à la valeur dans l'ordre. Si un motif correspond, la valeur du champ nommé par `key` est retournée. Si aucun motif ne correspond, la valeur du champ nommé par `else_key` est retournée. Voir `switch` (ci-dessous).
- `lowercase()` – renvoie la valeur du champ en minuscule.
- `rating_to_stars(use_half_stars)` – Renvoie la note sous forme de chaîne de caractères étoilés (★). La valeur doit être un nombre entre 0 et 5. Définissez use_half_stars à 1 si vous voulez des caractères de demi-étoile pour les nombres fractionnés disponibles avec les colonnes d'évaluation personnalisées.
- `re(pattern, replacement)` – retourne la valeur après application de l'expression régulière. Toutes les instances de `pattern` dans la valeur sont remplacées par `replacement`. Le langage des modèles utilise des Expressions régulières Python[83] insensibles à la casse.
- `select(key)` – interprète la valeur comme une liste d'éléments séparés par des virgules, chaque élément ayant la forme `id:value` (le format `identifiant` de calibre). La fonction trouve la première paire dont l'id est égal à key et renvoie la valeur correspondante. Si aucun id ne correspond, la fonction renvoie la chaîne vide.
- `shorten(left chars, middle text, right chars)` – Renvoie une version raccourcie de la valeur, composée de `left chars` du début de la valeur, suivis de `middle text`, suivis de `right chars` de la fin de la valeur. Les « left chars » et les « right chars » doivent être des entiers non négatifs. Exemple : supposons que vous vouliez afficher le titre avec une longueur maximale de 15 caractères. Un modèle qui fait cela est `{titre:shorten(9,-,5)}`. Pour un livre dont le titre est *Ancient English Laws in the Times of Ivanhoe*, le résultat sera *Ancient E-anhoe* : les 9 premiers caractères du titre, un −, puis les 5 derniers caractères. Si la longueur de la valeur est inférieure à `left chars` + `right chars` + la longueur du `middle text` alors la valeur sera retournée inchangée. Par exemple, le titre *Le Dôme* ne sera pas modifié.
- `str_in_list(separator, [ string, found_val, ]+ not_found_val)` – interprète la valeur comme une liste d'éléments séparés par le `separator` puis compare `string` à chaque valeur de la liste. `string` n'est pas une expression régulière. Si `string` est égal à n'importe quel élément (en ignorant la casse), alors il renvoie la `found_val` correspondante. Si `string` contient des `separators`, elle est également traitée comme

[80] https://docs.python.org/3/library/string.html#formatstrings
[81] https://www.loc.gov/standards/iso639-2/php/code_list.php
[82] https://www.loc.gov/standards/iso639-2/php/code_list.php
[83] https://docs.python.org/3/library/re.html

une liste et chaque sous-valeur est vérifiée. Les paires `string` et `found_value` peuvent être répétées autant de fois que souhaité, ce qui permet de renvoyer des valeurs différentes selon la valeur de la chaîne. Si aucune des chaînes de caractères ne correspond, le système renvoie `not_found_value`. Les chaînes de caractères sont vérifiées dans l'ordre. La première correspondance est retournée.

- `subitems(start_index, end_index)` – Cette fonction sépare les listes d'éléments hiérarchiques de type balise, comme les genres. Elle interprète la valeur comme une liste d'éléments de type balise séparés par des virgules, où chaque élément est une liste séparée par des points. Elle retourne une nouvelle liste faite en extrayant de chaque élément les composants de `start_index` à `end_index`, puis en fusionnant les résultats ensemble. Les doublons sont supprimés. Le premier sous-élément d'une liste séparée par un point a un indice de zéro. Si un index est négatif, il est compté à partir de la fin de la liste. Dans un cas particulier, un end_index de zéro est supposé être la longueur de la liste.

  Exemples :

  - Supposons une colonne #genre contenant *A.B.C* :
    - `{#genre:subitems(0,1)}` renvoie « A »
    - `{#genre:subitems(0,2)}` renvoie « A.B »
    - `{#genre:subitems(1,0)}` renvoie « B.C »
  - Supposons une colonne #genre contenant « A.B.C, D.E » :
    - `{#genre:subitems(0,1)}` renvoie « A, D »
    - `{#genre:subitems(0,2)}` renvoie « A.B, D.E »

- `sublist(start_index, end_index, separator)` – interprète la valeur comme une liste d'éléments séparés par le `separator`, renvoyant une nouvelle liste composée des éléments de `start_index` à `end_index`. Le premier élément est le numéro zéro. Si un index est négatif, alors il compte à partir de la fin de la liste. Dans un cas spécial, un end_index de zéro est supposé être la longueur de la liste.

  Exemples en supposant que la colonne des étiquettes (qui est séparée par des virgules) contient « A, B ,C » :

  - `{tags:sublist(0,1,\,)}` renvoie « A »
  - `{tags:sublist(-1,0,\,)}` renvoie « C »
  - `{tags:sublist(0,-1,\,)}` renvoie « A, B »

- `swap_around_articles(separator)` – renvoie la valeur avec les articles déplacés à la fin. La valeur peut être une liste, dans ce cas chaque article de la liste est traité. Si la valeur est une liste, vous devez fournir le `séparateur`. Si aucun `séparateur` n'est fourni, alors la valeur est traitée comme étant une valeur unique, et non une liste. Les *articles* sont ceux utilisés par calibre pour générer le `title_sort`.

- `swap_around_comma()` – étant donné une valeur de la forme `B, A`, retourne `A B`. Ceci est très utile pour convertir des noms au format LN, FN en FN LN. S'il n'y a pas de virgule dans la valeur, la fonction retourne la valeur inchangée.

- `switch([pattern, value,]+ else_value)` – pour chaque paire `pattern, value`, vérifie si la valeur correspond à l'expression régulière `pattern` et si oui, retourne la `value` associée. Si aucun `pattern` ne correspond, `else_value` est retourné. Vous pouvez avoir autant de couples `pattern, value` que vous le souhaitez. La première correspondance est retournée.

- `test(text if not empty, text if empty)` – retourne `text if not empty` si la valeur n'est pas vide, sinon retourne `text if empty`.

- `titlecase()` – renvoie la valeur du champ en casse titre.

- `transliterate()` - Restitue une chaîne en alphabet latin représentant approximativement le son des mots de la chaîne source. Par exemple, si le champ source est `Фёдор Михáйлович Достоевский` la fonction renverra `Fiodor Mikhailovich Dostoievskii`."

- `uppercase()` – renvoie la valeur du champ en majuscule.

Utilisation des fonctions et du formatage dans le même modèle

Supposons que vous ayez une colonne personnalisée de nombres entiers, `##myint`, que vous voulez afficher avec des zéros en tête, comme dans `003`. Une façon de le faire est d'utiliser le format `0>3s`. Cependant, par défaut, si un nombre (entier ou flottant) est égal à zéro, la valeur est affichée sous la forme d'une chaîne vide, de sorte que les valeurs nulles produiront la chaîne vide, et non `000`. Si vous voulez voir les valeurs `000`, vous devez utiliser à la fois la chaîne de format et la fonction `ifempty` pour transformer la valeur vide en zéro. Le modèle serait

```
{#myint:0>3s:ifempty(0)}
```

Notez que vous pouvez également utiliser le préfixe et le suffixe. Si vous voulez que le numéro apparaisse sous la forme `[003]` ou `[000]`, utilisez alors le modèle

```
{#myint:0>3s:ifempty(0)|[|]}
```

10.3.6 Mode Programme Général

Le *Mode Programme Général* (*MPG*) remplace les *expressions modèles* par un programme écrit dans le *langage modèle*. La syntaxe de ce langage est définie par la grammaire suivante

```
program          ::= 'program:' expression_list
expression_list  ::= top_expression [ ';' top_expression ]*
top_expression   ::= or_expression
or_expression    ::= and_expression [ '||' and_expression ]*
and_expression   ::= not_expression [ '&&' not_expression ]*
not_expression   ::= [ '!' not_expression ]* | concatenate_expr
concatenate_expr::= compare_expr [ '&' compare_expr ]*
compare_expr     ::= add_sub_expr [ compare_op add_sub_expr ]
compare_op       ::= '==' | '!=' | '>=' | '>' | '<=' | '<' | 'in' | 'inlist' |
                     '==#' | '!=#' | '>=#' | '>#' | '<=#' | '<#'
add_sub_expr     ::= times_div_expr [ add_sub_op times_div_expr ]*
add_sub_op       ::= '+' | '-'
times_div_expr   ::= unary_op_expr [ times_div_op unary_op_expr ]*
times_div_op     ::= '*' | '/'
unary_op_expr    ::= [ add_sub_op unary_op_expr ]* | expression
expression       ::= identifier | constant | function | assignment | field_reference |
                     if_expr | for_expr | break_expr | continue_expr |
                     '(' expression_list ')' | function_def
field_reference  ::= '$' [ '$' ] [ '#' ] identifier
identifier       ::= id_start [ id_rest ]*
id_start         ::= letter | underscore
id_rest          ::= id_start | digit
constant         ::= " string " | ' string ' | number
function         ::= identifier '(' expression_list [ ',' expression_list ]* ')'
function_def     ::= 'def' identifier '(' top_expression [ ',' top_expression ]* ')' ':
↪'
                     expression_list 'fed'
assignment       ::= identifier '=' top_expression
if_expr          ::= 'if' condition 'then' expression_list
                     [ elif_expr ] [ 'else' expression_list ] 'fi'
condition        ::= top_expression
elif_expr        ::= 'elif' condition 'then' expression_list elif_expr | ''
for_expr         ::= for_list | for_range
for_list         ::= 'for' identifier 'in' list_expr
```

(suite sur la page suivante)

(suite de la page précédente)

```
                    [ 'separator' separator_expr ] ':' expression_list 'rof'
for_range       ::= 'for' identifier 'in' range_expr ':' expression_list 'rof'
range_expr      ::= 'range' '(' [ start_expr ',' ] stop_expr
                    [ ',' step_expr [ ',' limit_expr ] ] ')'
list_expr       ::= top_expression
break_expr      ::= 'break'
continue_expr   ::= 'continue'
separator_expr  ::= top_expression
start_expr      ::= top_expression
stop_expr       ::= top_expression
step_expr       ::= top_expression
limit_expr      ::= top_expression
```

Notes :

- une `top_expression` a toujours une valeur. La valeur d'une `expression_list` est la valeur de la dernière `top_expression` Par exemple, la valeur de la liste d'expressions `1;2;'foobar';3` est `3`.
- Dans un contexte logique, toute valeur non vide est `True`.
- Dans un contexte logique, la valeur vide est `False`.
- Les chaînes de caractères et les nombres peuvent être utilisés indifféremment. Par exemple, `10` et `'10'` sont la même chose.
- Les commentaires sont des lignes commençant par le caractère « #». Les commentaires commençant plus loin dans une ligne ne sont pas pris en charge.

Préséance de l'opérateur

L'ordre de préséance de l'opérateurs (ordre d'évaluation), du plus élevé (évalué en premier) au plus bas (évalué en dernier), est le suivant :

- Appels de fonctions, constantes, expressions entre parenthèses, expressions d'instructions, expressions d'affectation, références de champs.
- Les opérateurs unaires plus (+) et moins (−). Ces opérateurs s'évaluent de droite à gauche.

  Ces opérateurs et tous les autres opérateurs arithmétiques renvoient des entiers si l'expression donne une partie fractionnaire égale à zéro. Par exemple, si une expression renvoie `3.0`, elle est transformée en `3`.
- Multiplier (`*`) et diviser (`/`). Ces opérateurs sont associatifs et s'évaluent de gauche à droite. Utilisez les parenthèses si vous souhaitez modifier l'ordre d'évaluation.
- Additionner (+) et soustraire (−). Ces opérateurs sont associatifs et s'évaluent de gauche à droite.
- Comparaisons de nombres et de chaînes de caractères. Ces opérateurs renvoient `'1'` si la comparaison réussit, sinon la chaîne vide (`''`). Les comparaisons ne sont pas associatives : `a < b < c` est une erreur de syntaxe.
- Concaténation de chaînes de caractères (`&`). L'opérateur `&` renvoie une chaîne de caractères formée par la concaténation des expressions de gauche et de droite. Exemple : « “aaa” & “bbb”`` renvoie « “aaabbb”``. L'opérateur est associatif et s'évalue de gauche à droite.
- Unaire non logique (`!`). Cet opérateur renvoie `'1'` si l'expression est fausse (évaluée à la chaîne vide), sinon `''`.
- L'opérateur logique et (`&&`). Cet opérateur renvoie “1” si les deux expressions de gauche et de droite sont vraies, ou la chaîne vide `''` si l'une ou l'autre est fausse. Il est associatif, s'évalue de gauche à droite, et fait du court-circuitage[84].

[84] https://chortle.ccsu.edu/java5/Notes/chap40/ch40_2.html

- Logique ou (||). Cet opérateur renvoie '1' si l'expression de gauche ou de droite est vraie, ou '' si les deux sont fausses. Il est associatif, s'évalue de gauche à droite, et fait du court-circuitage[85]. C'est un *or inclusif*, qui renvoie « 1 » si les deux expressions de gauche et de droite sont vraies.

Références des champs

Un « field_reference » est évalué à la valeur du champ de métadonnées nommé par le nom de la consultation qui suit le « $ » ou le « $$ ». L'utilisation de $ équivaut à l'utilisation de la fonction field(). L'utilisation de $$ équivaut à l'utilisation de la fonction raw_field. Exemples

```
* $authors ==> field('authors')
* $#genre ==> field('#genre')
* $$pubdate ==> raw_field('pubdate')
* $$#my_int ==> raw_field('#my_int')
```

Expressions if

Les expressions If évaluent d'abord la condition. Si la condition est True (une valeur non vide), alors expression_list de la clause then est évaluée. Si elle est False, expression_list de la clause elif ou else est évaluée si elle est présente. Les parties if et else sont facultatives. Les mots if ``, ``then, elif ``, ``else et fi sont réservés ; vous ne pouvez pas les utiliser comme noms d'identifiants. Vous pouvez mettre des retours à la ligne et des espaces partout où cela a un sens. La condition est une top_expression et non une expression_list ; les points-virgules ne sont pas autorisés. Les expression_list sont des séquences de top_expressions séparées par des points-virgules. Une expression if renvoie le résultat de la dernière top_expressions de la top_expression évaluée, ou la chaîne vide si aucune liste d'expression n'a été évaluée.

Exemples :

```
* program: if field('series') then 'yes' else 'no' fi
* program:
      if field('series') then
          a = 'yes';
          b = 'no'
      else
          a = 'no';
          b = 'yes'
      fi;
      strcat(a, '-', b)
```

Exemple de if imbriqué :

```
program:
  if field('series') then
    if check_yes_no(field('#mybool'), '', '', '1') then
      'yes'
    else
      'no'
    fi
  else
    'no series'
  fi
```

Comme dit plus haut, un if produit une valeur. Cela signifie que tous les cas suivants sont équivalents

```
* program: if field('series') then 'foo' else 'bar' fi
* program: if field('series') then a = 'foo' else a = 'bar' fi; a
* program: a = if field('series') then 'foo' else 'bar' fi; a
```

[85] https://chortle.ccsu.edu/java5/Notes/chap40/ch40_2.html

Comme dernier exemple, ce programme retourne la valeur de la colonne `series` si le livre a une série, sinon la valeur de la colonne `title`

```
program: field(if field('series') then 'series' else 'title' fi)
```

Expressions for

L'expression `for` itère sur une liste de valeurs, en les traitant une par une. L'expression « list_expression » doit être évaluée soit par un nom de champ de métadonnées, par exemple `tags` ou `#genre`, soit par une liste de valeurs. Si le résultat est un `lookup name` valide, alors la valeur du champ est récupérée et le séparateur spécifié pour ce type de champ est utilisé. Si le résultat n'est pas un nom de référence valide, il est supposé être une liste de valeurs. La liste est supposée être séparée par des virgules, sauf si le mot-clé optionnel `separator` est fourni, auquel cas les valeurs de la liste doivent être séparées par le résultat de l'évaluation de l'expression `separator_expr`. Un séparateur ne peut pas être utilisé si la liste est générée par `range()`. Chaque valeur de la liste est assignée à la variable spécifiée puis la `expression_list` est évaluée. Vous pouvez utiliser `break` pour sortir de la boucle, et `continue` pour revenir au début de la boucle pour la prochaine itération.

Exemple : Ce modèle supprime le premier nom hiérarchique pour chaque valeur dans le genre (`#genre`), en construisant une liste avec les nouveaux noms

```
program:
  new_tags = '';
  for i in '#genre':
    j = re(i, '^.*?\.(.*)$', '\1');
    new_tags = list_union(new_tags, j, ',')
  rof;
  new_tags
```

Si le Genre original est *Histoire.Militaire, Science Fiction.Histoire alternative, Lisez-moi*, alors le modèle renvoie *Militaire, Histoire alternative, Lisez-moi*. Vous pouvez utiliser ce modèle dans le calibre *Modifier les métadonnées par lot* → *Rechercher & remplacer* avec *Rechercher après* défini à `modèe` pour supprimer le premier niveau de la hiérarchie et attribuer la valeur résultante à Genre.

Note : la dernière ligne du modèle, `new_tags`, n'est pas strictement nécessaire dans ce cas car `for` renvoie la valeur de la dernière top_expression de la liste d'expressions. La valeur d'une affectation est la valeur de son expression, donc la valeur de l'instruction `for` est ce qui a été affecté à `new_tags`.

Définition de la fonction

Si vous avez du code dans un modèle qui se répète, vous pouvez mettre ce code dans une fonction locale. Le mot clé « def » commence la définition. Il est suivi par le nom de la fonction, la liste des arguments, puis le code de la fonction. La définition de la fonction se termine par le mot-clé ``fed"".

Les arguments sont positionnels. Lorsqu'une fonction est appelée, les arguments fournis sont comparés de gauche à droite aux paramètres définis, la valeur de l'argument étant attribuée au paramètre. C'est une erreur de fournir plus d'arguments que de paramètres définis. Les paramètres peuvent avoir des valeurs par défaut, comme « a = 25 ». Si un argument n'est pas fourni pour ce paramètre, alors la valeur par défaut est utilisée, sinon le paramètre est défini par la chaîne vide.

L'instruction `return` peut être utilisée dans une fonction locale.

Une fonction doit être définie avant de pouvoir être utilisée.

Exemple : Ce modèle calcule une durée approximative en années, mois et jours à partir d'un nombre de jours. La fonction `to_plural()` met en forme les valeurs calculées. Notez que l'exemple utilise également l'opérateur `&` :

```
program:
      days = 2112;
      years = floor(days/360);
      months = floor(mod(days, 360)/30);
```

(suite sur la page suivante)

(suite de la page précédente)

```
    days = days - ((years*360) + (months * 30));

    def to_plural(v, str):
            if v == 0 then return '' fi;
            return v & ' ' & (if v == 1 then str else str & 's' fi) & ' '
    fed;

    to_plural(years, 'year') & to_plural(months, 'month') & to_plural(days,'day')
```

Opérateurs relationnels

Les opérateurs relationnels renvoient `'1'` si la comparaison est vraie, sinon la chaîne vide (“”).

Il existe deux formes d'opérateurs relationnels : les comparaisons de chaînes de caractères et les comparaisons numériques.

Les comparaisons de chaînes de caractères effectuent une comparaison insensible à la casse en utilisant l'ordre lexical. Les opérateurs de comparaison de chaînes de caractères pris en charge sont ==, !=, <, <=, >, >=, `in`, et `inlist`. Pour l'opérateur `in`, le résultat de l'expression de gauche est interprété comme un motif d'expression régulière. L'opérateur `in` est True si la valeur de l'expression régulière de gauche correspond à la valeur de l'expression de droite. L'opérateur `inlist` est vrai si l'expression régulière de gauche correspond à l'un des éléments de la liste de droite, les éléments de la liste étant séparés par des virgules. Les correspondances sont insensibles à la casse.

Les opérateurs de comparaison numérique sont ==#, !=#, <#, <=#, >#, >=#. Les expressions de gauche et de droite doivent être évaluées par des valeurs numériques, à deux exceptions près : la chaîne de caractères « None » (champ indéfini) et la chaîne de caractères vide sont évaluées par la valeur zéro.

Exemples :

- `program:field('series') == 'foo'` renvoie `'1'` si la série du livre est “foo”, sinon `''`.
- `Program: 'f.o' in field('series')` renvoie *“1”* si la série du livre correspond à l'expression régulière `f.o` (par exemple, *foo*, *Off Onyx*, etc.), sinon `''`.
- `Program: 'science' inlist field('#genre')` renvoie `'1'` si l'un des genres du livre correspond à l'expression régulière `science`, par exemple, *Science*, *Histoire de la science*, *Science Fiction* etc., sinon `''`.
- `Program: '^science$' inlist field('#genre')` renvoie *“1”* si l'un des genres du livre correspond exactement à l'expression régulière `^science$`, par exemple, *Science*. Les genres *Histoire des sciences* et *Science Fiction* ne correspondent pas. S'il n'y a pas de correspondance, elle renvoie `''`.
- `Program: if field('series') != 'foo' then 'bar' else 'mumble' fi` renvoie `'bar'` si la série du livre n'est pas `foo`. Sinon, renvoie `mumble`.
- `Program: if field('series') == 'foo' || field('series') == '1632' then 'yes' else 'no' fi` renvoie `'yes'` si séries existe soit `'foo'` soit `'1632'`, sinon *“no”*.
- `program: if '^(foo|1632)$' in field('series') then 'yes' else 'no' fi` renvoie `'yes'` si la série existe soit `'foo'` ou `'1632'`, sinon `'no'`.
- `program: if '11' > '2' then 'yes' else 'no' fi` *renvoie* “”no”“ parce que l'opérateur > fait une comparaison lexicale.
- `Program: if 11 ># 2 then 'yes' else 'no' fi` renvoie `'yes'` parce que l'opérateur ># effectue une comparaison numérique.

Fonctions supplémentaires disponibles

Les fonctions suivantes sont disponibles en plus de celles décrites dans Mode fonction unique.

Dans *MPG*, les fonctions décrites dans *Mode Fonction Simple* requièrent toutes un premier paramètre supplémentaire spécifiant la valeur sur laquelle opérer. Tous les paramètres sont des listes d'expressions (voir la grammaire ci-dessus).

- `add(x [, y]*)` – renvoie la somme de ses arguments. Lance une exception si un argument n'est pas un nombre. Dans la plupart des cas, vous pouvez utiliser l'opérateur + à la place de cette fonction.
- `and(value [, value]*)` – retourne la chaîne « 1 » si toutes les valeurs ne sont pas vides, sinon retourne la chaîne vide. Vous pouvez avoir autant de valeurs que vous le souhaitez. Dans la plupart des cas, vous pouvez utiliser l'opérateur `&&` à la place de cette fonction. Une raison de ne pas remplacer `and` par `&&` est que le court-circuitage peut modifier les résultats à cause d'effets secondaires. Par exemple, `et(a='',b=5)` fera toujours les deux affectations, alors que l'opérateur `&&` ne fera pas la seconde.
- `assign(id, val)` – affecte `val` à `id`, puis retourne `val`. `id` doit être un identifiant, pas une expression. Dans la plupart des cas, vous pouvez utiliser l'opérateur = à la place de cette fonction.
- `approximate_formats()` – renvoie une liste, séparée par des virgules, des formats associés au livre. Il n'y a aucune garantie que la liste soit correcte, bien qu'elle le soit probablement. Cette fonction et d'autres fonctions à zéro paramètre peuvent être appelées en mode programme modèle (voir ci-dessous) en utilisant le modèle `{:'approximate_formats()'}`. Notez que les noms de formats résultants sont toujours en majuscules, comme dans EPUB. La fonction `approximate_formats()` est nettement plus rapide que les fonctions `formats_...` présentées ci-dessous.
- `author_links(val_separator, pair_separator)` – renvoie une chaîne contenant une liste d'auteurs et les valeurs des liens de ces auteurs sous la forme

```
author1 val_separator author1_link pair_separator author2 val_separator author2_
↪link etc.
```

  Un auteur est séparé de sa valeur de lien par la chaîne `val_separator` sans espaces ajoutés. Les paires `author:linkvalue` sont séparées par la chaîne de caractères `pair_separator`, sans espaces ajoutés. C'est à vous de choisir des chaînes de séparation qui n'apparaissent pas dans les noms d'auteurs ou les liens. Un auteur est inclus même si le lien de l'auteur est vide.

- `author_sorts(val_separator)` – renvoie une chaîne contenant une liste des valeurs de tri de l'auteur pour les auteurs du livre. Le tri est celui des métadonnées auteur (différente de l'author_sort dans les livres). La liste renvoyée a la forme `author sort 1 val_separator author sort 2` etc. Les valeurs de tri auteur dans cette liste sont dans le même ordre que les auteurs du livre. Si vous voulez les espaces autour du `val_separator` incluez les alors dans la chaîne `val_separator`.
- `book_count(query, use_vl)` – renvoie le nombre de livres trouvés en cherchant pour `query`. Si `use_vl` est `0` (zéro) alors les bibliothèques virtuelles sont ignorées. Cette fonction et son pendant `book_values()` sont particulièrement utiles dans les recherches de modèles, en supportant les recherches qui combinent les informations de plusieurs livres, comme la recherche de séries avec un seul livre. Elle ne peut pas être utilisée dans les colonnes composites, sauf si le paramètre « allow_template_database_functions_in_composites » a la valeur True. Elle ne peut être utilisée que dans l'interface graphique.

  Par exemple, ce modèle de recherche utilise cette fonction et son pendant pour trouver toutes les séries ne comportant qu'un seul livre :

  1) Définir un modèle enregistré (en utilisant *Préférences → Avancé → Modèles de fonctions*) nommé `série_uniquement_un_livre` (le nom est arbitraire). Le modèle est

```
program:
    vals = globals(vals='');
    if !vals then
        all_series = book_values('series', 'series:true', ',', 0);
        for series in all_series:
            if book_count('series:="' & series & '"', 0) == 1 then
                vals = list_join(',', vals, ',', series, ',')
            fi
        rof;
        set_globals(vals)
```

(suite sur la page suivante)

(suite de la page précédente)

```
    fi;
    str_in_list(vals, ',', $series, 1, '')
```

La première fois que le modèle s'exécute (le premier livre vérifié), il enregistre les résultats des recherches dans la base de données dans une variable de modèle « globale » appelée « vals ». Ces résultats sont utilisés pour vérifier les livres suivants sans avoir à refaire les recherches.

2) Utiliser le modèle enregistré dans une recherche de modèle

```
template:"program: series_only_one_book()#@#:n:1"
```

L'utilisation d'un modèle stocké au lieu de mettre le modèle dans la recherche élimine les problèmes causés par l'obligation d'échapper les guillemets dans les expressions de recherche.

- `book_values(column, query, sep, use_vl)` – renvoie une liste des valeurs uniques contenues dans la colonne `column` (un nom de recherche), séparées par `sep`, dans les livres trouvés en recherchant `query`. Si `use_vl` est `0` (zéro), alors les bibliothèques virtuelles sont ignorées. Cette fonction et son pendant `book_count()` sont particulièrement utiles dans les recherches de modèles, en supportant les recherches qui combinent les informations de plusieurs livres, comme la recherche de séries avec un seul livre. Elle ne peut pas être utilisée dans des colonnes composites, sauf si le paramètre « allow_template_database_functions_in_composites » a la valeur True. Elle ne peut être utilisée que dans l'interface graphique.
- `booksize()` – renvoie la valeur du champ "size" de calibre. Renvoie "" s'il n'y a pas de formats.
- `check_yes_no(field_name, is_undefined, is_false, is_true)` – vérifie si la valeur du champ oui/non nommé par le nom de la référence externe `field_name` est l'une des valeurs spécifiées par les paramètres, renvoyant `'yes'` si une correspondance est trouvée sinon renvoyant la chaîne vide. Définissez le paramètre `is_undefined`, `is_false` ou `is_true` à 1 (le nombre) pour vérifier cette condition, sinon définissez-le à 0. Exemple :

  `check_yes_no("#bool", 1, 0, 1)` renvoie `'Yes'` si le champ oui/non `#bool` est soit True soit non-défini (ni True ni False).

  Plus d'un élément parmi « is_undefined « , « is_false « ou « is_true « peut être mis à 1.
- `ceiling(x)` – renvoie le plus petit entier supérieur ou égal à `x`. Envoie une exception si `x` n'est pas un nombre.
- `character(character_name)` – renvoie le caractère désigné par character_name. Par exemple, `character('newline')` renvoie un caractère de nouvelle ligne (`'\n'`). Les noms de caractères pris en charge sont `newline`, `return`, `tab`, et `backslash`.
- `cmp(x, y, lt, eq, gt)` – compare `x` et `y` après les avoir convertis en nombres. Renvoie `lt` si `x <# y`, `eq` si `x ==# y`, sinon `gt`. Cette fonction peut généralement être remplacée par l'un des opérateurs de comparaison numérique (==#, <#, >#, etc).
- `connected_device_name(storage_location_key)` – si un périphérique est connecté alors retourne le nom du périphérique, sinon retourne la chaîne vide. Chaque emplacement de stockage sur un périphérique peut avoir un nom différent. Les noms `storage_location_key` sont `'main'`, `'carda'` et `'cardb'`. Cette fonction ne fonctionne que dans l'interface graphique.
- `connected_device_uuid(storage_location_key)` – si un périphérique est connecté alors retourne l'uuid du périphérique (id unique), sinon retourne la chaîne vide. Chaque emplacement de stockage sur un périphérique a un uuid différent. Les noms `storage_location_key` sont `'main'`, `'carda'` et `'cardb'`. Cette fonction ne fonctionne que dans l'interface graphique.
- `current_library_name()` – renvoie le dernier nom sur le chemin de la bibliothèque calibre actuelle.
- `current_library_path()` – renvoie le chemin complet vers la bibliothèque calibre actuelle.

- current_virtual_library_name() – retourne le nom de la bibliothèque virtuelle actuelle s'il y en a une, sinon la chaîne vide. La casse du nom de la bibliothèque est préservée. Exemple : « program :current_virtual_library_name() ». Cette fonction ne fonctionne que dans l'Interface Graphique.

- `date_arithmetic(date, calc_spec, fmt)` – Calcule une nouvelle date à partir de `date` en utilisant `calc_spec`. Retourne la nouvelle date formatée selon l'option `fmt` : si elle n'est pas fournie, le résultat sera au format ISO. calc_spec est une chaîne de caractères formée par la concaténation de paires de `vW` (valueWhat) où `v` est un nombre éventuellement négatif et W est l'une des lettres suivantes :

  - `s` : ajoute `v` secondes à `date`
  - `m` : ajoute `v` minutes à `date`
  - `h` : ajoute `v` heures à `date`
  - `d` : ajoute `v` jours à `date`
  - `w` : ajoute `v` semaines à `date`
  - `y` : ajoute `v` années à `date`, où une année compte 365 jours.

  Exemple : `'1s3d-1m'` ajoutera 1 seconde, ajoutera 3 jours, et soustraira 1 minute de `date`.

- `days_between(date1, date2)` – renvoie le nombre de jours entre `date1` et `date2`. Le nombre est positif si `date1` est plus grand que `date2`, sinon négatif. Si `date1` ou `date2` ne sont pas des dates, la fonction renvoie une chaîne vide.

- `divide(x, y)` – renvoie `x / y`. Lance une exception si `x` ou `y` ne sont pas des nombres. Cette fonction peut généralement être remplacée par l'opérateur `/`.

- `eval(string)` – évalue la chaîne comme un programme, en passant les variables locales. Cela permet d'utiliser le processeur de modèles pour construire des résultats complexes à partir de variables locales. En *Mode Modèle de Programme* (page 190), comme les caractères *{* et *}* sont interprétés avant l'évaluation du modèle, vous devez utiliser *[[*] pour le caractère *{* et *]]* pour le caractère `}`. Ils sont convertis automatiquement. Notez également que les préfixes et les suffixes (la syntaxe |*prefix*|*suffix*) ne peuvent pas être utilisés dans l'argument de cette fonction lors de l'utilisation de *Mode Modèle de Programme* (page 190).

- `extra_file_size(file_name)` – retourne la taille en octets du fichier supplémentaire `file_name` dans le dossier `data/` du livre s'il existe, sinon `-1`. Voir aussi les fonctions `has_extra_files()`, `extra_file_names()` et `extra_file_modtime()`. Cette fonction ne peut être utilisée que dans l'interface graphique.

- `extra_file_modtime(file_name, format_string)` – retourne l'heure de modification du fichier supplémentaire `file_name` dans le dossier `data/` du livre s'il existe, sinon `-1`. La date de modification est formatée selon `format_spec` (voir `format_date()` pour plus de détails). Si `format_spec` est une chaîne vide, elle retourne l'heure de modification sous la forme d'un nombre de secondes en virgule flottante depuis Epoch. Voir aussi les fonctions `has_extra_files()`, `extra_file_names()` et `extra_file_size()`. L'Epoch dépend du système d'exploitation. Cette fonction ne peut être utilisée que dans l'interface graphique.

- `extra_file_names(sep [, pattern])` – retourne une liste séparée de `sep` de fichiers supplémentaires dans le dossier `data/` du livre. Si le paramètre optionnel `pattern`, une expression régulière, est fourni, alors la liste est filtrée sur les fichiers qui correspondent à `pattern`. La correspondance est insensible à la casse. Voir aussi les fonctions `has_extra_files()`, `extra_file_modtime()` et `extra_file_size()`. Cette fonction ne peut être utilisée que dans l'interface graphique.

- `field(lookup_name)` – renvoie la valeur du champ de métadonnées avec le nom de recherche `lookup_name`.

- `field_exists(field_name)` – vérifie si un champ (colonne) avec le nom de recherche `field_name` existe, retournant *"1"* si c'est le cas et la chaîne vide si non.

- `finish_formatting(val, fmt, prefix, suffix)` – applique le format, le préfixe et le suffixe à une valeur de la même manière que dans un modèle comme `{series_index:05.2f| - |- }`. Cette fonction est fournie pour faciliter la conversion des modèles complexes single-function- ou modèles template-program-mode en modèles *MGP*. Par exemple, le programme suivant produit la même sortie que le modèle ci-dessus

```
program: finish_formatting(field("series_index"), "05.2f", " - ", " - ")
```

  Un autre exemple : pour le modèle `{series:re(([^\s])[^\s]+(\s|$),\1)}{series_index:0>2s| - | - }{title}` utilisez :

```
program:
  strcat(
    re(field('series'), '([^\s])[^\s]+(\s|$)', '\1'),
       finish_formatting(field('series_index'), '0>2s', ' - ', ' - '),
       field('title')
  )
```

- `first_matching_cmp(val, [ cmp, result, ]* else_result)` – compare `val < cmp` en séquence, en retournant le résultat associé à la première comparaison qui réussit. Retourne else_result si aucune comparaison ne réussit. Exemple

```
i = 10;
first_matching_cmp(i,5,"small",10,"middle",15,"large","giant")
```

  renvoie `"grand"`. Le même exemple avec une première valeur de 16 renvoie `"géant"`.

- `first_non_empty(value [, value]*)` – retourne la première valeur non vide. Si toutes les valeurs sont vides, alors la chaîne vide est retournée. Vous pouvez avoir autant de valeurs que vous voulez.
- `floor(x)` – renvoie le plus grand entier inférieur ou égal à `x`. Envoie une exception si `x` n'est pas un nombre.
- `format_date(val, format_string)` – formate la valeur, qui doit être une chaîne date, en utilisant format_string, retournant une chaîne. Les codes de formatage sont :
  - `d` : le jour sous la forme d'un nombre sans zéro initial (1 à 31)
  - `dd` : le jour sous la forme d'un nombre avec un zéro initial (1 à 31)
  - `ddd` : le nom abrégé du jour localisé (par exemple de « Lun » à « Dim »).
  - `dddd` : le nom long du jour localisé (par exemple de « lundi » à « dimanche »).
  - `M` : le mois sous la forme d'un nombre sans zéro initial (1 à 12).
  - `MM` : le mois sous la forme d'un nombre avec un zéro initial (1 à 12).
  - `MMM` : le nom abrégé du mois localisé (par exemple de « Jan » à « Déc »).
  - `MMMM` : le nom long du mois localisé (par exemple de « janvier » à « décembre »).
  - `yy` : l'année sous la forme d'un nombre à deux chiffres (00 à 99).
  - `yyyy` : l'année sous la forme d'un nombre à quatre chiffres.
  - `h` : les heures sans le premier 0 (0 à 11 ou 0 à 23, selon am/pm)
  - `hh` : les heures avec le premier 0 (00 à 11 ou 00 à 23, selon am/pm)
  - `m` : les minutes sans le premier 0 (0 à 59)
  - `mm` : les minutes avec le premier 0 (00 à 59)
  - `s` : les secondes sans le premier 0 (0 à 59)
  - `ss` : les secondes avec un 0 en tête (00 à 59)

  - `ap` : utilise une horloge de 12 heures au lieu d'une horloge de 24 heures, avec "ap" remplacé par la chaîne localisée pour am ou pm.
  - `AP` : utilise une horloge de 12 heures au lieu d'une horloge de 24 heures, avec "AP" remplacé par la chaîne localisée pour AM ou PM.
  - `iso` : la date avec l'heure et le fuseau horaire. Ce doit être le seul format présent.
  - `to_number` : convertit la date et l'heure en un nombre à virgule flottante (un *timestamp*)
  - `from_number` : convertit un nombre à virgule flottante (un *timestamp*) en une date formatée `iso`. Si vous souhaitez un format de date différent, ajoutez la chaîne de formatage souhaitée après `from_number` et un deux-points (`:`). Exemple : `from_number:MMM dd yyyy`.

  Vous pouvez obtenir des résultats inattendus si la date que vous formatez contient des noms de mois localisés, ce qui peut se produire si vous avez modifié les ajustements de format de date pour qu'ils contiennent `MMMM`. Dans ce cas, au lieu d'utiliser la fonction `field()` comme dans

```
format_date(field('pubdate'), 'yyyy')
```

  utiliser la fonction `raw_field()` comme dans

```
format_date(raw_field('pubdate'), 'yyyy')
```

- `format_date_field(field_name, format_string)` – formate la valeur du champ `field_name`, qui doit être le nom d'un champ date, standard ou personnalisé. Voir `format_date()` pour les codes de formatage. Cette fonction est beaucoup plus rapide que format_date et doit être utilisée lorsque vous formatez la valeur d'un champ (colonne). Elle ne peut pas être utilisée pour des dates calculées ou des dates dans des variables de type chaîne. Exemples :

```
format_date_field('pubdate', 'yyyy.MM.dd')
format_date_field('#date_read', 'MMM dd, yyyy')
```

- `formats_modtimes(date_format_string)` – renvoie une liste séparée par des virgules d'éléments séparés par des deux points `FMT:DATE` représentant les heures de modification des formats d'un livre. Le paramètre `date_format_string` spécifie comment la date doit être formatée. Voir la fonction `format_date()` pour plus de détails. Vous pouvez utiliser la fonction `select` pour obtenir l'heure de modification d'un format spécifique. Notez que les noms de format sont toujours en majuscules, comme dans EPUB.
- `formats_paths()` – retourne une liste d'éléments séparés par des virgules et des deux-points `FMT:PATH` donnant le chemin complet vers les formats d'un livre. Vous pouvez utiliser la fonction select pour obtenir le chemin d'accès à un format spécifique. Notez que les noms de format sont toujours en majuscules, comme dans EPUB.
- `formats_sizes()` – renvoie une liste d'éléments `FMT:SIZE` séparés par des virgules, donnant les tailles en octets des formats d'un livre. Vous pouvez utiliser la fonction select pour obtenir la taille d'un format spécifique. Notez que les noms de format sont toujours en majuscules, comme dans EPUB.
- `fractional_part(x)` – renvoie la valeur après la virgule. Par exemple, `fractional_part(3.14)` renvoie `0.14`. Envoie une exception si `x` n'est pas un nombre.
- `get_link(field_name, field_value)` – récupère le lien pour le champ `field_name` avec la valeur `field_value`. S'il n'y a pas de lien attaché, il renvoie "". Exemple :
- Ce qui suit renvoie le lien attaché à l'étiquette `Fiction`.

```
get_link('tags', 'Fiction')
```

- Ce modèle crée une liste de liens pour tous les tags associés à un livre sous la forme `valeur:lien, …` :

```
program:
 ans = '';
 for t in $tags:
     l = get_link('tags', t);
     if l then
         ans = list_join(', ', ans, ',', t & ':' & get_link('tags', t), ',')
     fi
 rof;
 ans
```

- has_cover() – retourne 'Yes' si le livre a une couverture, sinon la chaîne vide.
- has_extra_files([pattern]) – retourne le nombre de fichiers supplémentaires, sinon "" (la chaîne vide). Si le paramètre optionnel pattern (une expression régulière) est fourni, alors la liste est filtrée sur les fichiers qui correspondent à pattern avant que les fichiers ne soient comptés. La correspondance est insensible à la casse. Voir aussi les fonctions extra_file_names(), extra_file_size() et extra_file_modtime(). Cette fonction ne peut être utilisée que dans l'interface graphique.
- identifier_in_list(val, id_name [, found_val, not_found_val]) – traite val comme une liste d'identifiants séparés par des virgules. Un identifiant a le format id_name:value. Le paramètre id_name est le texte de l'identifiant à rechercher, soit id_name soit id_name:regexp. Dans le premier cas, la recherche s'effectue s'il existe un identifiant correspondant à ce nom d'utilisateur. Dans le second cas, la recherche se fait si le id_name correspond à un identifiant et que l'expression rationnelle correspond à la valeur de l'identifiant. Si found_val et not_found_val sont fournis, alors s'il y a une correspondance, il retourne found_val, sinon il retourne not_found_val. Si found_val et not_found_val ne sont pas fournis, alors s'il y a une correspondance, alors la paire identifier:value est retournée, sinon la chaîne vide ("").
- is_marked() – vérifie si le livre est *marqué* dans calibre. S'il l'est, retourne la valeur de la marque, soit true' (minuscule), soit une liste de marques nommées, séparées par des virgules.. Si le livre n'est pas marqué, retourne " (chaîne vide). Cette fonction ne fonctionne que dans l'interface graphique.
- language_codes(lang_strings) – retourne les codes de langue[86] pour les noms de langue passés dans *lang_strings*. Les chaînes de caractères doivent être dans la langue de la locale courante. Lang_strings est une liste séparée par des virgules.
- list_contains(value, separator, [pattern, found_val,]* not_found_val) – (Alias de in_list) Interprétant la valeur comme une liste d'éléments séparés par le separator, évalue le pattern par rapport à chaque valeur de la liste. Si le pattern correspond à une valeur, il renvoie found_val, sinon il renvoie not_found_val. Le pattern et la found_value peuvent être répétés autant de fois que vous le souhaitez, ce qui permet de renvoyer différentes valeurs en fonction de la recherche. Les motifs sont vérifiés dans l'ordre. La première correspondance est retournée. Alias : in_list(), list_contains().
- list_count(value, separator) – interprète value comme une liste d'éléments séparés par separator, et retourne le nombre d'éléments dans la liste. Alias : count(), list_count()
- list_count_matching(list, pattern, separator) – interprète list comme une liste d'éléments séparés par separator, en retournant le nombre d'éléments dans la liste qui correspondent à l'expression régulière pattern. Alias : list_count_matching(), count_matching()
- list_difference(list1, list2, separator) – restitue une liste en supprimant de list1 tout élément trouvé dans list2 ``en utilisant une comparaison insensible à la casse. Les éléments dans ``list1 et list2 sont séparés par le séparateur, comme le sont les éléments dans la liste restituée.
- list_equals(list1, sep1, list2, sep2, yes_val, no_val) –retourne yes_val si list1 et list2 contiennent les mêmes éléments, sinon renvoie no_val. Les éléments sont déterminés en éclatant

[86] https://www.loc.gov/standards/iso639-2/php/code_list.php

chaque liste à l'aide du séparateur approprié (`sep1` ou `sep2`). L'ordre des éléments dans la liste n'est pas pris en compte. La comparaison est insensible à la casse.

- `list_intersection(list1, list2, separator)` – retourne une liste faite en retirant de `list1` tout élément non trouvé dans `list2`, en utilisant une comparaison insensible à la casse. Les éléments de `list1` et `list2` sont séparés par un séparateur, comme le sont les éléments de la liste retournée.
- `list_join(with_separator, list1, separator1 [, list2, separator2]*)` – retourne une liste faite en joignant les éléments des listes sources (`list1` etc) en utilisant `with_separator` entre les éléments de la liste résultante. Les éléments de chaque `liste[123...]` source sont séparés par le `séparateur[123...]` associé. Une liste peut contenir zéro valeur. Il peut s'agir d'un champ comme `publisher` qui n'a qu'une seule valeur, c'est à dire une liste d'un seul élément. Les doublons sont supprimés en utilisant une comparaison insensible à la casse. Les éléments sont retournés dans l'ordre où ils apparaissent dans les listes sources. Si les éléments des listes ne diffèrent que par la casse des lettres, c'est la dernière qui est utilisée. Tous les séparateurs peuvent comporter plus d'un caractère.

  Exemple :

```
program:
  list_join('#@#', $authors, '&', $tags, ',')
```

  Vous pouvez utiliser `list_join` sur les résultats des appels précédents à `list_join` comme suit

```
program:
  a = list_join('#@#', $authors, '&', $tags, ',');
  b = list_join('#@#', a, '#@#', $#genre, ',', $#people, '&', 'some value', ',')
```

  Vous pouvez utiliser des expressions pour générer une liste. Par exemple, supposons que vous vouliez des éléments pour `auteurs` et `#genre`, mais avec le genre changé en mot « Genre : « suivi de la première lettre du genre, c'est-à-dire que le genre « Fiction » devient « Genre : F ». Ce qui suit fera l'affaire

```
program:
  list_join('#@#', $authors, '&', list_re($#genre, ',', '^(.).*$', 'Genre: \1'),
↪',')
```

- `list_re(src_list, separator, include_re, opt_replace)` – Construit une liste en séparant d'abord `src_list` en éléments en utilisant le caractère `separator`. Pour chaque élément de la liste, vérifiez s'il correspond à `include_re`. Si c'est le cas, ajoutez-le à la liste à retourner. Si `opt_replace` n'est pas une chaîne vide, il faut appliquer le remplacement avant d'ajouter l'élément à la liste retournée.
- `list_re_group(src_list, separator, include_re, search_re [, template_for_group]*)` – Comme list_re sauf que les remplacements ne sont pas optionnels. Elle utilise `re_group(item, search_re, template ...)` pour effectuer les remplacements.
- `list_remove_duplicates(list, separator)` – retourne une liste faite en supprimant les éléments en double dans `list`. Si les éléments ne diffèrent que par la casse, le dernier est retourné. Les éléments de `list` sont séparés par `separator`, comme le sont les éléments de la liste retournée.
- `list_sort(list, direction, separator)` – retourne `list` triée en utilisant un tri lexical insensible à la casse. Si `direction` est zéro, `list` est triée de manière ascendante, sinon de manière descendante. Les éléments de la liste sont séparés par le `separator`, comme le sont les éléments de la liste retournée.
- `list_split(list_val, sep, id_prefix)` – divise `list_val` en valeurs séparées en utilisant `sep`, puis affecte les valeurs à des variables locales nommées `id_prefix_N` où N est la position de la valeur dans la liste. Le premier élément a la position 0 (zéro). La fonction renvoie le dernier élément de la liste.

  Exemple :

```
list_split('one:two:foo', ':', 'var')
```

est équivalent à

```
var_0 = 'one';
var_1 = 'two';
var_2 = 'foo
```

- `list_union(list1, list2, separator)` – retourne une liste faite en fusionnant les éléments de `list1` et `list2`, en supprimant les éléments en double en utilisant une comparaison insensible à la casse. Si les éléments diffèrent selon la casse, c'est celui de la « list1 » qui est utilisé. Les éléments de « list1 » et « list2 » sont séparés par un « separator », tout comme les éléments de la liste retournée. Alias : `merge_lists()`, `list_union()`
- `mod(x, y)` – renvoie le `floor` du reste de `x / y`. Lance une exception si `x` ou `y` n'est pas un nombre.
- `multiply(x [, y]*)` – renvoie le produit de ses arguments. Lance une exception si l'un des arguments n'est pas un nombre. Cette fonction peut généralement être remplacée par l'opérateur `*`.
- `not(valeur)` – retourne la chaîne « 1 » si la valeur est vide, sinon retourne la chaîne vide. Cette fonction peut généralement être remplacée par l'opérateur unaire not (`!`).
- `ondevice()` – retourne la chaîne `Yes'` si `ondevice` est défini, sinon retourne la chaîne vide.
- `or(value [, value]*)` – retourne la chaîne `'1'` si une valeur n'est pas vide, sinon retourne la chaîne vide. Vous pouvez avoir autant de valeurs que vous le souhaitez. Cette fonction peut généralement être remplacée par l'opérateur `||`. Une raison pour laquelle elle ne peut pas être remplacée est que le court-circuitage modifiera les résultats à cause d'effets secondaires.
- `print(a [, b]*)` – imprime les arguments sur la sortie standard. A moins que vous ne lanciez calibre depuis la ligne de commande (`calibre-debug -g`), la sortie ira dans un trou noir. La fonction `print` retourne toujours son premier argument.
- `range(start, stop, step, limit)` – renvoie une liste de nombres générés en bouclant sur la plage spécifiée par les paramètres start, stop, et step, avec une longueur maximale de limit. La première valeur produite est « start ». Les valeurs suivantes sont `next_v = current_v + step`. La boucle continue tant que `next_v < stop` si *step*` est positif, sinon tant que `next_v > stop`. Une liste vide est produite si `start` échoue le test : `start >= stop` si `step` est positif. Le paramètre `limit` définit la longueur maximale de la liste et a une valeur par défaut de 1000. Les paramètres `start`, `step`, et `limit` sont facultatifs. Appeler `range()` avec un argument spécifie `stop`. Deux arguments spécifient « start « et « stop « . Trois arguments spécifient « start « , « stop « et « step « . Quatre arguments spécifient « start « , « stop « , « step « et « limit « . Exemples

```
range(5) -> '0, 1, 2, 3, 4'
range(0, 5) -> '0, 1, 2, 3, 4'
range(-1, 5) -> '-1, 0, 1, 2, 3, 4'
range(1, 5) -> '1, 2, 3, 4'
range(1, 5, 2) -> '1, 3'
range(1, 5, 2, 5) -> '1, 3'
range(1, 5, 2, 1) -> error(limit exceeded)
```

- `raw_field(lookup_name [, optional_default])` – renvoie le champ de métadonnées nommé par `lookup_name` sans appliquer de formatage. Il évalue et renvoie le second argument facultatif `optional_default` si la valeur du champ est indéfinie (`None`).
- `raw_list(lookup_name, separator)` – renvoie la liste de métadonnées nommée par `lookup_name` sans appliquer de formatage ou de tri, avec les éléments séparés par le séparateur.
- `re_group(value, pattern [, template_for_group]*)` – renvoie une chaîne faite en appliquant le motif de l'expression régulière à `value` et en remplaçant chaque instance trouvée par la valeur renvoyée par le

modèle correspondant. En *Mode Programme Modèle* (page 190), comme pour les fonctions `template` et `eval`, vous utilisez *[[* pour `{` et `]]` pour `}`.

L'exemple suivant recherche une série comportant plus d'un mot et met en majuscule le premier mot

```
program: re_group(field('series'), "(\S* )(.*)", "{$:uppercase()}", "{$}")'}
```

- `round(x)` – renvoie l'entier le plus proche de `x`. Envoie une exception si `x` n'est pas un nombre.
- `series_sort()` – renvoie la valeur de tri des séries.
- `strcat(a [, b]*)` – peut avoir n'importe quel nombre d'arguments. Renvoie une chaîne constituée par la concaténation de tous les arguments.
- `strcat_max(max, string1 [, prefix2, string2]*)` – Renvoie une chaîne formée par la concaténation des arguments. La valeur retournée est initialisée à `string1`. Les chaînes formées à partir des paires `prefix, string` sont ajoutées à la fin de la valeur tant que la longueur de la chaîne résultante est inférieure à `max`. Les préfixes peuvent être vides. Retourne `string1` même si `string1` est plus long que `max`. Vous pouvez passer autant de paires `prefix, string` que vous le souhaitez.
- `strcmp(x, y, lt, eq, gt)` – effectue une comparaison lexicale insensible à la casse de `x` et `y`. Elle retourne `lt` si `x < y`, `eq` si `x == y`, sinon `gt`. Cette fonction peut souvent être remplacée par l'un des opérateurs de comparaison lexicale (==, >, <, etc.).
- `strcmpcase(x, y, lt, eq, gt)` – effectue une comparaison sensible à la casse de `x` et `y` en tant que chaînes de caractères. Retourne `lt` si `x < y`. Retourne `eq` si `x == y`. Sinon, retourne `gt`.

  Note : Ce n'est PAS le comportement par défaut utilisé par calibre, par exemple, dans les opérateurs de comparaison lexicale (==, >, <, etc.). Cette fonction pourrait provoquer des résultats inattendus, utilisez de préférence `strcmp()` lorsque cela est possible.
- `strlen( value)` – Retourne la longueur de la chaîne `value`.
- `substr(str, start, end)` – renvoie les caractères de début jusqu'à fin de `str`. Le premier caractère dans `str` est le caractère zéro. Si `end` est négatif, cela indique que beaucoup de caractères sont comptés à partir de la droite. Si end est zéro, cela indique le dernier caractère. Par exemple, `substr('12345', 1, 0)` renvoie `'2345'`, et `substr('12345', 1, -1)` renvoie `'234'`.
- `substract(x, y)` – renvoie `x / y`. Lance une exception si `x` ou `y` ne sont pas des nombres. Cette fonction peut généralement être remplacée par l'opérateur −.
- `switch_if([test_expression, value_expression,]+ else_expression)` – pour chaque paire `test_expression, value_expression`, vérifie si `test_expression` est vraie (non vide) et si c'est le cas retourne le résultat de `value_expression`. Si aucune `expression_test` n'est vraie, alors le résultat de `expression_sel` est retourné. Vous pouvez avoir autant de paires `expression_test, expression_valeur` que vous le souhaitez.
- `today()` – retourne une chaîne date+heure pour aujourd'hui (now). Cette valeur est conçue pour être utilisée dans *format_date* ou *days_between*, mais peut être manipulée comme toute autre chaîne. La date est au format `ISO `_ date/heure.
- `template(x)` – évalue `x` comme un modèle. L'évaluation est faite dans son propre contexte, ce qui signifie que les variables ne sont pas partagées entre l'appelant et l'évaluation du modèle.
- `to_hex(val)` – renvoie la chaîne `val` encodée en hexadécimal. Ceci est utile lors de la construction d'URLs calibre.
- `urls_from_identifiers(identifiers, sort_results)` – donne une liste d'identifiants séparée par des virgules, où un identifiant est une paire de valeurs séparées par des deux points (`id_name:id_value`), renvoie une liste d'URL HTML séparées par des virgules, générées à partir des identifiants. La liste n'est pas triée si sort_results vaut `0` (caractère ou nombre), sinon elle est triée par ordre alphabétique du nom de l'identifiant. Les

URL sont générées de la même manière que la colonne d'identifiants intégrée lorsqu'elle est affichée dans *Détails du livre*.

10.3.7 Programmes plus complexes dans les expressions modèles - Mode Programme de Modèles

Le *Mode Programme de Modèle* (*MPM*) est un mélange du *Mode de Programme Général* (page 176) et du *Mode Fonction Unique* (page 173). Le *MPM* diffère du Mode Fonction Unique en ce qu'il permet d'écrire des expressions de modèle qui font référence à d'autres champs de métadonnées, utilisent des fonctions imbriquées, modifient des variables et font de l'arithmétique. Il diffère du *Mode Programme Général* en ce que le modèle est contenu entre les caractères *{`* et `}` et ne commence pas par le mot `program:`. La partie programme du modèle est une liste d'expressions du Mode Programme Général.

Exemple : supposons que vous vouliez qu'un modèle affiche la série d'un livre s'il en a une, sinon la valeur d'un champ personnalisé #genre. Vous ne pouvez pas faire cela dans le mode *Mode Fonction Unique* (page 173) car vous ne pouvez pas faire référence à un autre champ de métadonnées dans une expression de modèle. En *MPM*, vous le pouvez, comme le montre l'expression suivante

```
{#series:'ifempty($, field('#genre'))'}
```

L'exemple montre plusieurs choses :

- *MPP* est utilisé si l'expression commence par `:'` et se termine par `'}`. Tout le reste est supposé être en *Mode Fonction Unique* (page 173).
- la variable `$` représente le champ nommé dans le modèle : l'expression sur laquelle on opère, `#series` dans ce cas.
- les fonctions doivent être données avec tous leurs arguments. Il n'y a pas de valeur par défaut. Par exemple, les fonctions intégrées standard doivent recevoir un paramètre initial supplémentaire indiquant le champ source.
- les espaces sont ignorés et peuvent être utlisés n'importe où dans l'expression.
- les chaînes constantes sont entre guillemets, soit `'` ou `"`.

Toutes les fonctions listées dans les rubriques *Mode Fonction Unique* et *Mode Programme Général* peuvent être utilisées dans le *MPM*.

En *MPM*, l'utilisation des caractères *{`* et `}` dans les chaînes de caractères peut conduire à des erreurs ou des résultats inattendus parce qu'ils confondent le processeur de modèle. Il essaie de les traiter comme des limites d'expression de modèle et non comme des caractères. Dans certains cas, mais pas dans tous, vous pouvez remplacer un `{` par `[[` et un `}` par *]]*. En général, si votre programme contient des caractères `{` et `}`, vous devriez utiliser le *Mode Programme Général*.

Comme pour le *Mode Général de Programme*, pour les fonctions documentées sous Mode Fonction Unique, vous devez fournir la valeur sur laquelle la fonction doit agir comme premier paramètre en plus des paramètres documentés. En *MPM*, vous pouvez utiliser `$` pour accéder à la valeur spécifiée par le `lookup name` de l'expression modèle.

10.3.8 Mode Programmation Python

Mode Programmation Python (MPP) vous permet d'écrire des modèles en utilisant le langage python natif et l'API de calibre[87]. L'API de la base de données sera la plus utile ; une discussion plus approfondie dépasse le cadre de ce manuel. Les modèles MPP sont plus rapides et peuvent faire des opérations plus compliquées, mais vous devez savoir comment écrire du code en python en utilisant l'API calibre.

Un modèle de MPP commence par :

[87] https://manual.calibre-ebook.com/develop.html#api-documentation-for-various-parts-of-calibre

```
python:
def evaluate(book, context):
    # book is a calibre metadata object
    # context is an instance of calibre.utils.formatter.PythonTemplateContext,
    # which currently contains the following attributes:
    # db: a calibre legacy database object.
    # globals: the template global variable dictionary.
    # arguments: is a list of arguments if the template is called by a GPM template,␣
↪otherwise None.
    # funcs: used to call Built-in/User functions and Stored GPM/Python templates.
    # Example: context.funcs.list_re_group()

    # your Python code goes here
    return 'a string'
```

Vous pouvez ajouter le texte ci-dessus à votre modèle en utilisant le menu contextuel, généralement accessible par un clic droit. Les commentaires ne sont pas significatifs et peuvent être supprimés. Vous devez utiliser l'indentation python.

L'objet context prend en charge `str(context)` qui renvoie une chaîne de caractères du contenu du contexte, et `context.attributes` qui renvoie une liste des noms d'attributs dans le contexte.

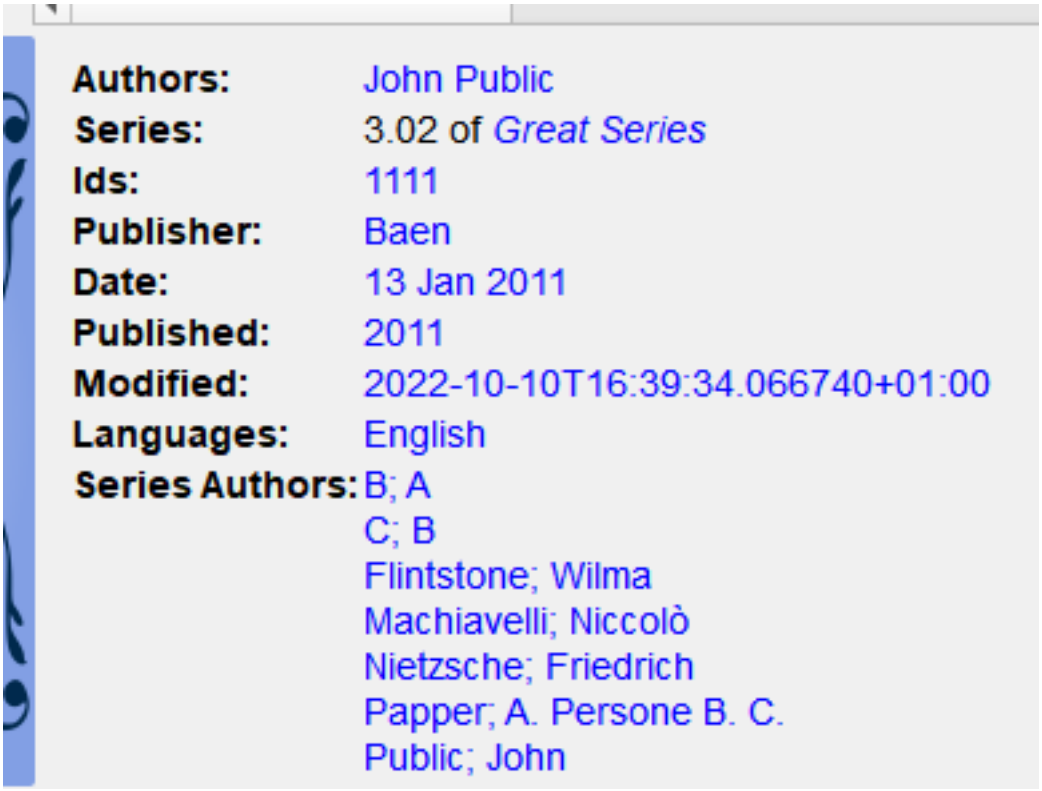
L'attribut `context.funcs` permet d'appeler les fonctions modèles intégrés et utilisateurs, ainsi que les modèles stockés MPG/Python, afin que vous puissiez les exécuter directement dans votre code. Les fonctions sont récupérées à l'aide de leur nom. Si le nom entre en conflit avec un mot clé Python, ajoutez un trait de underscore à la fin du nom. Exemples :

```
context.funcs.list_re_group()
context.funcs.assert_()
```

Voici un exemple de modèle MPP qui produit une liste de tous les auteurs d'une série. La liste est stockée dans une *Colonne construite à partir d'autres colonnes, se comporte comme des étiquettes*. Elle est affichée dans *Détails du livre* et l'option *sur des lignes séparées* est cochée (dans *Préférences → Apparence et présentation → Détails du livre*). Cette option exige que la liste soit séparée par des virgules. Pour satisfaire à cette exigence, le modèle convertit les virgules dans les noms d'auteurs en points-virgules puis construit une liste d'auteurs séparés par des virgules. Les auteurs sont ensuite triés, c'est pourquoi le modèle utilise author_sort.

```
python:
def evaluate(book, context):
    if book.series is None:
        return ''
    db = context.db.new_api
    ans = set()
    # Get the list of books in the series
    ids = db.search(f'series:"={book.series}"', '')
    if ids:
        # Get all the author_sort values for the books in the series
        author_sorts = (v for v in db.all_field_for('author_sort', ids).values())
        # Add the names to the result set, removing duplicates
        for aus in author_sorts:
            ans.update(v.strip() for v in aus.split('&'))
    # Make a sorted comma-separated string from the result set
    return ', '.join(v.replace(',', ';') for v in sorted(ans))
```

La sortie dans *Détails du livre* ressemble à ceci :



10.3.9 Modèles stockés

Les modes Mode Général de Programmation et Mode Programmation Python permettent d'enregistrer des modèles et d'appeler ces modèles à partir d'un autre modèle, de la même manière que l'on appelle des fonctions stockées. Vous enregistrez les modèles en utilisant *Préférences → Avancées → Modèles de fonctions*. Plus d'informations sont fournies dans cette boîte de dialogue. Vous appelez un modèle de la même manière que vous appelez une fonction, en passant des arguments positionnels si vous le souhaitez. Un argument peut être n'importe quelle expression. Exemples d'appel d'un modèle, en supposant que le modèle stocké est nommé `foo`

- `foo()` – appelle le modèle en ne passant aucun argument.
- `foo(a, b)` appelle le modèle en passant les valeurs des deux variables `a` et `b`.
- `foo(if field('series') then field('series_index') else 0 fi)` – si le livre a `series` alors passez le `series_index`, sinon passez la valeur `0`.

En MGP, vous récupérez les arguments passés dans l'appel au modèle stocké en utilisant la fonction `arguments`. Cette fonction déclare et initialise des variables locales, en fait des paramètres. Les variables sont positionnelles ; elles obtiennent la valeur du paramètre donné dans l'appel à la même position. Si le paramètre correspondant n'est pas fourni dans l'appel, alors `arguments` attribue à cette variable la valeur par défaut fournie. S'il n'y a pas de valeur par défaut, la variable prend la valeur de la chaîne vide. Par exemple, la fonction `arguments` suivante déclare 2 variables, `key`, `alternate` :

```
arguments(key, alternate='series')
```

Par exemple, en supposant encore une fois que le modèle stocké est nommé `foo` :

- `foo('#myseries')` – l'argument `key` se voit attribuer la valeur `myseries'` et l'argument `alternate` se voit attribuer la valeur par défaut `'series'`.
- `foo('series', '#genre')` la variable `key` prend la valeur `'series'` et la variable `alternate` prend la valeur `'#genre'`.

- `foo()` – la variable `key` se voit attribuer la chaîne vide et la variable `alternate` se voit attribuer la valeur `'series'`.

En MPP, les arguments sont passés dans le paramètre `arguments`, qui est une liste de chaînes de caractères. Il n'y a aucun moyen de spécifier des valeurs par défaut. Vous devez vérifier la longueur de la liste `arguments` pour être sûr que le nombre d'arguments est celui que vous attendez.

Une façon simple de tester les modèles stockés est d'utiliser le dialogue « Testeur de modèle ». Pour faciliter l'accès, donnez-lui un raccourci clavier dans *Préférences → Avancé → Raccourcis clavier → Tester le modèle`*. Donner un raccourci au dialogue `Modèles stockés` aidera à passer plus rapidement du testeur à l'édition du code source du modèle stocké.

10.3.10 Fournir des informations supplémentaires aux modèles

Un développeur peut choisir de transmettre des informations supplémentaires au processeur du modèle, telles que des métadonnées de livre spécifiques à l'application ou des informations sur ce qui est demandé au processeur. Un modèle peut accéder à ces informations et les utiliser pendant l'évaluation.

Développeur : comment transmettre des informations supplémentaires

L'information supplémentaire est un dictionnaire Python contenant des paires `nom_variable : valeur_variable` où les valeurs doivent être des chaînes. Le modèle peut accéder au dictionnaire, en créant des variables locales nommées `nom_de_variable` contenant la valeur `variable_valeur`. L'utilisateur ne peut pas changer le nom, il est donc préférable d'utiliser des noms qui n'entreront pas en collision avec d'autres variables locales du modèle, par exemple en préfixant le nom par un trait de soulignement.

Ce dictionnaire est passé au processeur de modèle (le `formatter`) en utilisant le paramètre nommé `global_vars=votre_dict`. La signature complète de la méthode est :

```
def safe_format(self, fmt, kwargs, error_value, book,
                column_name=None, template_cache=None,
                strip_results=True, template_functions=None,
                global_vars={})
```

Rédacteur de modèles : comment accéder aux informations complémentaires

Vous accédez aux informations supplémentaires (le dictionnaire `globals`) dans un modèle en utilisant la fonction de modèle :

```
globals(id[=expression] [, id[=expression]]*)
```

où `id` est un nom de variable légale quelconque. Cette fonction vérifie si les informations supplémentaires fournies par le développeur contiennent le nom. Si c'est le cas, la fonction attribue la valeur fournie à une variable locale modèle avec ce nom. Si le nom ne figure pas dans les informations supplémentaires et si une « expression » est fournie, cette « expression » est évaluée et le résultat est attribué à la variable locale. Si ni une valeur ni une expression n'est fournie, la fonction attribue la chaîne vide (`''`) à la variable locale.

Un modèle peut définir une valeur dans le dictionnaire `globals` en utilisant la fonction de modèle :

```
set_globals(id[=expression] [, id[=expression]]*)
```

Cette fonction définit la paire clé :valeur du dictionnaire `globals` `id:value` où `value` est la valeur de la variable locale du modèle `id`. Si cette variable locale n'existe pas, alors `valeur` est fixée au résultat de l'évaluation de `expression`.

10.3.11 Notes sur la différence entre les modes

Les trois modes de programmation, Mode Fonction Simple (MFS), Mode Modèle de Programmation (*MMP*), et Mode Général de Programmation (*MGP*), fonctionnent différemment. Le MFS est destiné à être "simple" et cache donc beaucoup de bits de langage de programmation.

Différences :

- Dans MFS, la valeur de la colonne est toujours passée comme premier argument "invisible" à une fonction incluse dans le modèle.
- Le MFS ne supporte pas la différence entre les variables et les chaînes de caractères ; toutes les valeurs sont des chaînes de caractères.
- Le modèle SFM suivant renvoie soit le nom de la série, soit la chaîne « pas de série »

```
{series:ifempty(no series)}
```

  Le modèle équivalent dans *MPM* est

```
{series:'ifempty($, 'no series')'}
```

  Le modèle équivalent dans *MGP* est

```
program: ifempty(field('series'), 'no series')
```

  Le premier argument de `ifempty` est la valeur du champ `série`. Le second argument est la chaîne `no series`. Dans SFM, le premier argument, la valeur du champ, est automatiquement passé (l'argument invisible).
- Plusieurs fonctions du modèle, par exemple `booksize()` et `current_library_name()`, ne prennent aucun argument. En raison de l'« argument invisible », vous ne pouvez pas utiliser ces fonctions dans SFM.
- Les fonctions imbriquées, où une fonction appelle une autre fonction pour calculer un argument, ne peuvent pas être utilisées dans SFM. Par exemple, ce modèle, destiné à renvoyer les 5 premiers caractères de la valeur de la série en majuscules, ne fonctionnera pas dans SFM

```
{series:uppercase(substr(0,5))}
```

- *MPM* et *MPG* soutiennent des fonctions imbriquées. Le modèle ci-dessus dans *MPM* serait

```
{series:'uppercase(substr($, 0,5))'}
```

  Dans *MPG*, ce serait

```
program: uppercase(substr(field('series'), 0,5))
```

- Comme indiqué dans la section Mode de Programmation des Modèles ci-dessus, l'utilisation des caractères `{` et `}` dans les chaînes de caractères *MPM* peut conduire à des erreurs ou à des résultats inattendus car ils perturbent le processeur de modèles. Il essaie de les traiter comme des limites de modèles et non comme des caractères. Dans certains cas, mais pas dans tous, vous pouvez remplacer un `{` par `[[` et un `}` par `]]`. En général, si votre programme contient des caractères ``{` et `}`, vous devriez utiliser le *Mode Général de Programmation*.

10.3.12 Fonctions des modèles Python définis par l'utilisateur

Vous pouvez ajouter vos propres fonctions Python au processeur de modèles. Ces fonctions peuvent être utilisées dans l'un des trois modes de programmation du modèle. Les fonctions sont ajoutées en allant dans *Préférences* → *Avancé* → *Fonctions de modèle*. Les instructions sont affichées dans cette boite de dialogue.

10.3.13 Notes spéciales pour la sauvergarde/l'envoi des modèles

Un traitement spécial est appliqué quand un modèle est utilisé dans un modèle *Enregistrer sur le disque* ou *Envoyer au périphérique*. Les valeurs des champs sont nettoyées, les caractères qui sont propres aux systèmes de fichiers sont remplacées par des traits de soulignement, y compris les barres obliques. Cela signifie que le texte d'un champ ne peut pas être utilisé pour créer des dossiers. Toutefois, les barres obliques ne sont pas modifiés dans des chaînes préfixe ou suffixe. Vous pouvez donc insérer des barres obliques qui et provoquer ainsi la création de dossiers. Ainsi, vous pouvez créer une structure de dossier avec des profondeurs variables.

Par exemple, supposons que nous voulons que la structure des dossiers soit *series/series_index - titre*, en prévoyant que si la série n'existe pas, alors le titre devrait être dans le dossier racine (le premier dossier, pas le sous-dossier). Le modèle pour faire cela est :

```
{series:||/}{series_index:|| - }{title}
```

La barre oblique et le trait d'union apparaissent uniquement si la série n'est pas vide.

La fonction de recherche nous permet de faire le traitement encore plus poussé. Par exemple, supposons que si un livre est dans une série, alors nous voulons que la structure soit la suivante : *series/series index - title.fmt*.. Si le livre n'est pas dans une série, alors nous voulons que la structure des dossiers soit *genre/author_sort/title.fmt*. Si le livre n'a pas de genre, alors nous voulons utiliser "Inconnu". Nous voulons deux chemins de répertoire totalement différents en fonction de la valeur de la série.

Pour accomplir cela, nous :

1. Créer un champ composite (donner-lui le nom de recherche #aa) contenant `{series}/{series_index} - {titre}`. Si la série n'est pas vide, alors ce modèle produira *series/series_index - titre*.
2. Créons une champ composite (appelons-le #bb) contenant `{#genre:ifempty(Unknown)}/{author_sort}/{title}`. Ce modèle produit *genre/author_sort/title*, où un genre vide est remplacé par *Inconnu*.
3. Paramétrez le modèle de sauvegarde à `{series:lookup(.,#aa,#bb)}`. Ce modèle choisit le champ composite `#aa` si série n'est pas vide, et le champ composite `#bb` si série est vide. Nous avons dès lors deux chemins de sauvegarde complètement différents, dépendant du fait que *series* soit vide ou pas.

10.3.14 Astuces

- Utilisez le Testeur de Modèles pour tester les modèles. Ajoutez le testeur au menu contextuel des livres de la bibliothèque et/ou donnez-lui un raccourci clavier.
- Les modèles peuvent utiliser d'autres modèles en faisant référence à des colonnes composites construites avec le modèle souhaité. Vous pouvez également utiliser des Modèles Stockés comme alternative.
- Dans un tableau de connexions, vous pouvez paramétrer un champ à vide (ou quelque soit sont équivalent à vide) en utilisant le modèle spécial `{}`. Ce modèle sera toujours évalué à une chaîne vide.
- La technique indiquée ci-dessus pour montrer les nombre même s'ils ont une valeur zéro fonctionne avec le champ standard series_index.

10.3.15 Référence de la fonction

Reference for all built-in template language functions

Here, we document all the built-in functions available in the calibre template language. Every function is implemented as a class in python and you can click the source links to see the source code, in case the documentation is insufficient. The functions are arranged in logical groups by type.

- *Arithmetic* (page 200)
  - *add(x [, y]*)* (page 200)
  - *ceiling(x)* (page 200)
  - *divide(x, y)* (page 200)
  - *floor(x)* (page 200)
  - *fractional_part(x)* (page 200)
  - *mod(x)* (page 200)
  - *multiply(x [, y]*)* (page 200)
  - *round(x)* (page 201)
  - *subtract(x, y)* (page 201)
- *Boolean* (page 201)
  - *and(valeur [, valeur]*)* (page 201)
  - *not(value)* (page 201)
  - *or(valeur [, valeur]*)* (page 201)
- *Date functions* (page 201)
  - *date_arithmetic(date, calc_spec, fmt)* (page 201)
  - *days_between(date1, date2)* (page 202)
  - *today()* (page 202)
- *Formatting values* (page 202)
  - *finish_formatting(val, fmt, prefix, suffix)* (page 202)
  - *format_date(val, chaine_format)* (page 202)
  - *format_date_field(field_name, format_string)* (page 203)
  - *format_number(v, template)* (page 203)
  - *human_readable(v)* (page 203)
  - *rating_to_stars(value, use_half_stars)* (page 203)
  - *urls_from_identifiers(identifiers, sort_results)* (page 203)
- *Get values from metadata* (page 204)
  - *annotation_count()* (page 204)
  - *approximate_formats()* (page 204)
  - *author_links(val_separator, pair_separator)* (page 204)

- *author_sorts(val_separator)* (page 204)
- *booksize()* (page 205)
- *connected_device_name(storage_location)* (page 205)
- *connected_device_uuid(storage_location)* (page 205)
- *current_library_name()* (page 205)
- *current_library_path()* (page 205)
- *current_virtual_library_name()* (page 205)
- *field(lookup_name)* (page 206)
- *formats_modtimes(date_format)* (page 206)
- *formats_paths()* (page 206)
- *formats_sizes()* (page 206)
- *has_cover()* (page 206)
- *is_marked()* (page 206)
- *language_codes(lang_strings)* (page 207)
- *language_strings(lang_codes, localize)* (page 207)
- *ondevice()* (page 207)
- *raw_field(lookup_name [, optional_default]))* (page 207)
- *raw_field(lookup_name, separator)* (page 207)
- *series_sort()* (page 207)
- *user_categories()* (page 208)
- *virtual_libraries()* (page 208)

If-then-else (page 208)

- *check_yes_no(field_name, is_undefined, is_false, is_true)* (page 208)
- *contains(val, pattern, texte si correspondance, texte si pas de correspondance)* (page 208)
- *field_exists(field_name)* (page 208)
- *ifempty(val, texte si vide)* (page 209)
- *test(val, text if not empty, text if empty)* (page 209)

Iterating over values (page 209)

- *first_non_empty(value [, value]*)* (page 209)
- *lookup(val, [pattern, field,]+ else_field)* (page 209)
- *switch(val, [pattern, value,]+ else_value)* (page 209)
- *switch_if([test_expression, value_expression,]+ else_expression)* (page 209)

List lookup (page 210)

- *identifier_in_list(val, id_name [, found_val, not_found_val])* (page 210)
- *in_list(val, separator, [pattern, found_val,]+ not_found_val)* (page 210)

- *list_item(val, index, separator)* (page 210)
- *select(val, key)* (page 210)
- *str_in_list(val, separator, [string, found_val,]+ not_found_val)* (page 210)

- *List manipulation* (page 211)
  - *count(separator)* (page 211)
  - *list_count_matching(list, pattern, separator)* (page 211)
  - *list_difference(list1, list2, separator)* (page 211)
  - *list_equals(list1, sep1, list2, sep2, yes_val, no_val)* (page 211)
  - *list_intersection(list1, list2, separator)* (page 211)
  - *list_join(with_separator, list1, separator1 [, list2, separator2]*)* (page 212)
  - *list_re(src_list, separator, include_re, opt_replace)* (page 212)
  - *list_re_group(src_list, separator, include_re, search_re [, group_template]+)* (page 212)
  - *list_remove_duplicates(list, separator)* (page 213)
  - *list_sort(liste, direction, separator)* (page 213)
  - *list_split(list_val, sep, id_prefix)* (page 213)
  - *list_union(list1, list2, separator)* (page 213)
  - *range(start, stop, step, limit)* (page 213)
  - *subitems(val, start_index, end_index)* (page 214)
  - *sublist(val, start_index, end_index, séparator)* (page 214)
- *Other* (page 214)
  - *arguments(id[=expression] [, id[=expression]]*)* (page 214)
  - *assign(id, val)* (page 214)
  - *globals(id[=expression] [, id[=expression]]*)* (page 215)
  - *print(a[, b]*)* (page 215)
- *Recursion* (page 215)
  - *eval(modèle)* (page 215)
  - *template(x)* (page 215)
- *Relational* (page 215)
  - *cmp(x, y, lt, eq, gt)* (page 215)
  - *first_matching_cmp(val, cmp1, result1,]+, else_result)* (page 216)
  - *strcmp(x, y, lt, eq, gt)* (page 216)
  - *strcmpcase(x, y, lt, eq, gt)* (page 216)
- *String case changes* (page 216)
  - *capitalize(val)* (page 216)
  - *lowercase(val)* (page 216)

- *titlecase(val)* (page 216)
- *uppercase(val)* (page 217)

- *String manipulation* (page 217)
  - *character(character_name)* (page 217)
  - *re(val, pattern, replacement)* (page 217)
  - *re_group(val, pattern [, template_for_group]*)* (page 217)
  - *shorten(val, left chars, middle text, right chars)* (page 217)
  - *strcat(a [, b]*)* (page 218)
  - *strcat_max(max, string1 [, prefix2, string2]*)* (page 218)
  - *strlen(a)* (page 218)
  - *substr(ch, start, end)* (page 218)
  - *swap_around_articles(separator)* (page 218)
  - *swap_around_comma(val)* (page 218)
  - *to_hex(val)* (page 219)
  - *transliterate(a)* (page 219)
- *Template database functions* (page 219)
  - *book_count(query, use_vl)* (page 219)
  - *book_values(column, query, sep, use_vl)* (page 219)
  - *extra_file_modtime(file_name, format_string)* (page 219)
  - *extra_file_names(sep [, pattern])* (page 219)
  - *extra_file_size(file_name)* (page 220)
  - *get_link(field_name, field_value)* (page 220)
  - *get_note(field_name, field_value, plain_text)* (page 220)
  - *has_extra_files([pattern])* (page 220)
  - *has_note(field_name, field_value)* (page 220)
- *other* (page 220)
  - *set_globals(id[=expression] [, id[=expression]]*)* (page 220)
- *API of the Metadata objects* (page 221)

Arithmetic

add(x [, y]*)

class calibre.utils.formatter_functions.**BuiltinAdd**

add(x [, y]*) – retourne la somme de ses arguments. Déclenche une exception si un argument n'est pas un nombre. Cette fonction peut souvent être remplacée par l'opérateur +.

ceiling(x)

class calibre.utils.formatter_functions.**BuiltinCeiling**

ceiling(x) – renvoie le plus petit entier supérieur ou égal à x. Envoie une exception si x n'est pas un nombre.

divide(x, y)

class calibre.utils.formatter_functions.**BuiltinDivide**

divide(x, y) – renvoie x / y. Déclenche une exception si x ou y ne sont pas des nombres. Cette fonction peut souvent être remplacée par l'opérateur /.

floor(x)

class calibre.utils.formatter_functions.**BuiltinFloor**

floor(x) – renvoie le plus grand entier inférieur ou égal à x. Envoie une exception si x n'est pas un nombre.

fractional_part(x)

class calibre.utils.formatter_functions.**BuiltinFractionalPart**

fractional_part(x) – renvoie la valeur après la virgule. Par exemple, fractional_part(3.14) renvoie 0.14. Envoie une exception si x n'est pas un nombre.

mod(x)

class calibre.utils.formatter_functions.**BuiltinMod**

mod(x) – renvoie le solde (reste de x / y). Déclenche une exception si x ou y n'est pas un nombre.

multiply(x [, y]*)

class calibre.utils.formatter_functions.**BuiltinMultiply**

multiply(x [, y]*) – renvoie le produit de ses arguments. Déclenche une exception si l'un des arguments n'est pas un nombre. Cette fonction peut souvent être remplacée par l'opérateur *.

round(x)

class calibre.utils.formatter_functions.**BuiltinRound**

round(x) – renvoie l'entier le plus proche de x. Envoie une exception si x n'est pas un nombre.

subtract(x, y)

class calibre.utils.formatter_functions.**BuiltinSubtract**

subtract(x, y) – renvoie x - y. Déclenche une exception si x ou y ne sont pas des nombres. Cette fonction peut souvent être remplacée par l'opérateur -.

Boolean

and(valeur [, valeur]*)

class calibre.utils.formatter_functions.**BuiltinAnd**

and(valeur [, valeur]*) – renvoie la chaîne « 1 » si toutes les valeurs ne sont pas vides, sinon renvoie la chaîne vide. Cette fonction fonctionne bien avec test ou first_non_empty. Vous pouvez avoir autant de valeurs que vous le souhaitez. Dans de nombreux cas, l'opérateur && peut remplacer cette fonction.

not(value)

class calibre.utils.formatter_functions.**BuiltinNot**

not(value) – retourne la chaîne « 1 » si la valeur est vide, sinon retourne la chaîne vide. Cette fonction fonctionne bien avec test ou first_non_empty. Dans de nombreux cas, l'opérateur ! peut remplacer cette fonction.

or(valeur [, valeur]*)

class calibre.utils.formatter_functions.**BuiltinOr**

or(valeur [, valeur]*) – retourne la chaîne « 1 » si une valeur n'est pas vide, sinon retourne la chaîne vide. Cette fonction fonctionne bien avec test ou first_non_empty. Vous pouvez avoir autant de valeurs que vous le souhaitez. Dans de nombreux cas, l'opérateur || peut remplacer cette fonction.

Date functions

date_arithmetic(date, calc_spec, fmt)

class calibre.utils.formatter_functions.**BuiltinDateArithmetic**

date_arithmetic(date, calc_spec, fmt) – Calcule une nouvelle date à partir de "date" en utilisant "calc_spec". Retourne la nouvelle date formatée selon l'option "fmt" : si elle n'est pas fournie, le résultat sera au format iso. calc_spec est une chaîne de caractères formée par la concaténation de paires de "vW" (valueWhat) où "v" est un nombre éventuellement négatif et W est l'une des lettres suivantes : s : ajoute "v" secondes à "date" m : ajoute "v" minutes à "date" h : ajoute "v" heures à "date" d : ajoute "v" jours à "date" w : ajoute "v" semaines à "date" y : ajoute "v" années à "date", où une année est de 365 jours. Exemple : "1s3d-1m" ajoutera 1 seconde, ajoutera 3 jours et soustraira 1 minute à la "date".

days_between(date1, date2)

class calibre.utils.formatter_functions.**BuiltinDaysBetween**

days_between(date1, date2) – renvoie le nombre de jours entre date1 et date2. Le nombre est positif si date1 est plus grand que date2, sinon négatif. Si date1 ou date2 ne sont pas des dates, la fonction renvoie une chaîne vide.

today()

class calibre.utils.formatter_functions.**BuiltinToday**

today() – renvoie une chaîne date pour aujourd'hui. Cette valeur est conçue pour être utilisée avec format_date ou days_between, mais peut être manipulée comme toute autre chaîne. La date est au format ISO.

Formatting values

finish_formatting(val, fmt, prefix, suffix)

class calibre.utils.formatter_functions.**BuiltinFinishFormatting**

finish_formatting(val, fmt, prefix, suffix) – applique le format, le préfixe et le suffixe à une valeur de la même manière que cela est effectué dans un modèle comme *{series_index :05.2f\ - \|- }*. Par exemple, le programme suivant produit le même résultat que le modèle ci-dessus : programme : finish_formatting(field(« series_index »), « 05.2f », « - « , « - «)

format_date(val, chaine_format)

class calibre.utils.formatter_functions.**BuiltinFormatDate**

format_date(val, chaine_format) – formate la valeur, qui doit être une date, en utilisant la chaine_format, retournant une chaine. Les codes de formatage sont : d : le jour comme nombre sans zéro (1 à 31) dd :le jour comme nombre avec zéro (01 à 31) ddd : le nom du jour localisé et abrégé (p. ex. « Lun » to « Dim »). dddd : le nom du jour localisé en entier (p. ex. « Lundi » to « Dimanche »). M : le mois comme nombre sans zéro (1 à 12). MM : le mois comme nombre avec zéro (01 à 12) MMM : le nom du mois localisé et abrégé (p. ex. « Jan » to « Dec »). MMMM : le nom du mois localisé en entier (p. ex. « Janvier » to « Décembre »). yy : l'année comme nombre à 2 chiffres (00 à 99). yyyy : l'année comme nombre à 4 chiffres. h : les heures sans zéro (0 à 11 où 0 à 23, en fonction de am/pm) hh : les heures avec zéro (00 à 11 où 00 à 23, en fonction de am/pm) m : les minutes sans zéro (0 à 59) mm : les minutes avec zéro(00 à 59) s : les secondes sans zéro (0 à 59) ss : les secondes avec zéro (00 à 59) ap : utiliser une horloge à 12-heures au lieu de 24-heures, avec « ap » remplacé par la chaine localisé pour am ou pm AP : utiliser une horloge à 12-heures au lieu de 24-heures, avec « AP » remplacé par la chaine localisé pour AM ou PM iso : la date avec le temps et le fuseau horaire. Doit être le seul format présent to_number : la date sous forme de nombre à virgule flottante from_number[:fmt] : formate l'horodatage en utilisant fmt si présent sinon iso

format_date_field(field_name, format_string)

class calibre.utils.formatter_functions.**BuiltinFormatDateField**

format_date_field(field_name, format_string) – formate la valeur du champ "field_name", qui doit être le nom d'un champ date, standard ou personnalisé. Voir "format_date()" pour les codes de formatage. Cette fonction est beaucoup plus rapide que format_date et doit être utilisée lorsque vous formatez la valeur d'un champ (colonne). Elle ne peut pas être utilisée pour des dates calculées ou des dates dans des variables de type chaîne. Exemples : format_date_field("pubdate", "yyyy.MM.dd")

format_number(v, template)

class calibre.utils.formatter_functions.**BuiltinFormatNumber**

format_number(v, template) – formate le nombre v en utilisant un modèle (template) de formatage Python comme « {0 :5.2f} » ou « {0 :,d} » ou « ${0 :5,.2f} ». Le nom du champs (field_name) dans le modèle doit être un 0 (zéro) (le « {0 : » dans les exemples ci-dessus). Regardez le langage des modèles et la documentation Python pour plus d'exemples.Vous pouvez omettre le « {0 : » au début et le « } » à la fin si le modèle contient uniquement un format. Retourne une chaîne vide si le formatage échoue.

human_readable(v)

class calibre.utils.formatter_functions.**BuiltinHumanReadable**

human_readable(v) – renvoie une chaine représentant le nombre v en Ko, Mo, Go, etc.

rating_to_stars(value, use_half_stars)

class calibre.utils.formatter_functions.**BuiltinRatingToStars**

`rating_to_stars(value, use_half_stars)` – Retourne la note sous forme de chaîne de caractères étoilés. La valeur est un nombre entre 0 et 5. Mettre use_half_stars à 1 si vous voulez des caractères demi-étoiles pour les colonnes de classement personnalisées qui ne sont pas des entiers, par exemple 2.5.

urls_from_identifiers(identifiers, sort_results)

class calibre.utils.formatter_functions.**BuiltinUrlsFromIdentifiers**

urls_from_identifiers(identifiers, sort_results) – donne une liste d'identifiants séparée par des virgules, où un identifiant est une paire de valeurs séparées par des deux points (name :id_value), renvoie une liste d'URL HTML séparées par des virgules, générées à partir des identifiants. La liste n'est pas triée si sort_results vaut 0 (caractère ou nombre), sinon elle est triée par ordre alphabétique du nom de l'identifiant. Les URL sont générées de la même manière que la colonne d'identifiants intégrée lorsqu'elle est affichée dans Détails du livre.

Get values from metadata

annotation_count()

class calibre.utils.formatter_functions.**BuiltinAnnotationCount**

annotation_count() – renvoie le nombre total d'annotations de tous types attachées au livre en cours. Cette fonction ne fonctionne que dans l'interface graphique.

approximate_formats()

class calibre.utils.formatter_functions.**BuiltinApproximateFormats**

approximate_formats() – retourne une liste de formats, séparées par des virgules, qui, à un certain moment, on été associés avec le livre. Il n'y a aucune garantie que cette liste est correcte, bien qu'elle le soit probablement. Cette fonction peut-être appelée sous forme de modèle de programme en utilisant le modèle « {:"approximate_formats()"} ». Notez que ces noms de formats sont toujours en majuscule, comme dans EPUB. Cette fonction n'est utilisable que dans l'interface utilisateur. Si vous désirez utiliser cette valeur dans « Enregistrer sous… » ou « Envoyer vers le périphérique connecté », vous devez créer une colonne personnalisée de type « Colonne construite à partir d'autres colonnes » et utiliser cette fonction dans le modèle de la colonne. Vous pourrez ensuite utiliser la valeur de la colonne dans vos modèles d'enregistrement / d'envoi.

author_links(val_separator, pair_separator)

class calibre.utils.formatter_functions.**BuiltinAuthorLinks**

author_links(val_separator, pair_separator) – retourne une chaîne contenant une liste d'auteurs et les valeurs de ce lien d'auteur dans la forme auteur1 val_separator author1link pair_separator auteur2 val_separator author2link etc. Un auteur est séparé de la valeur de son lien par la chaîne val_separator sans espaces ajoutés. Les paires author :linkvalue sont séparées par la chaîne d'argument pair_separator sans espaces ajoutés. C'est à vous de choisir des chaines de Il vous incombe de choisir des chaines de séparateur qui n'apparaissent pas dans des noms ou des liens d'auteur. Un auteur est inclus même si le lien d'auteur est vide.

author_sorts(val_separator)

class calibre.utils.formatter_functions.**BuiltinAuthorSorts**

`author_sorts(val_separator)` – renvoie une chaîne contenant une liste des valeurs de tri de l'auteur pour les auteurs du livre. Le tri est celui des métadonnées auteur (différente de l'author_sort dans les livres). La liste renvoyée a la forme tri auteur 1 `val_separator` tri auteur 2 etc. Les valeurs de tri auteur dans cette liste sont dans le même ordre que les auteurs du livre. Si vous voulez les espaces autour du `val_separator` incluez les alors dans la chaîne separator

booksize()

class calibre.utils.formatter_functions.**BuiltinBooksize**

booksize() – retourne la valeur de la taille du champ. Cette fonction n'est utilisable que dans l'interface utilisateur. Si vous désirez utiliser cette valeur dans « Enregistrer sous… » ou « Envoyer vers le périphérique connecté », vous devez créer une colonne personnalisée de type « Colonne construite à partir d'autres colonnes » et utiliser cette fonction dans le modèle de la colonne. Vous pourrez ensuite utiliser la valeur de la colonne dans vos modèles d'enregistrement / d'envoi.

connected_device_name(storage_location)

class calibre.utils.formatter_functions.**BuiltinConnectedDeviceName**

connected_device_name(storage_location) – si un périphérique est connecté alors retournez le nom du périphérique, sinon retournez la chaîne vide. Chaque emplacement de stockage sur un périphérique peut avoir un nom différent. Les noms des emplacements sont « main », « carda » et « cardb ». Cette fonction ne fonctionne que dans l'interface graphique.

connected_device_uuid(storage_location)

class calibre.utils.formatter_functions.**BuiltinConnectedDeviceUUID**

connected_device_uuid(storage_location) – si un périphérique est connecté alors retournez l'uuid du périphérique (id unique), sinon retournez la chaîne vide. Chaque emplacement de stockage sur un périphérique a un uuid différent. Les noms des emplacements sont "main", "carda" et "cardb". Cette fonction ne fonctionne que dans l'interface graphique.

current_library_name()

class calibre.utils.formatter_functions.**BuiltinCurrentLibraryName**

current_library_name() – retourne le dernier nom du chemin de la bibliothèque calibre actuelle. Cette fonction peut-être appelée en mode de modèle de programme en utilisant le modèle « {:"current_library_name()"} ».

current_library_path()

class calibre.utils.formatter_functions.**BuiltinCurrentLibraryPath**

current_library_path() – retourne le chemin de la bibliothèque calibre actuelle. Cette fonction peut-être appelée en mode de modèle de programme en utilisant le modèle « {:"current_library_path()"} ».

current_virtual_library_name()

class calibre.utils.formatter_functions.**BuiltinCurrentVirtualLibraryName**

current_virtual_library_name() – retourne le nom de la bibliothèque virtuelle actuelle s'il y en a une, sinon la chaîne vide. La casse du nom de la bibliothèque est préservée. Exemple : « program :current_virtual_library_name() ».

field(lookup_name)

class calibre.utils.formatter_functions.**BuiltinField**

field(lookup_name) – retourne le champ de métadonnées nommé par lookup_name

formats_modtimes(date_format)

class calibre.utils.formatter_functions.**BuiltinFormatsModtimes**

formats_modtimes(date_format) – retourne une liste séparée par des virgules d'éléments séparés par deux-points représentant les dates de modification pour les formats d'un livre. Le paramètre date_format indique comment la date doit être formatée. Regardez la fonction date_format pour des détails. Vous pouvez utiliser la fonction select pour obtenir le mod time pour un format spécifique.Veuillez noter que les noms de format sont toujours en majuscules, comme dans EPUB.

formats_paths()

class calibre.utils.formatter_functions.**BuiltinFormatsPaths**

formats_paths() – retourne une liste séparées par des virgules d'éléments séparés par deux points représentant le chemin absolu vers les formats d'un livre. Vous pouvez utiliser cette fonction pour obtenir le chemin d'un format spécifique. Notez que ces noms de formats sont toujours en majuscule, comme dans EPUB.

formats_sizes()

class calibre.utils.formatter_functions.**BuiltinFormatsSizes**

formats_sizes() – renvoie une liste séparée par des virgules correspondant à des articles représentant les tailles en octets des formats d'un livre. Vous pouvez utiliser la fonction select pour obtenir la taille d'un format spécifique. Notez que ces noms de formats sont toujours en majuscule, comme dans EPUB.

has_cover()

class calibre.utils.formatter_functions.**BuiltinHasCover**

has_cover() – renvoie Yes si le livre a une couverture, sinon renvoie une chaine vide

is_marked()

class calibre.utils.formatter_functions.**BuiltinIsMarked**

is_marked() – vérifie si le livre est "marqué" dans calibre. S'il l'est, il renvoie alors la valeur de la marque, soit « true », soit la liste des marques nommées séparées par des virgules. Retourne "" si le livre n'est pas marqué.

language_codes(lang_strings)

class calibre.utils.formatter_functions.**BuiltinLanguageCodes**

language_codes(lang_strings) – retourne les codes de langue pour les chaines passées dans lang_strings. Les chaines doivent être dans la langue actuelle de localisation. Lang_strings est une liste séparée par des virgules.

language_strings(lang_codes, localize)

class calibre.utils.formatter_functions.**BuiltinLanguageStrings**

language_strings(lang_codes, localize) – retourne les chaines des codes de langue passés dans lang_codes. Si localisation est égale à zéro, retourne les chaines en anglais. Si localisation est différente de zéro, retourne les chaines dans la langue utilisée actuellement. Lang_codes est une liste séparée par des virgules.

ondevice()

class calibre.utils.formatter_functions.**BuiltinOndevice**

ondevice() – renvoie Yes si « Sur le périphérique » est activé, sinon renvoie une chaîne vide. Cette fonction n'est utilisable que dans l'interface utilisateur. Si vous désirez utiliser cette valeur dans « Enregistrer sous… » ou « Envoyer vers le périphérique connecté », vous devez créer une colonne personnalisée de type « Colonne construite à partir d'autres colonnes » et utiliser cette fonction dans le modèle de la colonne. Vous pourrez ensuite utiliser la valeur de la colonne dans vos modèles d'enregistrement / d'envoi.

raw_field(lookup_name [, optional_default]))

class calibre.utils.formatter_functions.**BuiltinRawField**

raw_field(lookup_name [, optional_default])) – renvoie le champ de métadonnées nommé par lookup_name sans appliquer de formatage. Il évalue et renvoie le second argument optionnel "default" si le champ est indéfini ("None").

raw_field(lookup_name, separator)

class calibre.utils.formatter_functions.**BuiltinRawList**

`raw_field(lookup_name, separator)` – renvoie la liste de métadonnées nommée par lookup_name sans appliquer aucun formatage ou triage et avec les éléments séparés par separator.

series_sort()

class calibre.utils.formatter_functions.**BuiltinSeriesSort**

series_sort() – renvoie la valeur de tri des séries

user_categories()

class calibre.utils.formatter_functions.**BuiltinUserCategories**

user_categories() – retourne une liste séparée par des virgules des catégories utilisateur contenant ce livre, séparées par des virgules . Cette fonction n'est utilisable que dans l'interface utilisateur. Si vous désirez utiliser cette valeur dans « Enregistrer sous… » ou « Envoyer vers le périphérique connecté », vous devez créer une colonne personnalisée de type « Colonne construite à partir d'autres colonnes », utiliser cette fonction dans ce modèle de colonne, et utiliser la valeur de la colonne dans vos modèles d'enregistrement / d'envoi.

virtual_libraries()

class calibre.utils.formatter_functions.**BuiltinVirtualLibraries**

virtual_libraries() – retourne une liste de bibliothèques virtuelles contenant ce livre, séparées par des virgules. Cette fonction n'est utilisable que dans l'interface utilisateur. Si vous désirez utiliser ces valeurs dans les modèles enregistrement-sur-disque ou envoi-vers-appareil, vous devez créer une colonne personnalisée de type « Colonne construite à partir d'autres colonnes », utiliser cette fonction dans le modèle de la colonne, et utiliser la valeur de la colonne dans vos modèles d'enregistrement/envoi.

If-then-else

check_yes_no(field_name, is_undefined, is_false, is_true)

class calibre.utils.formatter_functions.**BuiltinCheckYesNo**

check_yes_no(field_name, is_undefined, is_false, is_true) – vérifie la valeur du champ oui/non nommé par la clé de recherche field_name pour une valeur spécifiée par les paramètres, en retournant « yes » si une correspondance est trouvée, sinon en retournant une chaîne vide. Définissez le paramètre is_undefined, is_false, ou is_true à 1 (le nombre) pour vérifier cette condition, sinon mettez le à 0. Exemple : check_yes_no(« #bool », 1, 0, 1) renvoie « yes » si le champ oui/non « #bool » est soit indéfini (ni True ni False) ou True. Plus d'un des champs is_undefined, is_false ou is_true peut être mis à 1. Cette fonction est généralement utilisée par les fonctions test() ou is_empty().

contains(val, pattern, texte si correspondance, texte si pas de correspondance)

class calibre.utils.formatter_functions.**BuiltinContains**

contains(val, pattern, texte si correspondance, texte si pas de correspondance) – vérifie si val contient des correspondances pour l'expression régulière *pattern*. Renvoie *texte si correspondance* si des correspondances sont trouvées, sinon il renvoie *texte si pas de correspondance*

field_exists(field_name)

class calibre.utils.formatter_functions.**BuiltinFieldExists**

field_exists(field_name) – vérifie si un champ (colonne) nommé field_name existe, en retournant "1" si oui et "" si non.

ifempty(val, texte si vide)

class calibre.utils.formatter_functions.**BuiltinIfempty**

ifempty(val, texte si vide) – renvoie val si val n'est pas vide, sinon renvoie *texte si vide*

test(val, text if not empty, text if empty)

class calibre.utils.formatter_functions.**BuiltinTest**

test(val, text if not empty, text if empty) – renvoie *texte si non vide* si val n'est pas vide, sinon renvoie *texte si vide*

Iterating over values

first_non_empty(value [, value]*)

class calibre.utils.formatter_functions.**BuiltinFirstNonEmpty**

first_non_empty(value [, value]*) – retourne la première valeur non vide. Si toutes les valeurs sont vides, alors la chaîne vide est retournée. Vous pouvez avoir autant de valeurs que vous voulez.

lookup(val, [pattern, field,]+ else_field)

class calibre.utils.formatter_functions.**BuiltinLookup**

lookup(val, [pattern, field,]+ else_field) – comme switch, sauf que les arguments sont des noms de champs (métadonnées) et non du texte. La valeur du champ approprié sera récupérée et utilisée. Notez que les colonnes composites étant des champs, vous pouvez utiliser cette fonction dans un champ composite pour utiliser la valeur d'un autre champ composite. Cette fonction est extrêmement utile pour construire des chemins de sauvegarde variables.

switch(val, [pattern, value,]+ else_value)

class calibre.utils.formatter_functions.**BuiltinSwitch**

switch(val, [pattern, value,]+ else_value) – pour chaque paire *pattern, value*, vérifie si *val* correspond à l'expression régulière *pattern* et si oui, retourne cette *value*. Si aucun motif ne correspond, alors *else_value* est retourné. Vous pouvez avoir autant de paires *pattern, value* que vous le souhaitez.

switch_if([test_expression, value_expression,]+ else_expression)

class calibre.utils.formatter_functions.**BuiltinSwitchIf**

switch_if([test_expression, value_expression,]+ else_expression) – pour chaque paire « test_expression, value_expression », vérifie si test_expression est vrai (non vide) et si c'est le cas, renvoie le résultat de value_expression. Si aucune test_expression n'est Vraie, le résultat de else_expression est renvoyé. Vous pouvez avoir autant de paires « test_expression, value_expression » que vous le souhaitez.

List lookup

identifier_in_list(val, id_name [, found_val, not_found_val])

class calibre.utils.formatter_functions.**BuiltinIdentifierInList**

identifier_in_list(val, id_name [, found_val, not_found_val]) – traite val comme une liste d'identifiants séparés par des virgules. Un identifiant a le format « id_name :value ». Le paramètre « id_name » est le texte de l'identifiant à rechercher, soit « id_name » soit « id_name :regexp ». Dans le premier cas, la recherche s'effectue s'il existe un identifiant correspondant à ce nom d'utilisateur. Dans le second cas, la recherche se fait si le « id_name » correspond à un identifiant et que l'expression rationnelle correspond à la valeur de l'identifiant. Si « found_val » et « not_found_val » sont fournis, alors s'il y a une correspondance, il retourne « found_val », sinon il retourne « not_found_val ». Si « found_val » et « not_found_val » ne sont pas fournis, alors s'il y a une correspondance, alors la paire « identifier :value » est retournée, sinon une chaîne vide.

in_list(val, separator, [pattern, found_val,]+ not_found_val)

class calibre.utils.formatter_functions.**BuiltinInList**

in_list(val, separator, [pattern, found_val,]+ not_found_val) – Traite val comme une liste d'éléments séparés par un séparateur, si le motif correspond à l'une des valeurs de la liste, retourne found_val. Si le motif ne correspond à aucune valeur de la liste, retourne not_found_val. Les paires pattern et found_value peuvent être répétées autant de fois que souhaité. Les patterns sont vérifiés dans l'ordre. La valeur found_val de la première correspondance est retournée. Alias : in_list(), list_contains()

list_item(val, index, separator)

class calibre.utils.formatter_functions.**BuiltinListitem**

list_item(val, index, separator) – interprète la valeur comme une liste d'articles séparée par *separator*, retourne l'article *index*. Le premier article est le nombre zéro. Le dernier article peut être retourné en utilisant *list_item(-1,separator)*. Si l'article n'est pas dans liste, alors une valeur vide est renvoyée. Le séparateur a la même signafication que dans la fonction *count*.

select(val, key)

class calibre.utils.formatter_functions.**BuiltinSelect**

select(val, key) – interprète la valeur comme une liste d'éléments séparés par des virgules, les éléments étant « id :value ». Trouve la paire dont l'id est égal à key, et retourne la valeur correspondante. Retourne la chaîne vide si aucune correspondance n'est trouvée.

str_in_list(val, separator, [string, found_val,]+ not_found_val)

class calibre.utils.formatter_functions.**BuiltinStrInList**

str_in_list(val, separator, [string, found_val,]+ not_found_val) – Traite val comme une liste d'éléments séparés par un séparateur, si la chaîne de caractères correspond à l'une des valeurs de la liste, elle renvoie found_val. La comparaison est exacte (pas de contenu) et n'est pas sensible à la casse. Les paires string et found_value peuvent être répétées autant de fois que souhaité. Les patterns sont vérifiés dans l'ordre. La valeur found_val de la première correspondance est renvoyée.

List manipulation

count(separator)

class calibre.utils.formatter_functions.**BuiltinCount**

count(separator) – interprète la valeur comme une liste d'articles séparés par *separator*, retourne le nombre d'articles dans la liste. Beaucoup de listes utilisent une virgule comme séparateur, mais les auteurs utilisent l'esperluette. Exemples : {tags :count(,)}, {authors :count(&)}. Alias : count(), list_count()

list_count_matching(list, pattern, separator)

class calibre.utils.formatter_functions.**BuiltinListCountMatching**

list_count_matching(list, pattern, separator) – interprète "list" comme une liste d'éléments séparés par "separator", en retournant le nombre d'éléments dans la liste qui correspondent à l'expression régulière "pattern". Alias : list_count_matching(), count_matching()

list_difference(list1, list2, separator)

class calibre.utils.formatter_functions.**BuiltinListDifference**

list_difference(list1, list2, separator) – restitue une liste en supprimant de `list1` tout élément trouvé dans `list2` en utilisant une comparaison insensible à la casse. Les éléments dans `list1` et `list2` sont séparés par le séparateur, comme le sont les éléments dans la liste restituée.

list_equals(list1, sep1, list2, sep2, yes_val, no_val)

class calibre.utils.formatter_functions.**BuiltinListEquals**

list_equals(list1, sep1, list2, sep2, yes_val, no_val) –retourne yes_val si la liste 1 et la liste 2 contiennent les mêmes éléments, sinon renvoie no_val. Les éléments sont déterminés en éclatant chaque liste à l'aide du séparateur approprié (sep1 ou sep2). L'ordre des éléments dans la liste n'est pas pris en compte. La comparaison est insensible à la casse.

list_intersection(list1, list2, separator)

class calibre.utils.formatter_functions.**BuiltinListIntersection**

list_intersection(list1, list2, separator) – restitue une liste en supprimant de list1 tout élément non trouvé dans list2 en utilisant une comparaison insensible à la casse. Les éléments dans list1 et list2 sont séparés par le séparateur, comme le sont les éléments dans la liste restituée.

list_join(with_separator, list1, separator1 [, list2, separator2]*)

class calibre.utils.formatter_functions.**BuiltinListJoin**

list_join(with_separator, list1, separator1 [, list2, separator2]*) – retourne une liste faite en joignant les éléments des listes sources (list1 etc) en utilisant with_separator entre les éléments de la liste résultante. Les éléments de chaque liste[123…] source sont séparés par le séparateur[123…] associé. Une liste peut contenir zéro valeur. Il peut s'agir d'un champ comme publisher qui n'a qu'une seule valeur, c'est à dire une liste d'un seul élément. Les doublons sont supprimés en utilisant une comparaison insensible à la casse. Les éléments sont retournés dans l'ordre où ils apparaissent dans les listes sources. Si les éléments des listes ne diffèrent que par la casse des lettres, c'est la dernière qui est utilisée. Tous les séparateurs peuvent comporter plus d'un caractère. Exemple :

program :
list_join("#@#", $authors, "&", $tags, ",")

Vous pouvez utiliser list_join sur les résultats des appels précédents à list_join comme suit :
program :

a = list_join("#@#", $authors, "&", $tags, ","); b = list_join("#@#", a, "#@#", $#genre, ",", $#people, "&")

Vous pouvez utiliser des expressions pour générer une liste. Par exemple, supposons que vous vouliez des éléments pour auteurs et #genre, mais avec le genre changé en mot "Genre
[" suivi de la première lettre du genre, c'est-à-dire que le genre "Fiction" devient "Genre][F". Ce qui suit fera l'affaire :]

program :
list_join("#@#", $authors, "&", list_re($#genre, ",", "^(.).*$", "Genre : 1"), ",")

list_re(src_list, separator, include_re, opt_replace)

class calibre.utils.formatter_functions.**BuiltinListRe**

list_re(src_list, separator, include_re, opt_replace) – Construit une liste en séparant d'abord src_list en éléments utilisant le caractère de séparation. Pour chaque élément dans la liste, vérifie s'il correspond à include_re. Si c'est le cas, alors il l'ajoute à la liste à renvoyer. Si opt_replace n'est pas une chaîne vide, alors effectue le remplacement avant d'ajouter l'élément à la liste renvoyée.

list_re_group(src_list, separator, include_re, search_re [, group_template]+)

class calibre.utils.formatter_functions.**BuiltinListReGroup**

list_re_group(src_list, separator, include_re, search_re [, group_template]+) – Comme list_re sauf que les remplacements ne sont pas optionnels. Elle utilise re_group(list_item, search_re, group_template, …) pour effectuer les remplacements sur la liste résultante.

list_remove_duplicates(list, separator)

class calibre.utils.formatter_functions.**BuiltinListRemoveDuplicates**

list_remove_duplicates(list, separator) – retourne une liste faite en supprimant les éléments en double dans la liste des sources. Si les éléments ne diffèrent qu'au cas par cas, le dernier d'entre eux est renvoyé. Les éléments de la liste source sont séparés par un séparateur, tout comme les éléments de la liste retournée.

list_sort(liste, direction, separator)

class calibre.utils.formatter_functions.**BuiltinListSort**

list_sort(liste, direction, separator) – retourne une liste triée en utilisant un tri insensible à la casse. Si la direction est zéro, la liste est triée de manière ascendante, sinon descendante. Les éléments des listes 1 et 2 sont séparés par le séparateur, comme ceux de la liste retournée.

list_split(list_val, sep, id_prefix)

class calibre.utils.formatter_functions.**BuiltinListSplit**

list_split(list_val, sep, id_prefix) – divise la list_val en valeurs séparées en utilisant "sep", puis affecte les valeurs à des variables nommées "id_prefix_N" où N est la position de la valeur dans la liste. Le premier élément a la position 0 (zéro). La fonction renvoie le dernier élément de la liste. Exemple : split("one :two :foo", " :", "var") est équivalent à var_0 = "one" ; var_1 = "two" ; var_2 = "foo".

list_union(list1, list2, separator)

class calibre.utils.formatter_functions.**BuiltinListUnion**

list_union(list1, list2, separator) – restitue une liste créée de la fusion des éléments dans list1 et list2, supprimant les doublons en utilisant une comparaison insensible à la casse. Si la casse des éléments diffère, celle de list1 est utilisée. Les éléments dans list1 et list2 sont séparés par le séparateur, comme ceux de la liste restituée. Alias : list_union(), merge_lists()

range(start, stop, step, limit)

class calibre.utils.formatter_functions.**BuiltinRange**

range(start, stop, step, limit) – renvoie une liste de nombres générés en bouclant sur la plage spécifiée par les paramètres start, stop, et step, avec une longueur maximale de limit. La première valeur produite est "start". Les valeurs suivantes next_v sont current_v+step. La boucle continue pendant que next_v < stop en supposant que le pas est positif, sinon pendant que next_v > stop. Une liste vide est produite si start échoue le test : start>=stop si step est positif. La limite définit la longueur maximale de la liste et a une valeur par défaut de 1000. Les paramètres start, step et limit sont facultatifs. L'appel de range() avec un seul argument spécifie stop. Deux arguments spécifient start et stop. Trois arguments spécifient start, stop et step. Quatre arguments spécifient start, stop, le step et limit. Exemples : range(5)->"0,1,2,3,4". range(0,5)->"0,1,2,3,4". range(-1,5)->"-1,0,1,2,3,4". range(1,5)->"1,2,3,4". range(1,5,2)->"1,3". range(1,5,2)->"1,3". range(1,5,2,5)->"1,3". range(1,5,2,1)->error(limite dépassée).

subitems(val, start_index, end_index)

class calibre.utils.formatter_functions.**BuiltinSubitems**

subitems(val, start_index, end_index) – Cette fonction est utilisée pour séparer des listes d'éléments comme les genres. Elle interprète la valeur comme une liste d'éléments séparés par des virgules, où chaque élément est une liste séparée par des points. Retourne une nouvelle liste composée en trouvant d'abord tous les éléments séparés par des points, puis pour chacun de ces éléments en extrayant les composantes entre *start_index* et *end_index* avant de les recombiner. Le premier élément dans une liste séparée par des points a un index de 0. Si un index est négatif, alors il part de la fin de la liste. Comme cas spécial, un fin_index de zéro est considéré comme étant la longueur de la liste. Des exemples utilisant le mode de modèle basique et supposant une valeur de #genre de « A.B.C » : {#genre :subitems(0,1)} retourne « A ». {#genre :subitems(0,2)} retourne « A.B ». {#genre :subitems(1,0)} retourne « B.C ». Supposant une valeur de #genre de « A.B.C, D.E.F », {#genre :subitems(0,1)} retourne « A, D ». {#genre :subitems(0,2)} retourne « A.B, D.E »

sublist(val, start_index, end_index, séparator)

class calibre.utils.formatter_functions.**BuiltinSublist**

sublist(val, start_index, end_index, séparator) – interprète la valeur comme une liste d'éléments séparés par *séparator*, retournant une nouvelle liste composée des éléments compris entre *start_index* et *fin_index*. Le premier élément est le numéro 0. Si un index est négatif, alors il part de la fin de la liste. Comme cas spécial, un fin_index de zéro est considéré comme étant la longueur de la liste. Des exemples utilisant le mode de modèle basique et supposant que la colonne des étiquettes (qui sont séparées par des virgules) contient « A, B, C » : {tags :sublist(0,1,\,)} retourne « A ». {tags :sublist(-1,0,\,)} retourne « C ». {tags :sublist(0,-1,\,)} retourne « A, B ».

Other

arguments(id[=expression] [, id[=expression]]*)

class calibre.utils.formatter_functions.**BuiltinArguments**

arguments(id[=expression] [, id[=expression]]*) – Utilisé dans un modèle stocké pour récupérer les arguments passés dans l'appel. Il déclare et initialise à la fois des variables locales, en fait des paramètres. Les variables sont positionnelles ; elles obtiennent la valeur du paramètre donné dans l'appel dans la même position. Si le paramètre correspondant n'est pas fourni dans l'appel, alors les arguments attribuent à cette variable la valeur par défaut fournie. S'il n'y a pas de valeur par défaut, alors la variable est définie comme une chaîne vide.

assign(id, val)

class calibre.utils.formatter_functions.**BuiltinAssign**

assign(id, val) – assigne val à id, puis retourne val. id doit être un identifiant, pas une expression. Cette fonction peut souvent être remplacée par l'opérateur =.

globals(id[=expression] [, id[=expression]]*)

class calibre.utils.formatter_functions.**BuiltinGlobals**

globals(id[=expression] [, id[=expression]]*) – Récupère les « variables globales » qui peuvent être passées dans le formateur. Il déclare et initialise les variables locales avec les noms des variables globales passées. Si la variable correspondante n'est pas fournie dans les variables globales transmises, il lui attribue la valeur par défaut fournie. S'il n'y a pas de valeur par défaut, alors la variable est définie comme une chaîne vide.

print(a[, b]*)

class calibre.utils.formatter_functions.**BuiltinPrint**

print(a[, b]*) – affiche les arguments sur la sortie standard. À moins que vous ne lanciez calibre depuis la ligne de commande (calibre-debug -g), la sortie ira dans un trou noir.

Recursion

eval(modèle)

class calibre.utils.formatter_functions.**BuiltinEval**

eval(modèle) – évalue le modèle, en passant les variables locales (celles assignées) à la place des métadonnées du livre. Cela permet en utilisant le traitement du modèle de construire des résultats complexes à partir de variables locales. Parce que les caractères { et } sont spéciaux, vous devez utiliser [[pour le caractère { et]] pour le caractère } ; ils sont convertis automatiquement. Notez également que les préfixes et les suffixes (la syntaxe *|prefix|suffix*) ne peuvent pas être utilisés dans les arguments de cette fonction lorsque le mode modèle de programme est utilisé.

template(x)

class calibre.utils.formatter_functions.**BuiltinTemplate**

template(x) – évalue x comme un modèle. L'évaluation est effectuée dans son propre contexte, cela veut dire que les variables ne sont pas partagées entre l'appelant et l'évaluation du modèle. Comme les caractères { et } sont spéciaux, vous devez utiliser [[pour le caractère { et]] pour le caractère } ; ils sont convertis automatiquement. Par exemple, template("[[title_sort]]") évaluera le modèle {title_sort} et retournera sa valeur. Notez également que les préfixes et les suffixes (la syntaxe *|prefix|suffix* syntax) ne peuvent pas être utilisés dans les arguments de cette fonction lorsque le mode modèle de programme est utilisé.

Relational

cmp(x, y, lt, eq, gt)

class calibre.utils.formatter_functions.**BuiltinCmp**

cmp(x, y, lt, eq, gt) – compare x et y après les avoir convertis en nombres. Retourne lt si x < y. Retourne eq si x == y. Sinon, retourne gt. Dans de nombreux cas, les opérateurs de comparaison numérique (>#, <#, ==# etc) peuvent remplacer cette fonction.

first_matching_cmp(val, cmp1, result1,]+, else_result)

class calibre.utils.formatter_functions.**BuiltinFirstMatchingCmp**

first_matching_cmp(val, cmp1, result1,]+, else_result) – compare « val < cmpN » dans la séquence, retourne resultN pour la première comparaison qui réussit. Retourne un autre résultat si aucune comparaison ne réussit. Exemple : first_matching_cmp(10,5,»petit»,10,»moyen»,15,»grand»,»géant») retourne « grand». Le même exemple avec une première valeur de 16 retourne « géant».

strcmp(x, y, lt, eq, gt)

class calibre.utils.formatter_functions.**BuiltinStrcmp**

strcmp(x, y, lt, eq, gt) – effectue une comparaison insensible à la casse de x et y en tant que chaînes de caractères. Retourne lt si x < y. Retourne eq si x == y. Sinon, retourne gt. Dans de nombreux cas, les opérateurs de comparaison lexicale (>, <, == etc) peuvent remplacer cette fonction.

strcmpcase(x, y, lt, eq, gt)

class calibre.utils.formatter_functions.**BuiltinStrcmpcase**

strcmpcase(x, y, lt, eq, gt) – effectue une comparaison sensible à la casse de x et y en tant que chaînes de caractères. Retourne lt si x < y. Retourne eq si x == y. Sinon, retourne gt. Note : Ce n'est PAS le comportement par défaut utilisé par calibre, par exemple, dans les opérateurs de comparaison lexicale (==, >, <, etc.). Cette fonction pourrait provoquer des résultats inattendus, utilisez de préférence strcmp() lorsque cela est possible.

String case changes

capitalize(val)

class calibre.utils.formatter_functions.**BuiltinCapitalize**

capitalize(val) – renvoie val en lettres capitales

lowercase(val)

class calibre.utils.formatter_functions.**BuiltinLowercase**

lowercase(val) – renvoie val en miniscule

titlecase(val)

class calibre.utils.formatter_functions.**BuiltinTitlecase**

titlecase(val) – renvoie val en casse titre

uppercase(val)

class calibre.utils.formatter_functions.**BuiltinUppercase**

uppercase(val) – renvoie val en majuscule

String manipulation

character(character_name)

class calibre.utils.formatter_functions.**BuiltinCharacter**

character(character_name) – renvoie le caractère nommé par character_name. Par exemple, character("newline") renvoie un caractère de nouvelle ligne ("n"). Les noms de caractères pris en charge sont "newline", "return", "tab" et "backslash".

re(val, pattern, replacement)

class calibre.utils.formatter_functions.**BuiltinRe**

re(val, pattern, replacement) – renvoie le champ après avoir appliquer l'expression régulière. Toutes les instances de *pattern* seront remplacées par *replacement*. Comme tout dans calibre, celles-ci sont des expressions régulières compatibles Python

re_group(val, pattern [, template_for_group]*)

class calibre.utils.formatter_functions.**BuiltinReGroup**

re_group(val, pattern [, template_for_group]*) – renvoie une chaîne faite en appliquant le modèle d'expression régulière à la valeur et en remplaçant chaque instance assortie par la chaîne calculée en remplaçant chaque groupe assorti par la valeur retournée par le modèle correspondant. La valeur assortie originale pour le groupe est disponible comme $. En mode programmation de modèle, comme pour le modèle et les fonctions eval, vous utiliserez [[pour { et]] pour}. L'exemple suivant en mode programmation de modèle recherche des séries avec plus d'un mot et des majuscules dans le premier mot : {series :"re_group($, « (S*)(.*)», « [[$:uppercase()]]», « [[$]]»)"}

shorten(val, left chars, middle text, right chars)

class calibre.utils.formatter_functions.**BuiltinShorten**

shorten(val, left chars, middle text, right chars) – Retourne une version raccourcie de val, constituée par les *left chars* caractères du début de val, suivis par *middle text* et enfin par *right chars* de la fin de la chaîne. *left chars* et *right chars* doivent être des entiers. Par exemple, supposons que le titre du livre est *Anciennes Lois Anglaises aux Temps d'Ivanhoe*, et que vous voulez que ce titre rentre dans un espace maximal de 15 caractères. Si vous utilisez {title :shorten(9,-,5)}, le resultat sera *Ancienn L-anhoe*. Si la longueur du champ est inférieure à caractères à gauche + caractères à droite + la longueur de *texte du milieu*, alors le champ ne sera pas modifié. Par exemple, le titre *Le Dôme* ne serait pas modifié.

strcat(a [, b]*)

class calibre.utils.formatter_functions.**BuiltinStrcat**

strcat(a [, b]*) – peut avoir tout nombre d'arguments. Renvoie la chaîne constituée par la concaténation de tous les arguments

strcat_max(max, string1 [, prefix2, string2]*)

class calibre.utils.formatter_functions.**BuiltinStrcatMax**

strcat_max(max, string1 [, prefix2, string2]*) – Retourne la chaîne de caractères formée de la concaténation des arguments. La valeur retournée est initialisée à string1. Les paires *Prefix,string* sont ajoutées au bout de la valeur tant que la taille de la chaîne de sortie est plus petite que *max*. String1 est retournée même si string1 est plus grande que max. Vous pouvez passer autant de paires de *prefix,string* que vous le souhaitez.

strlen(a)

class calibre.utils.formatter_functions.**BuiltinStrlen**

strlen(a) – Retourne la longueur de la chaîne de caractères passée en argument

substr(ch, start, end)

class calibre.utils.formatter_functions.**BuiltinSubstr**

substr(ch, start, end) – renvoie les caractères de début jusqu'à fin de la chaîne ch. Le premier caractère dans la chaine ch est le caractère 0. Si fin est négatif, cela indique que le comptage des caractères est effectué à partir de la droite. Si end est à zéro, cela indique le dernier caractère. Par exemple, substr("12345", 1, 0) renvoie "2345", et substr("12345", 1, -1) renvoie "234".

swap_around_articles(separator)

class calibre.utils.formatter_functions.**BuiltinSwapAroundArticles**

swap_around_articles(separator) – renvoie la val avec les articles déplacés à la fin. La valeur peut être une liste, auquel cas chaque membre de la liste est traité. Si la valeur est une liste, alors vous devez fournir le séparateur de valeur de liste. Si aucun séparateur n'est fourni, la valeur est traitée comme une valeur unique et non comme une liste.

swap_around_comma(val)

class calibre.utils.formatter_functions.**BuiltinSwapAroundComma**

swap_around_comma(val) – pour une valeur au format « B, A », renvoie « A B ». Ceci est très utile pour la conversion des noms du format LN, FN vers FN LN. Si il n'y a pas de virgule, la fonction renvoie val non modifiée

to_hex(val)

class calibre.utils.formatter_functions.**BuiltinToHex**

to_hex(val) – renvoie la chaîne encodée en hexadécimal. Ceci est utile lors de la construction d'URLs calibre.

transliterate(a)

class calibre.utils.formatter_functions.**BuiltinTransliterate**

transliterate(a) - Restitue une chaîne en alphabet latin représentant approximativement le son des mots de la chaîne source. Par exemple, avec la source « Фёдор Миха́йлович Достоевский » la fonction renverra « Fiodor Mikhailovich Dostoievskii ».

Template database functions

book_count(query, use_vl)

class calibre.utils.formatter_functions.**BuiltinBookCount**

book_count(query, use_vl) – retourne le nombre de livres trouvés en cherchant la requête. Si use_vl est 0 (zéro) alors les bibliothèques virtuelles sont ignorées. Cette fonction ne peut être utilisée que dans l'interface graphique.

book_values(column, query, sep, use_vl)

class calibre.utils.formatter_functions.**BuiltinBookValues**

book_values(column, query, sep, use_vl) – renvoie une liste des valeurs contenues dans la colonne « column », séparées par « sep », dans les livres trouvés en recherchant « query ». Si use_vl vaut 0 (zéro), les bibliothèques virtuelles sont ignorées. Cette fonction ne peut être utilisée que dans l'interface graphique.

extra_file_modtime(file_name, format_string)

class calibre.utils.formatter_functions.**BuiltinExtraFileModtime**

extra_file_modtime(file_name, format_string) – retourne l'heure de modification du fichier supplémentaire « file_name » dans le dossier « data/ » du livre s'il existe, sinon -1. La date de modification est formatée selon « format_spec » (voir format_date() pour plus de détails). Si « format_spec » est une chaîne vide, elle retourne l'heure de modification sous la forme d'un nombre de secondes en virgule flottante depuis Epoch. L'Epoch dépend du système d'exploitation. Cette fonction ne peut être utilisée que dans l'interface graphique.

extra_file_names(sep [, pattern])

class calibre.utils.formatter_functions.**BuiltinExtraFileNames**

extra_file_names(sep [, pattern]) – retourne une liste séparée de « sep » de fichiers supplémentaires dans le dossier « data/ » du livre. Si le paramètre optionnel « pattern », une expression régulière, est fourni, alors la liste est filtrée sur les fichiers qui correspondent à « pattern ». La correspondance est insensible à la casse. Cette fonction ne peut être utilisée que dans l'interface graphique.

extra_file_size(file_name)

class calibre.utils.formatter_functions.**BuiltinExtraFileSize**

extra_file_size(file_name) – retourne la taille en octets du fichier supplémentaire « file_name » dans le dossier « data/ » du livre s'il existe, sinon -1. Cette fonction ne peut être utilisée que dans l'interface graphique.

get_link(field_name, field_value)

class calibre.utils.formatter_functions.**BuiltinGetLink**

get_link(field_name, field_value) – récupère le lien pour le champ « field_name » avec la valeur « field_value ». S'il n'y a pas de lien attaché, il renvoie "". Exemple : get_link("tags", "Fiction") renvoie le lien attaché a l'étiquette « Fiction ».

get_note(field_name, field_value, plain_text)

class calibre.utils.formatter_functions.**BuiltinGetNote**

get_note(field_name, field_value, plain_text) – récupère la note pour le champ « field_name » avec la valeur « field_value ». Si « plain_text » est vide, le HTML de la note est retourné. Si « plain_text » n'est pas vide, la note est retournée en texte brut. Si la note n'existe pas, le système renvoie "" dans les deux cas. Exemple : get_note("tags", "Fiction", "") renvoie le code HTML de la note attachée a l'étiquette « Fiction ».

has_extra_files([pattern])

class calibre.utils.formatter_functions.**BuiltinHasExtraFiles**

has_extra_files([pattern]) – retourne le nombre de fichiers supplémentaires, sinon "" (la chaîne vide). Si le paramètre optionnel « pattern » (une expression régulière) est fourni, alors la liste est filtrée sur les fichiers qui correspondent à « pattern » avant que les fichiers ne soient comptés. La correspondance est insensible à la casse. Cette fonction ne peut être utilisée que dans l'interface graphique.

has_note(field_name, field_value)

class calibre.utils.formatter_functions.**BuiltinHasNote**

has_note(field_name, field_value) – renvoie "1" si la valeur « field_value » dans le champ « field_name » a une note attachée, "" sinon. Exemple : has_note("tags", "Fiction") renvoie "1" si le tag « Fiction » a une note attachée, sinon "".

other

set_globals(id[=expression] [, id[=expression]]*)

class calibre.utils.formatter_functions.**BuiltinSetGlobals**

set_globals(id[=expression] [, id[=expression]]*) – Définit les « variables globales » qui peuvent être passées au formateur. Les variables globales reçoivent le nom de l'id passé. La valeur de l'id est utilisée sauf si une expression est fournie.

API of the Metadata objects

The python implementation of the template functions is passed in a Metadata object. Knowing it's API is useful if you want to define your own template functions.

class calibre.ebooks.metadata.book.base.**Metadata**(*title*, *authors=('Inconnu(e)',)*, *other=None*, *template_cache=None*, *formatter=None*)

A class representing all the metadata for a book. The various standard metadata fields are available as attributes of this object. You can also stick arbitrary attributes onto this object.

Metadata from custom columns should be accessed via the get() method, passing in the lookup name for the column, for example : « #mytags ».

Use the *is_null()* (page 221) method to test if a field is null.

This object also has functions to format fields into strings.

The list of standard metadata fields grows with time is in *STANDARD_METADATA_FIELDS* (page 222).

Please keep the method based API of this class to a minimum. Every method becomes a reserved field name.

is_null(*field*)

Return True if the value of field is null in this object. "null" means it is unknown or evaluates to False. So a title of _("Unknown") is null or a language of "und" is null.

Be careful with numeric fields since this will return True for zero as well as None.

Also returns True if the field does not exist.

deepcopy(*class_generator=<function Metadata.<lambda>>*)

Do not use this method unless you know what you are doing, if you want to create a simple clone of this object, use `deepcopy_metadata()` instead. Class_generator must be a function that returns an instance of Metadata or a subclass of it.

get_identifiers()

Return a copy of the identifiers dictionary. The dict is small, and the penalty for using a reference where a copy is needed is large. Also, we don't want any manipulations of the returned dict to show up in the book.

set_identifiers(*identifiers*)

Set all identifiers. Note that if you previously set ISBN, calling this method will delete it.

set_identifier(*typ*, *val*)

If val is empty, deletes identifier of type typ

standard_field_keys()

return a list of all possible keys, even if this book doesn't have them

custom_field_keys()

return a list of the custom fields in this book

all_field_keys()

All field keys known by this instance, even if their value is None

metadata_for_field(*key*)

return metadata describing a standard or custom field.

all_non_none_fields()

Return a dictionary containing all non-None metadata fields, including the custom ones.

get_standard_metadata(*field*, *make_copy*)

return field metadata from the field if it is there. Otherwise return None. field is the key name, not the label. Return a copy if requested, just in case the user wants to change values in the dict.

get_all_standard_metadata(*make_copy*)

return a dict containing all the standard field metadata associated with the book.

get_all_user_metadata(*make_copy*)

return a dict containing all the custom field metadata associated with the book.

get_user_metadata(*field*, *make_copy*)

return field metadata from the object if it is there. Otherwise return None. field is the key name, not the label. Return a copy if requested, just in case the user wants to change values in the dict.

set_all_user_metadata(*metadata*)

store custom field metadata into the object. Field is the key name not the label

set_user_metadata(*field*, *metadata*)

store custom field metadata for one column into the object. Field is the key name not the label

remove_stale_user_metadata(*other_mi*)

Remove user metadata keys (custom column keys) if they don't exist in "other_mi", which must be a metadata object

template_to_attribute(*other*, *ops*)

Takes a list [(src,dest), (src,dest)], evaluates the template in the context of other, then copies the result to self[dest]. This is on a best-efforts basis. Some assignments can make no sense.

smart_update(*other*, *replace_metadata=False*)

Merge the information in *other* into self. In case of conflicts, the information in *other* takes precedence, unless the information in *other* is NULL.

format_field(*key*, *series_with_index=True*)

Returns the tuple (display_name, formatted_value)

to_html()

A HTML representation of this object.

calibre.ebooks.metadata.book.base.**STANDARD_METADATA_FIELDS**

The set of standard metadata fields.

```
'''
All fields must have a NULL value represented as None for simple types,
an empty list/dictionary for complex types and (None, None) for cover_data
'''

SOCIAL_METADATA_FIELDS = frozenset((
    'tags',             # Ordered list
    'rating',           # A floating point number between 0 and 10
    'comments',         # A simple HTML enabled string
    'series',           # A simple string
    'series_index',     # A floating point number
    # Of the form { scheme1:value1, scheme2:value2}
    # For example: {'isbn':'123456789', 'doi':'xxxx', ... }
    'identifiers',
))
```

(suite sur la page suivante)

(suite de la page précédente)

```
'''
The list of names that convert to identifiers when in get and set.
'''

TOP_LEVEL_IDENTIFIERS = frozenset((
    'isbn',
))

PUBLICATION_METADATA_FIELDS = frozenset((
    'title',            # title must never be None. Should be _('Unknown')
    # Pseudo field that can be set, but if not set is auto generated
    # from title and languages
    'title_sort',
    'authors',          # Ordered list. Must never be None, can be [_('Unknown')]
    'author_sort_map',  # Map of sort strings for each author
    # Pseudo field that can be set, but if not set is auto generated
    # from authors and languages
    'author_sort',
    'book_producer',
    'timestamp',        # Dates and times must be timezone aware
    'pubdate',
    'last_modified',
    'rights',
    # So far only known publication type is periodical:calibre
    # If None, means book
    'publication_type',
    'uuid',             # A UUID usually of type 4
    'languages',        # ordered list of languages in this publication
    'publisher',        # Simple string, no special semantics
    # Absolute path to image file encoded in filesystem_encoding
    'cover',
    # Of the form (format, data) where format is, e.g. 'jpeg', 'png', 'gif'...
    'cover_data',
    # Either thumbnail data, or an object with the attribute
    # image_path which is the path to an image file, encoded
    # in filesystem_encoding
    'thumbnail',
))

BOOK_STRUCTURE_FIELDS = frozenset((
    # These are used by code, Null values are None.
    'toc', 'spine', 'guide', 'manifest',
))

USER_METADATA_FIELDS = frozenset((
    # A dict of dicts similar to field_metadata. Each field description dict
    # also contains a value field with the key #value#.
    'user_metadata',
))

DEVICE_METADATA_FIELDS = frozenset((
    'device_collections',   # Ordered list of strings
    'lpath',                # Unicode, / separated
    'size',                 # In bytes
    'mime',                 # Mimetype of the book file being represented
))
```

(suite sur la page suivante)

(suite de la page précédente)

```
CALIBRE_METADATA_FIELDS = frozenset((
    'application_id',   # An application id, currently set to the db_id.
    'db_id',            # the calibre primary key of the item.
    'formats',          # list of formats (extensions) for this book
    # a dict of user category names, where the value is a list of item names
    # from the book that are in that category
    'user_categories',
    # a dict of items to associated hyperlink
    'link_maps',
))

ALL_METADATA_FIELDS =       SOCIAL_METADATA_FIELDS.union(
                            PUBLICATION_METADATA_FIELDS).union(
                            BOOK_STRUCTURE_FIELDS).union(
                            USER_METADATA_FIELDS).union(
                            DEVICE_METADATA_FIELDS).union(
                            CALIBRE_METADATA_FIELDS)

# All fields except custom fields
STANDARD_METADATA_FIELDS = SOCIAL_METADATA_FIELDS.union(
                            PUBLICATION_METADATA_FIELDS).union(
                            BOOK_STRUCTURE_FIELDS).union(
                            DEVICE_METADATA_FIELDS).union(
                            CALIBRE_METADATA_FIELDS)

# Metadata fields that smart update must do special processing to copy.
SC_FIELDS_NOT_COPIED =      frozenset(('title', 'title_sort', 'authors',
                                       'author_sort', 'author_sort_map',
                                       'cover_data', 'tags', 'languages',
                                       'identifiers'))

# Metadata fields that smart update should copy only if the source is not None
SC_FIELDS_COPY_NOT_NULL =  frozenset(('device_collections', 'lpath', 'size', 'comments
↪', 'thumbnail'))

# Metadata fields that smart update should copy without special handling
SC_COPYABLE_FIELDS =        SOCIAL_METADATA_FIELDS.union(
                            PUBLICATION_METADATA_FIELDS).union(
                            BOOK_STRUCTURE_FIELDS).union(
                            DEVICE_METADATA_FIELDS).union(
                            CALIBRE_METADATA_FIELDS) - \
                            SC_FIELDS_NOT_COPIED.union(
                            SC_FIELDS_COPY_NOT_NULL)

SERIALIZABLE_FIELDS =       SOCIAL_METADATA_FIELDS.union(
                            USER_METADATA_FIELDS).union(
                            PUBLICATION_METADATA_FIELDS).union(
                            CALIBRE_METADATA_FIELDS).union(
                            DEVICE_METADATA_FIELDS) - \
                            frozenset(('device_collections', 'formats',
                                'cover_data'))
# these are rebuilt when needed
```

10.4 Tout à propos de l'utilisation des expressions régulières dans calibre

Les expressions régulières sont des fonctions utilisées à différents endroits dans calibre pour effectuer des manipulations élaborées sur le contenu et sur les métadonnées de vos livres numériques. Ce tutoriel est une introduction légère pour vous permettre de démarrer dans l'utilisation des expressions régulières de calibre.

Contenu

- *Tout d'abord, un mot d'avertissement et un mot d'encouragement* (page 225)
- *Où pouvez-vous utiliser les expressions régulières dans calibre ?* (page 226)
- *Que diable* est *une expression régulière ?* (page 226)
- *Prendre soin d'expliquer ?* (page 226)
- *Cela ne semble pas trop mauvais. Ensuite ?* (page 226)
- *He, joli ! Cela commence à avoir du sens !* (page 227)
- *Bien, ces caractères spéciaux sont bien jolis et tout, mais qu'en est-il si je veux faire correspondre à un point ou à un point d'interrogation ?* (page 227)
- *Bien, quels sont les ensembles les plus utiles ?* (page 228)
- *Mais si j'avais quelques chaînes variables que je voudrais faire correspondre, les choses deviennent-elles compliquées ?* (page 228)
- *Vous avez manqué…* (page 229)
- *Au début vous disiez qu'il y avait un moyen de rendre les expressions régulières insensibles à la casse ?* (page 229)
- *Je pense que je commence à comprendre ces expressions régulières maintenant… comment dois-je les utiliser dans calibre ?* (page 229)
  - *Conversions* (page 229)
  - *Ajout de livres* (page 230)
  - *Edition des métadonnées par lot* (page 230)
- *Mémento* (page 231)
- *Crédits* (page 236)

10.4.1 Tout d'abord, un mot d'avertissement et un mot d'encouragement

Ceci est, inévitablement, en train de devenir quelque peu technique- après tout, les expressions régulières sont des outils techniques pour faire une chose technique. Je vais devoir utiliser un certain jargon et des concepts qui peuvent sembler compliqués ou alambiqués. Je vais essayer d'expliquer ces concepts aussi clairement que possible, mais je ne peux vraiment pas le faire sans les utiliser du tout… Donc, ne soyez pas découragé par un tel jargon, comme j'ai essayé d'expliquer tout ce qui est nouveau. Et alors que les expressions régulières en elles mêmes peuvent paraître comme une arcane, de la magie noire (ou, pour être plus prosaïque, une chaîne aléatoire d'un charabia de lettres et de signes). Je promets qu'elles n'ont pas toutes cette complexité. Même ceux qui comprennent bien les expressions régulières ont des difficultés à lire les plus complexes, mais les écrire n'est pas une difficulté- vous construisez l'expression pas à pas. Aussi, faites un pas et suivez-moi dans le trou du lapin.

10.4.2 Où pouvez-vous utiliser les expressions régulières dans calibre ?

Il y a dans calibre quelques endroits utilisant des expressions régulières. Il y en *Rechercher & remplacer* dans les options de conversion, dans la détection de métadonnées à partir des noms de fichiers dans les paramètres d'importation ainsi que dans Rechercher et remplacer lors de l'édition des métadonnées de livres par lot. L'éditeur de livre de calibre peut également utiliser des expressions régulières dans sa fonctionnalité *Rechercher et remplacer*. Enfin, vous pouvez utiliser des expressions régulières lors de la recherche dans la liste des livres de calibre et lors de la recherche dans la visionneuse de calibre.

10.4.3 Que diable *est* une expression régulière ?

Une expression régulière est une manière de décrire des ensembles de chaînes. Une expression régulière simple peut *correspondre* à un nombre différent de chaînes. C'est ce qui rend les expressions régulières si puissantes – elles sont une manière concise de décrire un nombre potentiellement grand de variations.

Note : J'utilise ici le mot chaîne dans le sens où il est utilisé dans les langages de programmation : une chaîne d'un ou plusieurs caractères, caractères incluant des caractères habituels, des nombres, de la ponctuation et ce qu'on appelle les whitespace (retours de ligne, tabulations, etc…). Veuillez noter que généralement, les caractères majuscules et minuscules ne sont pas considérés les mêmes, donc « a » est un caractère différent de « A » et ainsi de suite. Dans calibre, les expressions régulières sont sensibles à la casse dans la Barre de recherche, mais pas dans les options de conversion. Il y a une manière pour rendre chaque expression régulière insensible à la casse, mais nous en discuterons plus tard. Ça devient compliqué parce que les expressions régulières tiennent compte des variations des chaînes qu'elles font correspondre, ainsi une expression peut correspondre à des chaînes multiples, ce qui est pourquoi les gens ne prennent pas la peine de les employer du tout. Plus sur cela dans un moment.

10.4.4 Prendre soin d'expliquer ?

Bien, c'est pourquoi nous sommes ici. D'abord, c'est le concept le plus important dans les expressions régulières : *Une chaîne en-soi est une expression régulière qui correspond à elle même*. Ceci est pour dire, si je voulais faire correspondre la chaîne `"Hello, World!"` en utilisant une expression régulière, l'expression régulière à utiliser devra être `"Hello, World!"`. Et oui, c'est vraiment aussi simple. Vous noterez, cependant, que cela correspond *uniquement* à la chaîne exacte `"Hello, World!"`, pas par ex. à `"Hello, wOrld!"` ou à `"hello, world!"` ou à n'importe quelle autre variation.

10.4.5 Cela ne semble pas trop mauvais. Ensuite ?

La suite est le commencement des choses vraiment intéressantes. Vous souvenez-vous quand j'ai dit que les expressions régulières peuvent correspondre à plusieurs chaînes ? C'est là où ça les rendait un peu plus compliquées. Disons, comme une sorte d'exercice plus pratique, que le livre numérique que vous vouliez convertir avait de méchantes notes de bas de page comptant les pages, comme « Page 5 de 423 ». Évidemment le nombre de pages devrait augmenter de 1 à 423, donc vous devriez avoir à faire correspondre 423 chaînes, juste ? Faux, en fait : les expressions régulières vous permettent de définir un ensemble de caractères qui correspond : Pour définir un ensemble, vous mettez tous les caractères que vous voulez y voir figurer entre crochets. Aussi, par exemple, l'ensemble `[abc]``devra correspondre soit au caractère "a", soit au "b" ou au "c". *Les ensembles correspondront toujours uniquement à un des caractères dans l'ensemble*, c'est à dire, si vous voulez faire correspondre tous les caractères minuscules, vous devrez utiliser l'ensemble ``[a-z]` pour les caractères minuscules et majuscules vous devrez utiliser l'ensemble `[a-zA-Z]` et ainsi de suite. Vous voyez l'idée ? Ainsi, évidemment, en utilisant l'expression `Page [0-9] de 423` vous serez capable de correspondre aux 9 premières pages, donc par conséquent les expressions en ont besoin de trois : la seconde expression

`Page [0-9][0-9] de 423` correspondra aux numéros de pages à deux chiffres, et je suis sûr que vous pouvez deviner à quoi ressemblera la troisième expression. Oui, allez-y. Mettez-la par écrit.

10.4.6 He, joli ! Cela commence à avoir du sens !

J'espère que vous vous dites cela. Mais accrochez vous, maintenant cela va devenir encore mieux ! Nous avons juste dit qu'en utilisant les ensembles, nous pouvons correspondre à un de plusieurs caractères en une fois. Mais vous pouvez même répéter un caractère ou un ensemble, réduisant le nombres d'expressions nécessaires pour gérer l'exemple numéro de page ci dessus à une. Oui, UNE ! Enthousiaste ? Vous devriez l'être ! Cela fonctionne comme ceci : certains caractères qu'on appelle spéciaux, « +», « ?» et « *», répète l'élément simple le précédant. (Élément signifie soit un caractère seul, un ensemble de caractères, une séquence d'échappement ou un groupe) (nous aborderons ces deux derniers plus tard)- en bref, n'importe quelle entité simple dans une expression régulière). Ces caractères sont appelés des caractères de remplacement ou des quantificateurs. Pour être plus précis, « ?» correspond à *0 ou 1* de l'élément précédant, « *» correspond à *0 ou plus* de l'élément précédant et « +» correspond à *1 ou plus* de l'élément précédant. Quelques exemples : l'expression `a?` correspondrait soit à « » (qui est la chaîne vide, pas strictement utile dans ce cas) ou « a», l'expression « a*» correspondrait à « « , « a», « aa» ou n'importe quel nombre de a dans une ligne, finalement, l'expression « a+» correspondrait à « a», « aa» ou n'importe quel nombre de a dans une ligne(Notez : il ne correspondra pas à une chaîne vide !). Même affaire avec les ensembles, l'expression `[0-9]+` correspondra à *chaque numéro entier qu'il y a* ! Je sais à quoi vous êtes occupé à penser, et vous avez raison : si vous utilisez cela dans le cas ci-dessus de la correspondance du numéro de page, est-ce que cette simple expression ne pourrait pas correspondre à tous les numéros de page ? Oui, l'expression `[0-9]+ de 423` correspondra à chaque numéro de page dans ce livre !

Note : Une note sur ces quantificateurs : Ils essaient généralement de correspondre au plus de texte possible, aussi soyez prudent lors de leur utilisation. Ceci est appelé « un comportement glouton »- Je suis sûr que vous comprenez pourquoi. Ils deviennent problématiques quand vous, disons, essayez de correspondre à une balise. Considérons, par exemple, la chaîne `"<p class="calibre2">Titre ici</p>"` et disons que vous voulez correspondre à la balise d'ouverture (la partie entre la première paire de chevrons, un peu plus sur les balises plus tard). Vous devez penser que l'expression `<p.*>` devrait correspondre à cette balise, mais actuellement, elle correspond à l'entièreté de la chaîne ! (Le caractère « .» est un autre caractère spécial. Il correspond à n'importe quoi *excepté* les sauts de ligne, aussi, essentiellement, l'expression `.*` devrait correspondre à n'importe quelle ligne à laquelle vous pensez). A la place, essayez d'utiliser `<p.*?>` qui rend le quantificateur `"*"` non-glouton. Cette expression ne correspondrait qu'à la première balise d'ouverture, comme attendu. Il y a en fait une autre manière d'y parvenir : L'expression `<p[^>]*>` correspondra à la même balise d'ouverture - vous verrez pourquoi après la prochaine section. Notez juste qu'il y a assez fréquemment plus d'une manière d'écrire une expression régulière.

10.4.7 Bien, ces caractères spéciaux sont bien jolis et tout, mais qu'en est-il si je veux faire correspondre à un point ou à un point d'interrogation ?

Vous pouvez bien sûr faire cela : insérez juste une barre oblique inversée devant tout caractère spécial et il sera interprété comme un caractère littéral, sans aucune autre signification. Cette paire de barres obliques inversées suivie d'un caractère seul est appelée une séquence d'échappement, et l'action de mettre une barre oblique inversée devant un caractère spécial est appelée un échappant de ce caractère. Une séquence d'échappement est interprétée comme un élément seul. Il y a bien sûr des séquences d'échappement qui font plus que juste échapper des caractères spéciaux, par exemple `"\t"` signifie une tabulation. Nous verrons certaines des séquences d'échappement plus tard. Ah, et temps que nous y sommes, au sujet de ces caractères spéciaux : Considérez n'importe quel caractère dont nous discutons dans cette introduction comme ayant une certaine fonction pour être spéciaux et de ce fait devant être échappé si vous voulez le caractère littéral.

10.4.8 Bien, quels sont les ensembles les plus utiles ?

Je savais que vous demanderiez. Certains des ensembles utiles sont `[0-9]` correspondant à un nombre unique, `[a-z]` correspondant à un caractère minuscule unique, `[A-Z]` correspondant à un caractère majuscule unique, `[a-zA-Z]` correspondant à un caractère unique et `[a-zA-Z0-9]` correspondant à une lettre ou un nombre unique. Vous pouvez aussi utiliser une séquence d'échappement comme raccourci :

\d
: est équivalent à `[0-9]`

\w
: est équivalent à `[a-zA-Z0-9_]`

\s
: est équivalent à tout espace blanc

Note : « L'espace blanc » est un terme désignant tout ce qui ne sera pas imprimé. Ces caractères comprennent l'espace, la tabulation, le saut de ligne, le saut de page, le retour chariot, les espaces insécables, etc.

Note : Les jeux de majuscules et de minuscules peuvent correspondre à la fois aux majuscules et aux minuscules si le paramètre permettant de rendre les recherches insensibles à la casse est activé. De tels paramètres se trouvent, par exemple, dans Préférences->Recherche dans calibre lui-même et sur le panneau de recherche dans la visionneuse de calibre *Livre numérique* ainsi que dans l'outil calibre *Editer un livre*.

Comme dernière note sur les ensembles, vous pouvez aussi définir un ensemble de n'importe quel caractère*mais* ceux dans l'ensemble. Vous faites cela en incluant le caractère **"^"** comme le *tout premier caractère dans l'ensemble*. Donc, `[^a]` correspondra à n'importe quel caractère en excluant le « a ». C'est ce qu'on appelle complémenter l'ensemble. Ces sténographies de séquences d'échappement que nous avons vues plus tôt peuvent aussi être complémentées : **"\D"** signifie n'importe quel caractère qui n'est pas un nombre, donc étant équivalent à `[^0-9]`. Les autres sténographies peuvent être complémentées par, vous le devinez, l'utilisation de la majuscule respective à la place de la minuscule. Aussi, revenons à l'exemple `<p[^>]*>` de la section précédente, maintenant vous pouvez voir que le jeu de caractères utilise des essais pour correspondre à n'importe quel caractère excepté un chevron fermant.

10.4.9 Mais si j'avais quelques chaînes variables que je voudrais faire correspondre, les choses deviennent-elles compliquées ?

N'ayez pas peur, la vie est belle et facile. Considérons cet exemple : Le livre que vous convertissez a « Titre » écrit sur chaque page impaire et « Auteur » écrit sur chaque page paire. Harmonieux à l'impression, pas vrai ? Mais dans les livres numériques, c'est ennuyeux. Vous pouvez grouper ensemble les expressions dans des parenthèses normales et le caractère **"|"** vous permettra de correspondre soit à l'expression à sa droite *ou* à l'expression à sa gauche. Combinez les et vous avez fini. Trop rapide pour vous ? Ok, tout d'abord, nous groupons les expressions pour les pages paires et impaires, donc nous obtenons *(Titre)(Auteur)`* comme nos deux expressions nécessaires. Maintenant nous rendons les choses plus simples en utilisant la barre verticale (**"|"** est appelé le caractère barre verticale) : si vous utilisez l'expression `(Titre|Auteur)` vous obtiendrez une correspondance pour « Titre » (sur les pages impaires) ou vous correspondrez à « Auteur » (sur les pages paires). Bien, n'était-ce pas facile ?

Vous pouvez, bien sûr, utiliser la barre verticale sans utiliser les parenthèses de regroupement, aussi bien. Vous vous souvenez quand je disais que les caractères spéciaux répétaient l'élément le précédant, et bien pour le caractère barre verticale, c'est différent.L'expression « Titre|Auteur » permet de sélectionner soit auteur, soit titre, comme sur l'exemple précédent avec les parenthèses. *Le caractère barre verticale sélectionne l'expression devant lui ou celle derrière lui*. Donc si on veut sélectionner les chaînes « calibre » et « Calibre », il faudra utiliser l'expression (c|C)alibre, où le groupement

assure qu'uniquement le « c » sera sélectionné. Si on avait utilisé `c|Calibre`, alors on aurait obtenu c ou Calibre, ce qui n'est pas ce qui était attendu. Donc en cas de doute, il est nécessaire de grouper avec les parenthèses.

10.4.10 Vous avez manqué...

… Attendez une minute, il y a une dernière chose, vraiment bien, que vous pouvez faire avec les groupes. A partir d'un groupe que vous avez déjà récupéré, vous pouvez utiliser une référence à ce groupe plus tard dans l'expression. Les groupes trouvés sont numérotés à partir de 1, et vous pouvez y faire référence en échappant le numéro du groupe auquel vous vous référencé, donc le cinquième groupe devra être référencé comme `\5`. Aussi si vous recherchiez après `([^ ]+) \1` dans la chaîne « Test Test », vous correspondriez à la chaîne entière !

10.4.11 Au début vous disiez qu'il y avait un moyen de rendre les expressions régulières insensibles à la casse ?

Oui, je l'ai fait, merci de votre attention et de me l'avoir rappelé. Vous pouvez dire à calibre comment vous voulez que certaines choses soient traitées en utilisant quelque chose appelée bannière. Vous incluez des bannières dans votre expression en utilisant la construction spéciale `(?bannières vont ici)` où, évidemment, vous devrez remplacer « bannières vont ici » avec les bannières spécifiques que vous voulez. Pour ignorer la casse, la bannière est le `i`, donc vous incluez `test(?i)` dans votre expression. Dès lors, `(?i)test` correspondra à Test, tEst, TEst et toutes les variations imaginables.

Une autre bannière utile laisse le point corresponde à n'importe quel caractère du tout, *incluant* le saut de ligne, la bannière `s`. Si vous voulez utiliser de multiples bannières dans une expression, mettez les juste dans la même déclaration : `('?is)` ignorera la casse fera que le point correspondra à tout. Quelle bannière vous placez en premier n'est pas important, `(?si)` sera équivalent à celui ci-dessus.

10.4.12 Je pense que je commence à comprendre ces expressions régulières maintenant... comment dois-je les utiliser dans calibre ?

Conversions

Commençons avec les paramètres de conversion, qui est vraiment bien. Dans la partie *Rechercher & remplacer*, on peut utiliser une regexp (abréviation pour expression régulière) qui décrit la chaîne qui sera remplacée pendant la conversion. La partie magique, c'est l'assistant. En cliquant sur l'assistant, on obtient la prévisualisation de ce que calibre « verra » pendant le processus de conversion. Défilez vers le bas jusqu'à la chaque vous voulez supprimer, sélectionnez-la et copiez-la, collez la dans le champ regexp au dessus de la fenêtre. S'ils ont des parties variables, comme les numéros de page ou autre, utilisez les ensembles et les quantificateurs pour couvrir ceux-ci, et pendant que vous y êtes, souvenez-vous d'échapper les caractères spéciaux, s'il y en a certains. Appuyez le bouton libellé *Test* et calibre mettra en évidence les parties qu'il devrait remplacer s'il utilisait la regexp. Une fois que vous êtes satisfait, appuyez OK et convertissez. Soyez prudent si votre source de conversion à des balises comme dans cet exemple :

```
Maybe, but the cops feel like you do, Anita. What's one more dead vampire?
New laws don't change that. </p>
<p class="calibre4"> <b class="calibre2">Generated by ABC Amber LIT Conv
<a href="http://www.processtext.com/abclit.html" class="calibre3">erter,
http://www.processtext.com/abclit.html</a></b></p>
<p class="calibre4"> It had only been two years since Addison v. Clark.
The court case gave us a revised version of what life was
```

(honteusement tiré de ce fil de discussion[88]). Vous devriez également enlever certaines des balises . Dans cet exemple, je recommanderais de commencer par la balise `<b class="calibre2">`, maintenant vous devez terminé avec

[88] https://www.mobileread.com/forums/showthread.php?t=75594"

la balise fermante correspondante (les balises ouvrantes sont `<balise>`, les balises fermantes sont `</tag>`), qui est simplement le prochain `</b>` dans ce cas. (Référez-vous à un bon manuel HTML ou posez la question sur le forum si vous n'êtes pas au clair sur ce point). La balise ouvrante peut être décrite en utilisant `<b.*?>`, la balise fermante en utilisant `</b>`, donc nous pourrions retirer entièrement ce qui se trouve entre ces deux balises en utilisant `<b.*?>.*?</b>`. Mais utiliser cette expression serait une mauvaise idée, parce qu'elle retire tout ce qui se trouve encadré par des balises <b>- (qui, par ailleurs, rend le texte encadré en gras), et c'est un pari gagné que nous enlèverons des parties du livre de cette façon. À la place, inclure également le commencement de la chaîne encadrée, faisant l'expression régulière `<b.*?>\s*Generated\s+by\s+ABC\s+Amber\s+LIT.*?</b>` Le `\s` avec les quantificateurs sont inclus ici à la place d'utiliser explicitement les espaces comme considéré dans la chaîne pour saisir toutes les variations de la chaîne qui pourraient survenir. Souvenez-vous de vérifier qu'est ce que calibre retirera pour vous assurer que vous ne retirez pas des portions que vous voulez garder si vous testez une nouvelle expression. Si vous vérifiez uniquement une occurrence, vous pourriez manquer une disparité ailleurs dans le texte. Notez également que si vous enlevez accidentellement plus ou moins de balises que vous ne le vouliez réellement, calibre essaye de réparer le code endommagé après avoir fait le retrait.

Ajout de livres

Une autre chose pour laquelle vous pouvez utiliser les expressions régulières est l'extraction des métadonnées à partir des noms de fichier. Vous pouvez trouver cette fonctionnalité dans la partie *Ajout de livres* des préférences. Il y a une fonctionnalité spéciale ici : vous vous utiliser les noms de champ pour les champs de métadonnées, par exemple `(?P<title>)` indiquera que calibre utilise cette parie de la chaîne comme titre du livre. Les noms de champs permis sont listés dans la fenêtre, accompagné d'un autre joli champ de test. Un exemple : disons que vous voulez importer un tas de fichiers nommés `Textes Classiques: La Divine Comédie par Dante Alighieri.mobi`. (Evidemment, ceci est déjà dans votre bibliothèque, puisque nous aimons tous la poésie italienne classique) ou `Les épiques de la Science Fiction : La Trilogie Fondation par Isaac Asimov.epub`. Ceci est évidemment un schéma de nommage dont calibre ne peut extraire aucunes données significatives - son expression standard pour l'extraction de métadonnées est `(?P<title>.+) - (?P<author>[^_]+)`. Une expression régulière qui fonctionnera ici devra être `[a-zA-Z]+: (?P<title>.+) par (?P<author>.+)`. Merci de noter que, à l'intérieur du groupe pour le champ métadonnées , vous devez utiliser des expressions pour décrire ce à quoi le champ correspond réellement. Et notez également que, lors de l'utilisation de la zone de test de calibre, vous devez ajouter l'extension de fichier, sinon vous n'aurez pas de résultats du tout, malgré l'utilisation d'une expression correcte.

Edition des métadonnées par lot

La dernière partie est le *Rechercher et remplacer* par expression régulière dans les champs de métadonnées. Vous pouvez accéder à ceci en sélectionnant de multiples livres dans la bibliothèque et en utilisant l'édition de métadonnées par lot. Soyez très prudent en utilisant cette dernière fonction, vu qu'elle peut faire de **Très Vilaines Choses** à votre bibliothèque. Vérifiez par deux fois que vos expressions font ce que vous attendez d'elles en utilisant les champs de test, et marquez uniquement les livres que vous voulez réellement changer ! Dans le mode recherche par expression régulière, vous pouvez rechercher dans un champ, remplacer le texte avec quelque chose et même écrire le résultat dans un autre champ. Un exemple pratique : disons que votre bibliothèque contient les livres de la série Dune de Frank Herbert, nommés selon le modèle `Dune 1 - Dune`, `Dune 2 - Dune Messiah` et ainsi de suite. Maintenant vous voulez obtenir `Dune` dans le champ série Vous pouvez faire cela en recherchant après `(.*?) \d+ - .*` dans le champ titre et en le remplaçant avec `\1` dans le champ séries. Vous voyez ce que je veux dire ? C'est une référence au premier groupe avec lequel vous allez remplacer le champ série. Maintenant que vous avez toutes vos séries prêtes, vous avez uniquement besoin de faire une autre recherche de `.*? -` et le remplacer avec `""` (une chaîne vide) encore dans le champ titre, et vos métadonnées sont toutes ordonnées et rangées. De cette manière, au lieu de remplacer le champ entier, vous pouvez également apposer ou ajouter au début au champ, ainsi, si vous *voulez* le titre de livre à ajouter au début avec des infos de séries, vous pourriez faire cela également. Comme vous l'avez surement noté, il y a une case à cocher marquée *Sensible à la casse*, ainsi vous ne devrez pas employer des bannières pour sélectionner le comportement ici.

Voilà, c'est juste pour conclure cette très courte introduction aux expressions régulières. Si tout va bien je vous aurai montré assez que pour vous permettre au moins de démarrer et de vous donner l'envie de continuer d'apprendre par

vous-même- un bon point de départ serait la Documentation Python pour les regexps[89].

Un dernier mot de l'avertissement, cependant : Les regexps sont puissantes, mais également vraiment facile à devenir erronées. calibre fournit des possibilités de vraiment grandes d'essai pour voir si vos expressions se comportent que vous vous y attendez à . Employez-les. Essayez de ne pas vous tirer dans le pied. (Dieu, j'aime cette expression…). Mais si vous vous, en dépit de l'avertissement, blessez votre pied (ou toutes autres parties du corps), essayez d'apprendre de celui-ci.

10.4.13 Mémento

Mémento pour la syntaxe regexp

Cet aide-mémoire résume les parties du moteur regexp les plus couramment utilisées/difficilement mémorisables disponibles dans la plupart des parties de calibre.

Contenu

- *Les classes de caractères* (page 231)
- *Les classes de caractères abrégées* (page 232)
- *Les quantificateurs* (page 232)
- *La gourmandise* (page 233)
- *L'alternative* (page 233)
- *L'exclusion* (page 233)
- *Les ancres* (page 233)
- *Groupes* (page 234)
- *Les assertiions* (page 234)
- *La récursion* (page 235)
- *Les caractères spéciaux* (page 235)
- *Les métacaractères* (page 235)
- *Modes* (page 236)

Les classes de caractères

Les classes de caractères sont utiles pour représenter différents groupes de caractères, succinctement.

Exemples :

[89] https://docs.python.org/library/re.html

Représentatio	Classe
`[a-z]`	Lettres minuscules. Ne comprend pas les caractères accentués et les ligatures
`[a-z0-9]`	Lettres minuscules de a à z ou chiffres de 0 à 9
`[A-Za-z-]`	Lettre majuscule ou minuscule, ou un tiret. Pour inclure le tiret dans une classe, il faut le mettre au début ou à la fin pour ne pas le confondre avec le tiret qui spécifie une plage de caractères
`[^0-9]`	Tout caractère sauf un chiffre. Le circonflexe (^) placé en début de classe exclut les caractères de la classe (classe complémentée)
`[[a-z]--[a`	Les consonnes minuscules. Une classe peut être incluse dans une classe. Les caractères `--` excluent ce qui les suit
`[\w--[\ d_]]`	Toutes les lettres (y compris les caractères accentués étrangers). Les classes abrégées peuvent être utilisées à l'intérieur d'une classe

Exemple :

```
<[^<>]+> to select an HTML tag
```

Les classes de caractères abrégées

Représentat	Classe
`\d`	Un chiffre (identique à [0-9])
`\D`	Tout caractère non numérique (identique à [^0-9])
`\w`	Un caractère alphanumérique plus le souligné (`[a-zA-Z0-9_]`) y compris les caractères accentués et les ligatures
`\W`	Tout caractère « non-word »
`\s`	Espace, espace insécable, tabulation, retour ligne
`\S`	Tout caractère « non-whitespace »
`.`	N'importe quel caractère sauf le retour ligne. Il faut cocher la case « dot all » ou utiliser la regexp `(?s)` pour inclure le caractère de retour de ligne

Les quantificateurs

Quantificateur	Nombre occurrences de l'expression précédant le quantificateur
`?`	0 ou 1 occurrence de l'expression. Identique à `{0,1}`
`+`	1 ou plusieurs occurrences de l'expression. Identique à `{1,}`
`*`	0, 1 ou plusieurs occurrences de l'expression. Identique à `{0,}`
`{n}`	Exactement n occurrences de l'expression
`{min,max}`	Nombre d'occurrences compris entre les valeurs minimales et maximales incluses
`{min,}`	Nombre d'occurrences compris entre la valeur minimale incluse et l'infini
`{,max}`	Nombre d'occurrences compris entre 0 et la valeur maximale incluse

La gourmandise

Par défaut, avec les quantificateurs, le moteur d'expressions rationnelles est gourmand (greedy) : il étend la sélection autant que possible. Cela réserve des surprises, au début. `?` suit un quantificateur pour le rendre non gourmand (lazy). Éviter d'en mettre deux dans la même expression, le résultat peut être imprévisible.

Attention à l'imbrication des quantificateurs, comme, par exemple, le motif `(a*)*` : il augmente de façon exponentielle les traitements.

L'alternative

Le caractère `|` dans une expression régulière est un `OR` logique. Cela signifie que soit l'expression précédente soit l'expression suivante peut correspondre.

L'exclusion

Méthode 1

```
motif_à_exclure(*SKIP)(*FAIL)|motif_à_sélectionner
```

Exemple :

```
"Blabla"(*SKIP)(*FAIL)|Blabla
```

sélectionne Blabla, dans les chaînes Blabla ou « Blabla ou Blabla », mais pas dans « Blabla ».

Méthode 2

```
motif_à_exclure\K|(motif_à_sélectionner)
```

```
"Blabla"\K|(Blabla)
```

sélectionne Blabla, dans les chaînes Blabla ou « Blabla ou Blabla », mais pas dans « Blabla ».

Les ancres

Une ancre est un moyen de faire correspondre une position logique dans une chaîne, plutôt qu'un caractère. Les ancres les plus utiles pour le traitement de texte sont :

`\b`
: Désigne une limite de mot, càd une transition depuis un espace à caractère non-espace. Par exemple, vous pouvez utiliser `\bsurd` pour correspondre à `the surd` mais pas `absurd`.

`^`
: Correspond au début d'une ligne (en mode multi lignes, qui est le mode par défaut)

`$`
: Correspond à la fin d'une ligne (en mode multi lignes, qui est le mode par défaut)

`\K`
: Réinitialise l'endroit de début de la sélection à sa position dans le motif. Certains moteurs de regexp (mais pas celui de calibre) n'autorisent pas les assertions arrière de longueur variable, et notamment avec des quantificateurs. Lorsque l'on peut utiliser K avec ces moteurs-là, il permet aussi de s'affranchir de cette limite en écrivant l'équivalent d'une assertions arrière positive de longueur variable.

Groupes

(expression)
Groupe de capture, qui mémorise la sélection et qui peut être rappelée par la suite dans les motifs « rechercher » ou « remplacer » par n, où `n` est le numéro d'ordre du groupe de capture (en commençant à 1 dans l'ordre de lecture)

(?:expression)
Groupe qui ne mémorise pas la sélection

(?>expression)
Groupe atomique : dès que l'expression est satisfaite, le moteur de regexp passe à la suite, et si le reste du motif échoue, il ne fera pas marche arrière pour essayer d'autres combinaisons avec l'expression. Les groupes atomiques ne capturent pas.

(?|expression)
Groupe de réinitialisation de branche : les branches des alternatives inclues dans l'expression partagent les mêmes numéros de groupe

(?<name>expression)
Groupe nommé « name ». La sélection peut être rappelée par la suite dans le motif *rechercher* par `(?P=name)` et dans le *remplacer* par `\g<name>`. Deux groupes différents peuvent utiliser le même nom.

Les assertiions

Assertion	Signification
`?=`	Assertion avant positive (à placer après la sélection)
`?!`	Assertion avant négative (à placer après la sélection)
`?<=`	Assertion arrière positive (à placer avant la sélection)
`?<!`	Assertion arrière négative (à placer avant la sélection)

Les assertions avant et arrière ne consomment pas de caractère, elles sont de longueur nulle et ne capturent pas. Ce sont des groupes atomiques : dès que l'assertion est satisfaite, le moteur de regexp passe à la suite, et si le reste du motif échoue, il ne fera pas marche arrière à l'intérieur de l'assertion pour essayer d'autres combinaisons.

Lorsque l'on cherche plusieurs correspondances dans une chaîne, à la position de départ de chaque tentative de correspondance, une assertion arrière peut inspecter les caractères situés avant la position actuelle. Par conséquent, sur la chaîne 123, le motif `(?<=\d)\d` (un chiffre précédé d'un chiffre) devrait, en théorie, sélectionner 2 et 3. En revanche, `\d\K\d` ne peut sélectionner que 2, car la position de départ après la première sélection est immédiatement avant 3, et il n'y a plus assez de chiffres pour une seconde correspondance. De même, `\d(\d)` ne capture que 2. Dans la pratique du moteur de regexp de calibre, l'assertion arrière positive se comporte de la même façon, et ne sélectionne que 2, contrairement à ce que prévoit la théorie.

On peut placer des groupes à l'intérieur des assertions, mais la capture est rarement utile. Néanmoins, si elle est utile, il faudra être très prudent dans l'utilisation d'un quantificateur dans une assertion arrière : la gourmandise associée à l'absence de marche arrière peut donner une capture surprenante. Pour cette raison, utilisez `\K` plutôt qu'une assertion arrière positive quand vous avez un quantificateur (ou pire, plusieurs) dans un groupe capturant de l'assertion arrière positive.

Exemple d'assertion avant négative :

```
(?![^<>{}]*[>}])
```

Placée à la fin du motif, cette assertion empêche de sélectionner à l'intérieur d'une balise ou d'un style embarqué dans le fichier.

Lorsque c'est possible, il est toujours préférable d'»ancrer» les assertions, pour diminuer le nombre d'étapes nécessaires à l'obtention du résultat.

La récursion

Représentation	Signification
`(?R)`	Récursion du motif entier
`(?1)`	Récursion du seul motif du groupe de capture numéroté, ici le groupe 1

La récursion consiste à s'appeler soi-même. C'est utile pour des recherches balancées, par exemple des chaînes entre apostrophes, qui peuvent contenir des chaînes entre apostrophes enchâssées. Ainsi, si au cours du traitement d'une chaîne entre apostrophes, on rencontre le début d'une nouvelle chaîne entre apostrophes, et bien on sait faire, et on s'appelle soi-même. On a alors un motif comme :

```
start-pattern(?>atomic sub-pattern|(?R))*end-pattern
```

Pour sélectionner une chaîne entre apostrophes sans s'arrêter aux chaînes enchâssées :

```
“((?>[^“”]+|(?R))*[^“”]+)”
```

C'est également ce modèle qu'il faut suivre pour modifier les paires de balises qui peuvent s'enchâsser, telles que les balises `<div>`.

Les caractères spéciaux

Représentation	Caractère
`\t`	tabulation
`\n`	saut de ligne
`\x20`	espace (sécable)
`\xa0`	espace insécable

Les métacaractères

Les métacaractères sont ceux qui ont une signification spéciale pour le moteur regexp. Parmi ceux-ci, douze doivent être précédés d'un caractère d'échappement, l'antislash (`\`), pour perdre leur signification spéciale et redevenir un simple caractère :

```
^ . [ ] $ ( ) * + ? | \
```

Sept autres métacaractères n'ont pas besoin d'être précédés de l'antislash (mais peuvent l'être sans autre conséquence) :

```
{ } ! < > = :
```

Les caractères spéciaux perdent leur statut s'ils sont utilisés à l'intérieur d'une classe (entre les crochets `[]`). Le crochet fermant et le tiret ont un statut spécial dans une classe. En dehors de la classe, le tiret est un simple littéral, le crochet reste un métacaractère.

Le slash (/) et le croisillon (#) [carré au Québec] ne sont pas des métacaractères, a fortiori ils n'ont pas besoin d'être échappés.

Dans certains outils, comme regex101.com, avec le moteur Python, les doubles quotes ont le statut spécial de séparateur, et doivent être échappés. Ce n'est pas le cas dans l'éditeur de calibre.

Modes

(?s)
: Amène le point (`.`) à correspondre également à des caractères de nouvelle ligne

(?m)
: Rend les ancres `^` et `$` à correspondre au début et à la fin de la ligne plutôt que le début et la fin de la chaîne entière.

10.4.14 Crédits

Remerciements pour l'aide avec les astuces, les correction et analogues :

- ldolse
- kovidgoyal
- chaley
- dwanthny
- kacir
- Starson17
- Orpheu

Pour plus d'informations sur les regexps, voir The Python User Manual[90]. La bibliothèque d'expressions régulières actuellement utilisée par calibre est : regex[91] qui supporte plusieurs améliorations utiles par rapport à la bibliothèque standard Python.

10.5 Écrire vos propres extensions pour étendre les fonctionnalités de calibre

calibre a un design hautement modulaire. Presque toutes les fonctionnalités dans calibre viennent sous la forme d'extensions. Les extensions sont utilisées pour la conversion, pour télécharger des actualités (bien que celles-ci s'appellent des recettes), pour divers composants de l'interface utilisateur, pour connecter différents périphériques, pour traiter les fichiers lorsque vous les ajouter à calibre. Vous pouvez obtenir une liste complète de toutes les extensions intégrées dans calibre en allant dans *Préférences → Avancé → Extensions*.

Ici, nous vous apprendrons comment créer vos propres extensions pour ajouter de nouvelles fonctionnalités à calibre.

Contenu

- *Anatomie d'une extension calibre* (page 237)
- *Une extension d'Interface Utilisateur* (page 238)
  - *__init__.py* (page 239)

[90] https://docs.python.org/library/re.html
[91] https://bitbucket.org/mrabarnett/mrab-regex/src/hg/

  - *ui.py* (page 241)
  - *main.py* (page 242)
  - *Obtention des ressources à partir du fichier ZIP de l'extension* (page 244)
  - *Activation de la configuration utilisateur pour votre extension* (page 245)
- *Extensions d'édition de livre* (page 246)
  - *main.py* (page 247)
- *Ajout des traductions à votre extension* (page 250)
- *L'API extension* (page 250)
- *Dépannage des extensions* (page 251)
- *Plus d'exemples d'extensions* (page 251)
- *Partager vos extensions avec les autres* (page 251)

Note : Ceci s'applique uniquement aux versions de calibre >=0.8.60

10.5.1 Anatomie d'une extension calibre

Une extension calibre est très simple, c'est juste un fichier ZIP qui contient un peu de code Python et toutes les autres ressources comme les fichiers images nécessaires à l'extension. Sans plus tarder, voyons un exemple basique.

Supposons que vous avez une installation de calibre que vous utilisez pour l'auto publication de divers e-documents aux formats EPUB et MOBI. Vous aimeriez que tous les fichiers générés par calibre aient leur éditeur paramétré à *Salut le monde*, voici comment faire. Créez un fichier nommé `__init__.py` (c'est un nom spécial et doit toujours être utilisé pour le fichier principal de votre extension) entrez-y le code Python suivant :

```
from calibre.customize import FileTypePlugin


class HelloWorld(FileTypePlugin):

    name                = 'Hello World Plugin' # Name of the plugin
    description         = 'Set the publisher to Hello World for all new conversions'
    supported_platforms = ['windows', 'osx', 'linux'] # Platforms this plugin will␣
↪run on
    author              = 'Acme Inc.' # The author of this plugin
    version             = (1, 0, 0)   # The version number of this plugin
    file_types          = {'epub', 'mobi'} # The file types that this plugin will be␣
↪applied to
    on_postprocess      = True # Run this plugin after conversion is complete
    minimum_calibre_version = (0, 7, 53)

    def run(self, path_to_ebook):
        from calibre.ebooks.metadata.meta import get_metadata, set_metadata
        with open(path_to_ebook, 'r+b') as file:
            ext  = os.path.splitext(path_to_ebook)[-1][1:].lower()
            mi = get_metadata(file, ext)
            mi.publisher = 'Hello World'
```

(suite sur la page suivante)

(suite de la page précédente)

```
            set_metadata(file, mi, ext)
        return path_to_ebook
```

C'est tout. Pour ajouter ce code à calibre en tant qu'extension, exécuter simplement ce qui suit dans le dossier dans lequel vous avez créé `__init__.py` :

```
calibre-customize -b .
```

Note : Sous macOS, les outils en de ligne de commande sont à l'intérieur de la suite logicielle calibre, par exemple, si vous installez calibre dans `/Applications` les outils de ligne de commande sont dans `/Applications/calibre.app/Contents/MacOS/`.

Vous pouvez télécharger le plugin Hello World depuis helloworld_plugin.zip[92].

A chaque fois que vous utilisez calibre pour convertir un livre, la méthode de l'extension `run()` sera appelée et les livres convertis auront leur éditeur établi à *Salut le monde*. C'est une extension banale, passons à un exemple plus complexe qui habituellement ajoute un composant à l'interface utilisateur.

10.5.2 Une extension d'Interface Utilisateur

Cette extension sera répartie sur quelques fichiers (pour maintenir le code propre). Il vous montrera comment obtenir des ressources (des fichiers d'images ou de données) à partir du fichier extension ZIP, permettra à des utilisateurs de configurer votre extension, comment créer des éléments dans l'Interface Utilisateur de calibre et comment accéder et questionner la base de données de livres de calibre.

Vous pouvez télécharger ce plugin à partir de interface_demo_plugin.zip[93].

La première chose à noter est que le fichier ZIP contient beaucoup plus de fichiers, expliqués ci-dessous, prêtez particulièrement attention à `plugin-import-name-interface_demo.txt`.

plugin-import-name-interface_demo.txt
: Un fichier texte vide utilisé pour activer la magie de l'extension multi fichiers. Ce fichier doit être présent dans toutes les extensions qui utilisent plus d'un fichier .py. Il devra être vide et son nom de fichier doit être de la forme : plugin-import-name-**un_nom**.txt. La présence de ce fichier vous permet d'importer du code à partir de fichiers .py présents à l'intérieur du fichier ZIP, utilisant une déclaration comme :

  ```
  from calibre_plugins.some_name.some_module import some_object
  ```

  Le préfixe `calibre_plugins` doit toujours être présent. `some_name` provient du nom de fichier du fichier texte vide. `some_module` se réfère au fichier `some_module.py` dans le fichier ZIP. Notez que cette importation est tout aussi puissante que les importations Python régulières. Vous pouvez créer des paquets et des sous paquets de modules .py à l'intérieur du fichier ZIP, tout comme vous le feriez normalement (en définissant __init__.py dans chaque sous dossier), et tout devrait « fonctionner correctement ».

  Le nom que vous utilisez pour `un_nom` introduit un espace de nom global partagé par toutes les extensions. **Aussi rendez le aussi unique que possible**. Mais rappelez-vous qu'il doit être un identifiant Python valide(uniquement des lettres, des nombres et le tiret bas).

__init__.py
: Comme précédemment, le fichier qui définit la classe de l'extension

[92] https://calibre-ebook.com/downloads/helloworld_plugin.zip
[93] https://calibre-ebook.com/downloads/interface_demo_plugin.zip

main.py
Ce fichier contient le code actuel qui fait quelque chose d'utile

ui.py
Ce fichier définit la partie interface de l'extension

images/icon.png
L'icône pour ce plugin

about.txt
Un fichier texte contenant des informations sur le plugin

traductions
Un répertoire contenant des fichiers .mo avec les traductions de l'interface utilisateur de votre extension dans différentes langues. Voir ci-dessous pour les détails.

Maintenant regardons le code

__init__.py

Tout d'abord, l'obligatoirement `__init__.py` pour définir les métadonnées de l'extension :

```
from calibre.customize import InterfaceActionBase


class InterfacePluginDemo(InterfaceActionBase):
    '''
    This class is a simple wrapper that provides information about the actual
    plugin class. The actual interface plugin class is called InterfacePlugin
    and is defined in the ui.py file, as specified in the actual_plugin field
    below.

    The reason for having two classes is that it allows the command line
    calibre utilities to run without needing to load the GUI libraries.
    '''
    name                = 'Interface Plugin Demo'
    description         = 'An advanced plugin demo'
    supported_platforms = ['windows', 'osx', 'linux']
    author              = 'Kovid Goyal'
    version             = (1, 0, 0)
    minimum_calibre_version = (0, 7, 53)

    #: This field defines the GUI plugin class that contains all the code
    #: that actually does something. Its format is module_path:class_name
    #: The specified class must be defined in the specified module.
    actual_plugin       = 'calibre_plugins.interface_demo.ui:InterfacePlugin'

    def is_customizable(self):
        '''
        This method must return True to enable customization via
        Preferences->Plugins
        '''
        return True

    def config_widget(self):
        '''
        Implement this method and :meth:`save_settings` in your plugin to
        use a custom configuration dialog.
```

(suite sur la page suivante)

(suite de la page précédente)

```
        This method, if implemented, must return a QWidget. The widget can have
        an optional method validate() that takes no arguments and is called
        immediately after the user clicks OK. Changes are applied if and only
        if the method returns True.

        If for some reason you cannot perform the configuration at this time,
        return a tuple of two strings (message, details), these will be
        displayed as a warning dialog to the user and the process will be
        aborted.

        The base class implementation of this method raises NotImplementedError
        so by default no user configuration is possible.
        '''
        # It is important to put this import statement here rather than at the
        # top of the module as importing the config class will also cause the
        # GUI libraries to be loaded, which we do not want when using calibre
        # from the command line
        from calibre_plugins.interface_demo.config import ConfigWidget
        return ConfigWidget()

    def save_settings(self, config_widget):
        '''
        Save the settings specified by the user with config_widget.

        :param config_widget: The widget returned by :meth:`config_widget`.
        '''
        config_widget.save_settings()

        # Apply the changes
        ac = self.actual_plugin_
        if ac is not None:
            ac.apply_settings()
```

La seule caractéristique remarquable est le champ `actual_plugin`. Depuis que calibre a la ligne de commande et les interfaces de GUI, les extensions avec GUI comme celle-ci ne devrait charger aucune bibliothèques de GUI dans __init__.py. Le champ actual_plugin fait ceci pour vous, en disant à calibre que l'extension actuelle doit être trouvée dans un autre fichier à l'intérieur de votre archive ZIP, qui seront seulement chargées dans un contexte GUI.

Rappelez-vous que pour que cela fonctionne, vous devez avoir un fichier plugin-import-name-un_nom.txt dans le fichier ZIP de votre extension, comme discuté plus haut.

Il y a également quelques méthodes pour permettre la configuration utilisateur de l'extension. Celles-ci sont discutées ci-dessous.

ui.py

Maintenant regardons à ui.py qui définit le GUI habituel de l'extension. Le code source est fortement commenté et devrait être explicite :

```
from calibre.gui2.actions import InterfaceAction
from calibre_plugins.interface_demo.main import DemoDialog


class InterfacePlugin(InterfaceAction):

    name = 'Interface Plugin Demo'

    # Declare the main action associated with this plugin
    # The keyboard shortcut can be None if you dont want to use a keyboard
    # shortcut. Remember that currently calibre has no central management for
    # keyboard shortcuts, so try to use an unusual/unused shortcut.
    action_spec = ('Interface Plugin Demo', None,
            'Run the Interface Plugin Demo', 'Ctrl+Shift+F1')

    def genesis(self):
        # This method is called once per plugin, do initial setup here

        # Set the icon for this interface action
        # The get_icons function is a builtin function defined for all your
        # plugin code. It loads icons from the plugin zip file. It returns
        # QIcon objects, if you want the actual data, use the analogous
        # get_resources builtin function.
        #
        # Note that if you are loading more than one icon, for performance, you
        # should pass a list of names to get_icons. In this case, get_icons
        # will return a dictionary mapping names to QIcons. Names that
        # are not found in the zip file will result in null QIcons.
        icon = get_icons('images/icon.png', 'Interface Demo Plugin')

        # The qaction is automatically created from the action_spec defined
        # above
        self.qaction.setIcon(icon)
        self.qaction.triggered.connect(self.show_dialog)

    def show_dialog(self):
        # The base plugin object defined in __init__.py
        base_plugin_object = self.interface_action_base_plugin
        # Show the config dialog
        # The config dialog can also be shown from within
        # Preferences->Plugins, which is why the do_user_config
        # method is defined on the base plugin class
        do_user_config = base_plugin_object.do_user_config

        # self.gui is the main calibre GUI. It acts as the gateway to access
        # all the elements of the calibre user interface, it should also be the
        # parent of the dialog
        d = DemoDialog(self.gui, self.qaction.icon(), do_user_config)
        d.show()

    def apply_settings(self):
        from calibre_plugins.interface_demo.config import prefs
        # In an actual non trivial plugin, you would probably need to
```

(suite sur la page suivante)

(suite de la page précédente)

```
        # do something based on the settings in prefs
        prefs
```

main.py

La logique actuelle pour mettre en application le dialogue de démo de l'interface de l'extension.

```
from qt.core import QDialog, QLabel, QMessageBox, QPushButton, QVBoxLayout


class DemoDialog(QDialog):

    def __init__(self, gui, icon, do_user_config):
        QDialog.__init__(self, gui)
        self.gui = gui
        self.do_user_config = do_user_config

        # The current database shown in the GUI
        # db is an instance of the class LibraryDatabase from db/legacy.py
        # This class has many, many methods that allow you to do a lot of
        # things. For most purposes you should use db.new_api, which has
        # a much nicer interface from db/cache.py
        self.db = gui.current_db

        self.l = QVBoxLayout()
        self.setLayout(self.l)

        self.label = QLabel(prefs['hello_world_msg'])
        self.l.addWidget(self.label)

        self.setWindowTitle('Interface Plugin Demo')
        self.setWindowIcon(icon)

        self.about_button = QPushButton('About', self)
        self.about_button.clicked.connect(self.about)
        self.l.addWidget(self.about_button)

        self.marked_button = QPushButton(
            'Show books with only one format in the calibre GUI', self)
        self.marked_button.clicked.connect(self.marked)
        self.l.addWidget(self.marked_button)

        self.view_button = QPushButton(
            'View the most recently added book', self)
        self.view_button.clicked.connect(self.view)
        self.l.addWidget(self.view_button)

        self.update_metadata_button = QPushButton(
            'Update metadata in a book\'s files', self)
        self.update_metadata_button.clicked.connect(self.update_metadata)
        self.l.addWidget(self.update_metadata_button)

        self.conf_button = QPushButton(
                'Configure this plugin', self)
        self.conf_button.clicked.connect(self.config)
```

(suite sur la page suivante)

(suite de la page précédente)

```
        self.l.addWidget(self.conf_button)

        self.resize(self.sizeHint())

    def about(self):
        # Get the about text from a file inside the plugin zip file
        # The get_resources function is a builtin function defined for all your
        # plugin code. It loads files from the plugin zip file. It returns
        # the bytes from the specified file.
        #
        # Note that if you are loading more than one file, for performance, you
        # should pass a list of names to get_resources. In this case,
        # get_resources will return a dictionary mapping names to bytes. Names that
        # are not found in the zip file will not be in the returned dictionary.
        text = get_resources('about.txt')
        QMessageBox.about(self, 'About the Interface Plugin Demo',
                text.decode('utf-8'))

    def marked(self):
        ''' Show books with only one format '''
        db = self.db.new_api
        matched_ids = {book_id for book_id in db.all_book_ids() if len(db.
↪formats(book_id)) == 1}
        # Mark the records with the matching ids
        # new_api does not know anything about marked books, so we use the full
        # db object
        self.db.set_marked_ids(matched_ids)

        # Tell the GUI to search for all marked records
        self.gui.search.setEditText('marked:true')
        self.gui.search.do_search()

    def view(self):
        ''' View the most recently added book '''
        most_recent = most_recent_id = None
        db = self.db.new_api
        for book_id, timestamp in db.all_field_for('timestamp', db.all_book_ids()).
↪items():
            if most_recent is None or timestamp > most_recent:
                most_recent = timestamp
                most_recent_id = book_id

        if most_recent_id is not None:
            # Get a reference to the View plugin
            view_plugin = self.gui.iactions['View']
            # Ask the view plugin to launch the viewer for row_number
            view_plugin._view_calibre_books([most_recent_id])

    def update_metadata(self):
        '''
        Set the metadata in the files in the selected book's record to
        match the current metadata in the database.
        '''
        from calibre.ebooks.metadata.meta import set_metadata
        from calibre.gui2 import error_dialog, info_dialog

        # Get currently selected books
```

(suite sur la page suivante)

(suite de la page précédente)

```
        rows = self.gui.library_view.selectionModel().selectedRows()
        if not rows or len(rows) == 0:
            return error_dialog(self.gui, 'Cannot update metadata',
                                'No books selected', show=True)
        # Map the rows to book ids
        ids = list(map(self.gui.library_view.model().id, rows))
        db = self.db.new_api
        for book_id in ids:
            # Get the current metadata for this book from the db
            mi = db.get_metadata(book_id, get_cover=True, cover_as_data=True)
            fmts = db.formats(book_id)
            if not fmts:
                continue
            for fmt in fmts:
                fmt = fmt.lower()
                # Get a python file object for the format. This will be either
                # an in memory file or a temporary on disk file
                ffile = db.format(book_id, fmt, as_file=True)
                ffile.seek(0)
                # Set metadata in the format
                set_metadata(ffile, mi, fmt)
                ffile.seek(0)
                # Now replace the file in the calibre library with the updated
                # file. We dont use add_format_with_hooks as the hooks were
                # already run when the file was first added to calibre.
                db.add_format(book_id, fmt, ffile, run_hooks=False)

        info_dialog(self, 'Updated files',
                'Updated the metadata in the files of %d book(s)'%len(ids),
                show=True)

    def config(self):
        self.do_user_config(parent=self)
        # Apply the changes
        self.label.setText(prefs['hello_world_msg'])
```

Obtention des ressources à partir du fichier ZIP de l'extension

Le système de chargement d'une extension calibre définit quelques fonctions intégrées qui permettent d'obtenir confortablement des fichiers à partir du fichier ZIP de l'extension.

get_resources(name_or_list_of_names)
: Cette fonction devra être appelée avec une liste de chemins vers les fichiers à l'intérieur du fichier ZIP. Par exemple pour accéder au fichier `icon.png` dans le dossier images dans le fichier ZIP, vous devriez utiliser : `images/icon.png`. Toujours utiliser une barre oblique comme séparateur de chemin, même sous Windows. Quand vous analysez un nom seul, la fonction retournera les octets bruts ou Aucun si le nom n'est pas trouvé dans le fichier ZIP. Si vous analysez plus d'un nom alors il renvoie un dictionnaire mappant les noms aux octets. Si un nom n'est pas trouvé, il ne sera pas présent dans le dictionnaire retourné.

get_icons(name_or_list_of_names, plugin_name='')
: Un enveloppeur pour get_resources() qui crée des objets QIcon à partir des octets bruts retournés par get_resources. Si un nom n'est pas trouvé dans le fichier ZIP, le QIcon correspondant sera nul. Afin de supporter les thèmes d'icônes, entrez le nom convivial de votre plugin comme `plugin_name`. Si l'utilisateur utilise un thème d'icônes avec des icônes pour votre plugin, ils seront chargés de préférence.

Activation de la configuration utilisateur pour votre extension

Pour permettre aux utilisateurs de configurer votre extension, vous devez définir trois méthodes dans votre classe de base de l'extension, **is_customizable**, **config_widget** et **save_settings** comme montré ci-dessous :

```
    def is_customizable(self):
        '''
        This method must return True to enable customization via
        Preferences->Plugins
        '''
        return True
```

```
    def config_widget(self):
        '''
        Implement this method and :meth:`save_settings` in your plugin to
        use a custom configuration dialog.

        This method, if implemented, must return a QWidget. The widget can have
        an optional method validate() that takes no arguments and is called
        immediately after the user clicks OK. Changes are applied if and only
        if the method returns True.

        If for some reason you cannot perform the configuration at this time,
        return a tuple of two strings (message, details), these will be
        displayed as a warning dialog to the user and the process will be
        aborted.

        The base class implementation of this method raises NotImplementedError
        so by default no user configuration is possible.
        '''
        # It is important to put this import statement here rather than at the
        # top of the module as importing the config class will also cause the
        # GUI libraries to be loaded, which we do not want when using calibre
        # from the command line
        from calibre_plugins.interface_demo.config import ConfigWidget
        return ConfigWidget()
```

```
    def save_settings(self, config_widget):
        '''
        Save the settings specified by the user with config_widget.

        :param config_widget: The widget returned by :meth:`config_widget`.
        '''
        config_widget.save_settings()

        # Apply the changes
        ac = self.actual_plugin_
        if ac is not None:
            ac.apply_settings()
```

calibre a beaucoup de manières différentes pour stocker les données de configuration (un héritage de sa longue histoire). La manière recommandée est d'utiliser la classe **JSONConfig**, qui stocke votre information de configuration dans un fichier .json.

Le code pour gérer le données de configuration dans l'extension démo est dans config.py :

```
from qt.core import QHBoxLayout, QLabel, QLineEdit, QWidget

# This is where all preferences for this plugin will be stored
# Remember that this name (i.e. plugins/interface_demo) is also
# in a global namespace, so make it as unique as possible.
# You should always prefix your config file name with plugins/,
# so as to ensure you dont accidentally clobber a calibre config file
prefs = JSONConfig('plugins/interface_demo')

# Set defaults
prefs.defaults['hello_world_msg'] = 'Hello, World!'


class ConfigWidget(QWidget):

    def __init__(self):
        QWidget.__init__(self)
        self.l = QHBoxLayout()
        self.setLayout(self.l)

        self.label = QLabel('Hello world &message:')
        self.l.addWidget(self.label)

        self.msg = QLineEdit(self)
        self.msg.setText(prefs['hello_world_msg'])
        self.l.addWidget(self.msg)
        self.label.setBuddy(self.msg)

    def save_settings(self):
        prefs['hello_world_msg'] = self.msg.text()
```

L'objet `prefs` est maintenant disponible dans tout le code de l'extension par un simple :

```
from calibre_plugins.interface_demo.config import prefs
```

Vous pouvez voir l'objet `prefs` en cours d'utilisation dans main.py :

```
        self.do_user_config(parent=self)
        # Apply the changes
        self.label.setText(prefs['hello_world_msg'])
```

10.5.3 Extensions d'édition de livre

Maintenant changeons de direction pour un moment et regardons à la création d'une extension pour ajouter des outils à l'éditeur de livre calibre. L'extension est disponible ici : editor_demo_plugin.zip[94].

La première étape comme pour toutes les extensions est de créer le nom du fichier txt vide d'importation, comme décrit *plus haut* (page 238). Nous appellerons vle fichier `plugin-import-name-editor_plugin_demo.txt`.

Maintenant nous créons le fichier mandataire `__init__.py` qui contient les métadonnées à propos de l'extension – ses nom, auteur, version, etc.

```
from calibre.customize import EditBookToolPlugin
```

(suite sur la page suivante)

[94] https://calibre-ebook.com/downloads/editor_demo_plugin.zip

(suite de la page précédente)

```
class DemoPlugin(EditBookToolPlugin):

    name = 'Edit Book plugin demo'
    version = (1, 0, 0)
    author = 'Kovid Goyal'
    supported_platforms = ['windows', 'osx', 'linux']
    description = 'A demonstration of the plugin interface for the ebook editor'
    minimum_calibre_version = (1, 46, 0)
```

Une simple extension éditeur peut procurer de multiples outils chaque outil correspond à un seul bouton dans la barre d'outils et une entrée dans le menu *Extensions* dans l'éditeur. Celle-ci peut avoir des sous menus dans le cas où l'outil à de multiples actions liées.

Les outils doivent tous être définis dans le fichier `main.py` dans votre extension. Chaque outil est une classe qui hérite de la classe *calibre.gui2.tweak_book.plugin.Tool* (page 384). Regardons attentivement le `main.py` de l'extention démo, le code source est fortement commenté et devrait être explicite. Lisez les documents API de la classe *calibre.gui2.tweak_book.plugin.Tool* (page 384) pour plus de détails.

main.py

Nous verrons ici la définition d'un outil simple qui multipliera toutes les tailles de police dans le livre par un nombre fourni par l'utilisateur ? Cet outil démontre les différents concepts importants dont vous aurez besoin dans le développement de vos propres extensions, aussi vous devriez lire attentivement le code source (fortement commenté).

```
import re

from calibre import force_unicode
from calibre.ebooks.oeb.polish.container import OEB_DOCS, OEB_STYLES, serialize
from calibre.gui2 import error_dialog

# The base class that all tools must inherit from
from calibre.gui2.tweak_book.plugin import Tool
from css_parser.css import CSSRule
from qt.core import QAction, QInputDialog


class DemoTool(Tool):

    #: Set this to a unique name it will be used as a key
    name = 'demo-tool'

    #: If True the user can choose to place this tool in the plugins toolbar
    allowed_in_toolbar = True

    #: If True the user can choose to place this tool in the plugins menu
    allowed_in_menu = True

    def create_action(self, for_toolbar=True):
        # Create an action, this will be added to the plugins toolbar and
        # the plugins menu
        ac = QAction(get_icons('images/icon.png'), 'Magnify fonts', self.gui)  # noqa
        if not for_toolbar:
            # Register a keyboard shortcut for this toolbar action. We only
            # register it for the action created for the menu, not the toolbar,
```

(suite sur la page suivante)

(suite de la page précédente)

```
            # to avoid a double trigger
            self.register_shortcut(ac, 'magnify-fonts-tool', default_keys=(
↪'Ctrl+Shift+Alt+D',))
        ac.triggered.connect(self.ask_user)
        return ac

    def ask_user(self):
        # Ask the user for a factor by which to multiply all font sizes
        factor, ok = QInputDialog.getDouble(
            self.gui, 'Enter a magnification factor', 'Allow font sizes in the book␣
↪will be multiplied by the specified factor',
            value=2, min=0.1, max=4
        )
        if ok:
            # Ensure any in progress editing the user is doing is present in the␣
↪container
            self.boss.commit_all_editors_to_container()
            try:
                self.magnify_fonts(factor)
            except Exception:
                # Something bad happened report the error to the user
                import traceback
                error_dialog(self.gui, _('Failed to magnify fonts'), _(
                    'Failed to magnify fonts, click "Show details" for more info'),
                    det_msg=traceback.format_exc(), show=True)
                # Revert to the saved restore point
                self.boss.revert_requested(self.boss.global_undo.previous_container)
            else:
                # Show the user what changes we have made, allowing her to
                # revert them if necessary
                self.boss.show_current_diff()
                # Update the editor UI to take into account all the changes we
                # have made
                self.boss.apply_container_update_to_gui()

    def magnify_fonts(self, factor):
        # Magnify all font sizes defined in the book by the specified factor
        # First we create a restore point so that the user can undo all changes
        # we make.
        self.boss.add_savepoint('Before: Magnify fonts')

        container = self.current_container  # The book being edited as a container␣
↪object

        # Iterate over all style declarations in the book, this means css
        # stylesheets, <style> tags and style="" attributes
        for name, media_type in container.mime_map.items():
            if media_type in OEB_STYLES:
                # A stylesheet. Parsed stylesheets are css_parser CSSStylesheet
                # objects.
                self.magnify_stylesheet(container.parsed(name), factor)
                container.dirty(name)  # Tell the container that we have changed the␣
↪stylesheet
            elif media_type in OEB_DOCS:
                # A HTML file. Parsed HTML files are lxml elements

                for style_tag in container.parsed(name).xpath('//*[local-name="style"]
```

(suite sur la page suivante)

(suite de la page précédente)

```
↪'):
                    if style_tag.text and style_tag.get('type', None) in {None, 'text/
↪css'}:
                        # We have an inline CSS <style> tag, parse it into a
                        # stylesheet object
                        sheet = container.parse_css(style_tag.text)
                        self.magnify_stylesheet(sheet, factor)
                        style_tag.text = serialize(sheet, 'text/css', pretty_
↪print=True)
                        container.dirty(name)  # Tell the container that we have␣
↪changed the stylesheet
                for elem in container.parsed(name).xpath('//*[@style]'):
                    # Process inline style attributes
                    block = container.parse_css(elem.get('style'), is_
↪declaration=True)
                    self.magnify_declaration(block, factor)
                    elem.set('style', force_unicode(block.getCssText(separator=' '),
↪'utf-8'))

    def magnify_stylesheet(self, sheet, factor):
        # Magnify all fonts in the specified stylesheet by the specified
        # factor.
        for rule in sheet.cssRules.rulesOfType(CSSRule.STYLE_RULE):
            self.magnify_declaration(rule.style, factor)

    def magnify_declaration(self, style, factor):
        # Magnify all fonts in the specified style declaration by the specified
        # factor
        val = style.getPropertyValue('font-size')
        if not val:
            return
        # see if the font-size contains a number
        num = re.search(r'[0-9.]+', val)
        if num is not None:
            num = num.group()
            val = val.replace(num, '%f' % (float(num) * factor))
            style.setProperty('font-size', val)
        # We should also be dealing with the font shorthand property and
        # font sizes specified as non numbers, but those are left as exercises
        # for the reader
```

Décomposons `main.py`. Nous voyons qu'il définit une seul outil, nommé *Magnify fonts*. Cet outil demandera à l'utilisateur un nombre et multipliera toutes les tailles de police dans le livre par ce nombre.

La première chose importante est le nom d'outil que vous devez établir à une certaine chaîne relativement unique car elle sera employée comme clé pour cet outil.

L'entrée importante suivante est la *calibre.gui2.tweak_book.plugin.Tool.create_action()* (page 385). Cette méthode crée les objects QAction qui apparaissent dans la barre d'outils et le menu extensions. Elle assigne aussi, optionnellement, un raccourci clavier que l'utilisateur peut personnaliser. Le signal de déclenchement de QAction est relié à la méthode ask_user () qui demande à l'utilisateur le multiplicateur de taille de la police, et puis exécute le code de grossissement.

Le code de grossissement est bien commenté assez simple. Les choses les plus importantes à noter sont que vous obtenez une référence à la fenêtre de l'éditeur comme `self.gui` et l'éditeur *Boss* comme `self.boss`. Le *Boss* est l'objet qui contrôle l'interface utilisateur de l'éditeur. Il a beaucoup de méthodes utiles, elles sont documentées dans la classe *calibre.gui2.tweak_book.boss.Boss* (page 386).

Finalement, il y a `self.current_container` qui est ue référence au livre en cours d'édition comme un objet *`calibre.ebooks.oeb.polish.container.Container`* (page 377). Ceci représente le livre comme une collection de ses fichiers constituants HTML/CSS/image et a des méthodes commodes pour faire beaucoup de choses utiles. L'objet container et diverses fonctions de service utiles qui peuvent être réutilisées en votre code d'extension sont documentées dans *Documentation API pour l'outil d'édition de livre numérique* (page 376).

10.5.4 Ajout des traductions à votre extension

Vous pouvez avoir toutes les chaînes de l'interface utilisateur dans votre extension traduites et affichées dans n'importe quelle langue paramétrée pour l'interface utilisateur principale de calibre.

La première étape est de faire le tour du code source de votre extension et marquer toutes les chaînes visibles par l'utilisateur comme traduisible, en les entourant dans _(). Par exemple :

```
action_spec = (_('My plugin'), None, _('My plugin is cool'), None)
```

Utilisez alors un programme pour générer les fichiers .po à partir du code source de votre extension. Il devrait y avoir un fichier .po pour chaque langue dans laquelle vous voulez une traduction. Par exemple de.po pour l'allemand, fr.po pour le français et ainsi de suite. Vous pouvez utiliser le programme Poedit[95] pour cela.

Envoyez ces fichiers .po à vos traducteurs. Une fois que vous les obtenez en retour, compilez les dans des fichiers .mo. Vous pouvez encore utilisez Poedit pour cela, ou juste faire :

```
calibre-debug -c "from calibre.translations.msgfmt import main; main()" filename.po
```

Déposez les fichiers .mo dans le répertoire `translations` dans votre extension.

La dernière étape est de simplement appeler la fonction *load_translations()* en haut de vos fichiers .py d'extension. Pour des raisons de performance, vous devriez seulement appeler cette fonction dans les fichiers .py qui ont réellement des chaînes traduisibles. Aussi dans une Interface Utilisateur typique vous devriez l'appeler en haut de `ui.py` mais pas de `__init__.py`.

Vous pouvez tester les traductions de vos extensions en changeant la langue de l'interface utilisateur dans calibre dans *Préférences → Interface → Apparence* ou en exécutant calibre avec la variable d'environnement `CALIBRE_OVERRIDE_LANG`. Par exemple :

```
CALIBRE_OVERRIDE_LANG=de
```

Remplac``de`` avec le code de langue de la langue que vous voulez tester.

Pour les traductions avec des pluriels, utilisez la fonction `ngettext()` au lieu de `_()`. Par exemple :

```
ngettext('Delete a book', 'Delete {} books', num_books).format(num_books)
```

10.5.5 L'API extension

Comme vous avez pu le constatez plus haut, une extension dans calibre est une classe. Il a différentes classes pour les différents types d'extension dans calibre. Les détails sur chaque classe, incluant la classe de base de toutes les extensions peuvent être trouvés dans *Documentation API pour les extensions* (page 268).

Votre extension va presque certainement employer le code de calibre. Pour apprendre comment trouver les diverses parties de fonctionnalités dans le code de base de calibre, lisez la section sur le *Disposition du code* (page 358) calibre.

[95] https://poedit.net/

10.5.6 Dépannage des extensions

Tout d'abord, l'étape la plus importante est d'exécuter calibre en mode dépannage. Vous pouvez faire cela depuis la ligne de commande avec :

```
calibre-debug -g
```

Ou, dans calibre, en faisant un clic droit sur le bouton *Préférences* ou en utilisant le raccourci clavier `Ctrl+Shift+R`.

Lors de l'exécution en ligne de commande, la sortie de dépannage sera affichée dans la console, lors de l'exécution depuis calibre la sortie ira dans un fichier txt.

Vous pouvez insérer des déclarations n'importe où dans votre code d'extension, elles seront reprises dans le mode dépannage. Rappelez-vous, ceci est du Python, vous ne devriez n'avoir vraiment besoin de rien de plus que les déclarations pour corriger ;) J'ai développé entièrement calibre en utilisant juste cette technique d'élimination des imperfections.

Vous pouvez rapidement tester les changements de votre extension en utilisant la ligne de commande suivante :

```
calibre-debug -s; calibre-customize -b /path/to/your/plugin/folder; calibre
```

Ceci arrêtera un calibre en cours d'exécution, attendez que l'arrêt soit complet, mettez alors à jour votre extension à jour dans calibre et relancez calibre.

10.5.7 Plus d'exemples d'extensions

Vous pouvez trouver une liste de nombreuses extensions calibre sophistiquées **`ici<https ://www.mobileread.com/forums/showthread.php ?t=118764>`_**.

10.5.8 Partager vos extensions avec les autres

Si vous aimeriez partager les extensions que vous avez créées avec d'autres utilisateurs de calibre, postez votre extension dans un nouveau fil de discussion sur le forum des extensions calibre[96].

10.6 Compositions mathématiques dans des livres numériques

La Visionneuse de livre numérique calibre a la capacité d'afficher les mathématiques intégrées dans les livres numériques (fichiers EPUB et HTML). Vous pouvez composer directement les mathématiques avec TeX, MathML ou AsciiMath. La visionneuse de calibre utilise l'excellente librairie MathJax[97] pour faire cela. Ceci est un bref tutoriel sur la création de livres numériques contenant des mathématiques qui fonctionnent bien avec la Visionneuse de livre numérique calibre.

[96] https://www.mobileread.com/forums/forumdisplay.php?f=237
[97] https://www.mathjax.org

10.6.1 Un simple fichier HTML avec des mathématiques

Vous pouvez écrire des mathématiques en ligne à l'intérieur d'un fichier HTML simple et la Visionneuse de livre numérique de calibre rendra des mathématiques correctement composées. Dans l'exemple ci-dessous, nous employons la notation de Tex pour les mathématiques. Vous verrez que vous pouvez employer des commandes normales de Tex, avec la petite mise en garde que les esperluettes et les signes moins grand que et plus grand que doivent être écrits respectivement comme & < et > .

La première étape est de dire à calibre que ceci contiendra des mathématiques. Vous faites cela en ajoutant l'extrait de code suivant à la section <head> du HTML :

```
<script type="text/x-mathjax-config"></script>
```

C'est tout, maintenant vous pouvez entrer des mathématiques comme vous le feriez dans un fichier .tex. Par exemple, voici les équations de Lorentz :

```
<h2>The Lorenz Equations</h2>

<p>
\begin{align}
\dot{x} &amp;= \sigma(y-x) \\
\dot{y} &amp;= \rho x - y - xz \\
\dot{z} &amp;= -\beta z + xy
\end{align}
</p>
```

Cet extrait ressemble à la capture d'écran suivante dans la Visionneuse de livre numérique calibre.

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

Fig. 1 : *Les Équations de Lorenz*

Le fichier HTML complet, avec plus d'équations et de mathématiques intégrées est reproduit ci-dessous. Vous pouvez convertir ce fichier HTML en EPUB dans calibre pour obtenir un livre numérique que vous pouvez distribuer facilement à d'autres personnes.

```
<!DOCTYPE html>
<html>
<!-- Copyright (c) 2012 Design Science, Inc. -->
<head>
<title>Math Test Page</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />

<!-- This script tag is needed to make calibre's ebook-viewer recpgnize that this␣
↪file needs math typesetting -->
<script type="text/x-mathjax-config">
    // This line adds numbers to all equations automatically, unless explicitly␣
↪suppressed.
    MathJax.tex = {tags: 'all'};
```

(suite sur la page suivante)

(suite de la page précédente)

```
</script>

<style>
h1 {text-align:center}
h2 {
  font-weight: bold;
  background-color: #DDDDDD;
  padding: .2em .5em;
  margin-top: 1.5em;
  border-top: 3px solid #666666;
  border-bottom: 2px solid #999999;
}
</style>
</head>
<body>

<h1>Sample Equations</h1>

<h2>The Lorenz Equations</h2>

<p>
\begin{align}
\dot{x} &amp; = \sigma(y-x) \label{lorenz}\\
\dot{y} &amp; = \rho x - y - xz \\
\dot{z} &amp; = -\beta z + xy
\end{align}
</p>

<h2>The Cauchy-Schwarz Inequality</h2>

<p>\[
\left( \sum_{k=1}^n a_k b_k \right)^{\!\!2} \leq
 \left( \sum_{k=1}^n a_k^2 \right) \left( \sum_{k=1}^n b_k^2 \right)
\]</p>

<h2>A Cross Product Formula</h2>

<p>\[
  \mathbf{V}_1 \times \mathbf{V}_2 =
   \begin{vmatrix}
    \mathbf{i} &amp; \mathbf{j} &amp; \mathbf{k} \\
    \frac{\partial X}{\partial u} &amp; \frac{\partial Y}{\partial u} &amp; 0 \\
    \frac{\partial X}{\partial v} &amp; \frac{\partial Y}{\partial v} &amp; 0 \\
   \end{vmatrix}
\]</p>

<h2>The probability of getting \(k\) heads when flipping \(n\) coins is:</h2>

<p>\[P(E) = {n \choose k} p^k (1-p)^{ n-k} \]</p>

<h2>An Identity of Ramanujan</h2>

<p>\[
   \frac{1}{(\sqrt{\phi \sqrt{5}}-\phi) e^{\frac25 \pi}} =
     1+\frac{e^{-2\pi}} {1+\frac{e^{-4\pi}} {1+\frac{e^{-6\pi}}
      {1+\frac{e^{-8\pi}} {1+\ldots} } } }
\]</p>
```

(suite sur la page suivante)

(suite de la page précédente)

```
<h2>A Rogers-Ramanujan Identity</h2>

<p>\[
  1 +  \frac{q^2}{(1-q)}+\frac{q^6}{(1-q)(1-q^2)}+\cdots =
    \prod_{j=0}^{\infty}\frac{1}{(1-q^{5j+2})(1-q^{5j+3})},
     \quad\quad \text{for $|q|&lt;1$}.
\]</p>

<h2>Maxwell's Equations</h2>

<p>
\begin{align}
  \nabla \times \vec{\mathbf{B}} -\, \frac1c\, \frac{\partial\vec{\mathbf{E}}}{\
↪partial t} &amp; = \frac{4\pi}{c}\vec{\mathbf{j}} \\
  \nabla \cdot \vec{\mathbf{E}} &amp; = 4 \pi \rho \\
  \nabla \times \vec{\mathbf{E}}\, +\, \frac1c\, \frac{\partial\vec{\mathbf{B}}}{\
↪partial t} &amp; = \vec{\mathbf{0}} \\
  \nabla \cdot \vec{\mathbf{B}} &amp; = 0
\end{align}
</p>

<h2>In-line Mathematics</h2>

<p>While display equations look good for a page of samples, the
ability to mix math and text in a paragraph is also important.  This
expression \(\sqrt{3x-1}+(1+x)^2\) is an example of an inline equation.  As
you see, equations can be used this way as well, without unduly
disturbing the spacing between lines.</p>

<h2>References to equations</h2>

<p>Here is a reference to the Lorenz Equations (\ref{lorenz}). Clicking on the␣
↪equation number will take you back to the equation.</p>

</body>
</html>
```

10.6.2 Plus d'information

Depuis que la Visionneuse de livre numérique calibre utilise la librairie MathJax pour afficher les mathématiques, le meilleur endroit pour en savoir plus à propos des mathématiques dans les livres numériques et obtenir de l'aide est le site web de MathJax[98].

[98] https://www.mathjax.org

10.7 Création de catalogues AZW3 • EPUB • MOBI

L'option Créer un catalogue de calibre vous permet de créer un catalogue de votre librairie dans une variété de formats. Ce fichier d'aide décrit les options de catalogage lors de la génération d'un catalogue aux formats AZW3, EPUB et MOBI.

- *Sélection des livres à cataloguer* (page 255)
- *Sections incluses* (page 256)
- *Préfixes* (page 256)
- *Livres exclus* (page 257)
- *Exclure les genres* (page 257)
- *Autres options* (page 258)
- *Couvertures de catalogue personnalisées* (page 259)
- *Ressources d'aide supplémentaires* (page 259)

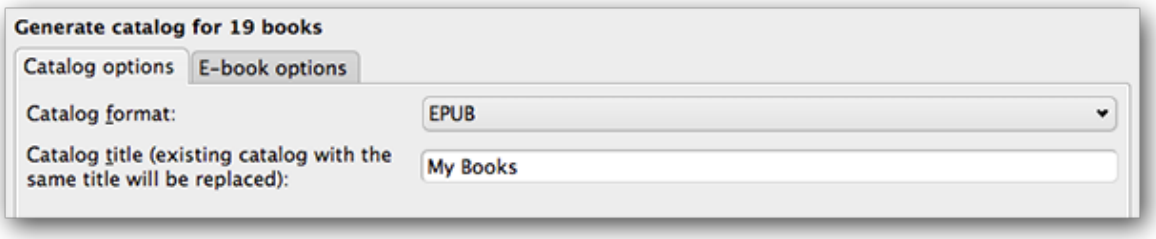
10.7.1 Sélection des livres à cataloguer

Si vous voulez que « toute » votre bibliothèque soit cataloguée, supprimez tout critère de recherche ou de filtrage dans votre fenêtre principale. Avec un seul livre sélectionné, tous les livres de votre bibliothèque seront des candidats à l'inclusion dans le catalogue généré. Des livres individuels peuvent être exclus par différents critères ; voir la section *Exclure les genres* (page 257) ci-dessous pour plus d'informations.

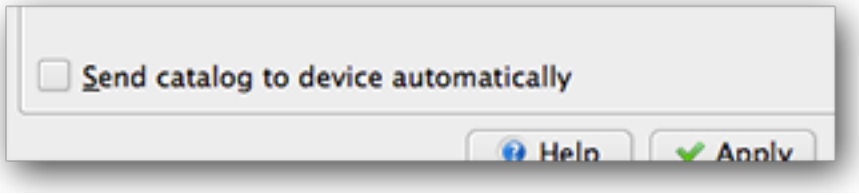
Si vous voulez cataloguer *une certaine* partie de votre librairie, vous avez deux options :

- Créer un sélection multiple des livres que vous voulez cataloguer. Avec plus d'un livre sélectionné dans la fenêtre principale de calibre, seuls les livres sélectionnés seront catalogués.
- Utiliser le champ Recherche ou le Navigateur d'étiquettes pour filtrer les livres affichés. Seuls les livres affichés seront catalogués.

Pour commencer la génération d'un catalogue, cliquer sur la flèche descendante à droite de *Convertir des livres* > *Créer un catalogue des livres de la bibliothèque calibre*. Vous pouvez également ajouter un bouton *Créer un catalogue* à la barre d'outil dans *Préférences* > *Interface* > *Barre d'outils & menus* pour accéder facilement à la boîte de dialogue Générer un catalogue.

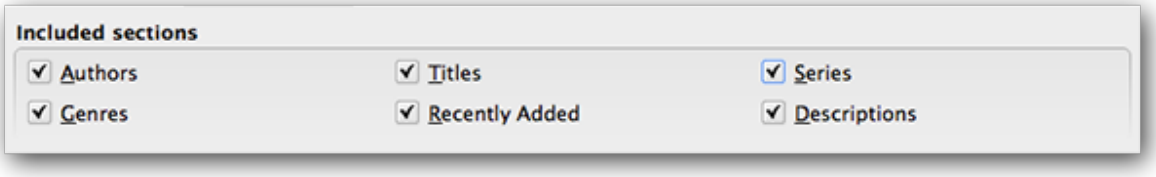


Dans *Options du catalogue*, sélectionnez **AZW3, EPUB ou MOBI** comme format de catalogue. Dans le champ *Titre du catalogue*, donner un nom qui sera utilisé pour le catalogue généré. Si un catalogue avec le même nom et le même format existe déjà, il sera remplacé par le catalogue nouvellement généré.



L'activation de *Envoyer le catalogue automatiquement vers le périphérique connecté* téléchargera le catalogue généré vers un périphérique connecté.

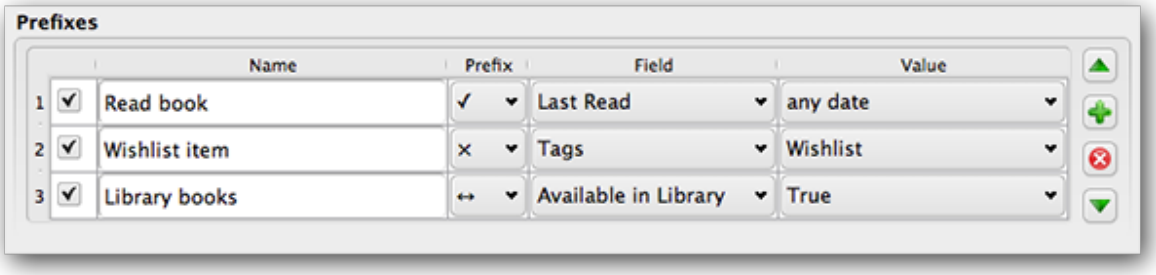
10.7.2 Sections incluses



Les sections activées par une coche seront incluses dans le catalogue généré :

- *Auteurs* - tout les livres, triés par auteur, présentés en format liste. Les livres ne faisant pas partie d'une série sont listés avant ceux qui sont dans une série.
- *Titres* - tous les livres, triés par titre, présentés en format liste.
- *Séries* - Tous les livres qui font partie d'une série , triés par séries, présentés en format liste.
- *Genres* - genres individuels présentés dans une liste, triée par Auteur et Séries.
- *Récemment Ajoutés* - tous les livres, triés dans l'ordre chronologique inverse. La liste inclut les livres ajoutés dans les 30 derniers jours, ensuite un listing mois par mois des livres ajoutés.
- *Descriptions* - page de description détaillée pour chaque livre, incluant une vignette de couverture et les commentaires. Triés par auteur, avec les livres hors séries listés avant les livres faisant partie de séries.

10.7.3 Préfixes



Les règles de préfixe vous permettent d'ajouter un préfixe à une liste de livres quand certains critères sont rencontrés. Par exemple, vous pouvez vouloir marquer des livres que vous avez lu avec une coche, ou des livres de votre liste de souhaits avec un X.

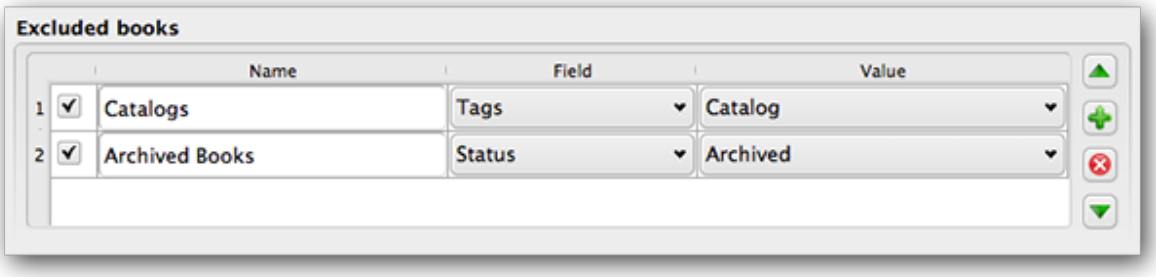
La case à cocher dans la première colonne active la règle. :guilabel :"Nom" est un nom de règle que vous fournissez. *Champ* est soit *Etiquettes* ou une colonne personnalisée de votre bibliothèque. *Valeur* est le contenu de *Champ* à faire correspondre. Quand une règle de préfixe est satisfaite, le livre sera marqué avec le *Préfixe* sélectionné.

Trois règles de préfixe ont été spécifiées dans l'exemple ci-dessus :

1. *Livre lu* spécifie qu'un livre avec n'importe quelle date dans une colonne personnalisée nommée *Dernier lu* sera préfixé avec le symbole coché.
2. L'élément *Liste de souhaits* spécifie que tout livre avec une étiquette *Liste de souhaits* sera préfixée avec le symbole X.
3. *Livres de la bibliothèque* spécifie que tout livre avec une valeur True (ou Oui) dans une colonne personnalisée *Disponible dans la bibliothèque* sera préfixé avec un symbole flèche double.

La première règle correspondante active le préfixe. des règles désactivées ou incomplètes sont ignorées.

10.7.4 Livres exclus



Les règles d'exclusion vous permettent de spécifier des livres qui ne devront pas être catalogués.

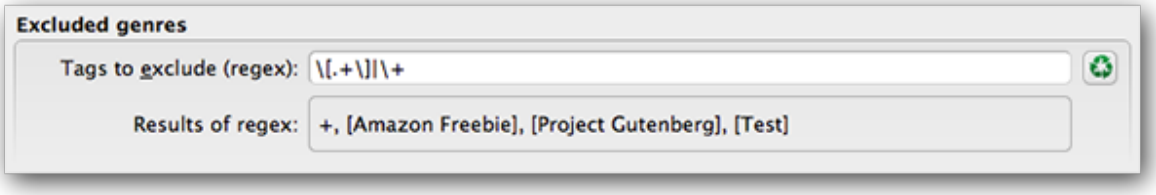
La case à cocher dans la première colonne active la règle. :guilabel :"Nom" est un nom de règle que vous fournissez. *Champ* est soit *Etiquettes* ou une colonne personnalisée de votre bibliothèque. *Valeur* est le contenu de *Champ* à faire correspondre. Quand une règle d'exclusion est satisfaite, le livre sera exclu du catalogue généré.

Deux règles d'exclusions ont été spécifiées dans l'exemple ci-dessous :

1. La règle *Catalogues* spécifie que tout livre avec une valeur *Catalogue* sera exclu du catalogue généré.
2. La règle des Livres *Archived* spécifie que tout livre avec une étiquette *Archived* dans la colonne personnalisée *Status* sera exclu du catalogue généré.

Toutes les règles sont évaluées pour chaque livre. Les règles désactivées ou incomplètes sont ignorées.

10.7.5 Exclure les genres



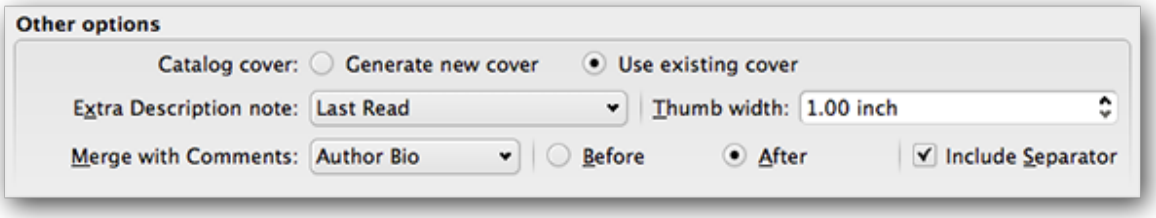
Quand le catalogue est généré, les étiquettes dans votre base de données sont utilisées comme genres. Par exemple, si vous pouvez utiliser les étiquettes `Fiction` et `Non fiction`. Ces étiquettes deviennent des genres dans le catalogue généré, avec les livres listés sous leurs listes de genre respectives basées sur les étiquettes qui leurs ont été assignées.Un livre sera listé dans chaque section genre pour laquelle il a une étiquette correspondante.

Vous pouvez avoir utilisé certaines étiquettes pour d'autres finalités, peut-être un + pour indiquer un livre lu, ou une étiquette entre crochets comme `[Amazon Freebie]` pour indiquer une source de livre. La regex *Genres exclus* vous permet de spécifier des étiquettes que vous ne voulez pas utiliser comme genre dans le catalogue généré. Le modèle par défaut regex d'exclusion `\[.+\]\+` exclut toutes étiquettes de la forme `[étiquette]`, aussi bien que d'exclure +, l'étiquette par défaut pour les livres lus, d'être utilisés comme genres dans le catalogue généré.

Vous pouvez aussi utiliser un nom d'étiquette exact dans une regex. Par exemple, `[Amazon Freebie]` ou `[Project Gutenberg]`. Si vous voulez lister de multiples étiquettes exactes pour exclusion, mettez un caractère pipe (barre verticale) entre eux : `[Amazon Freebie]|[Project Gutenberg]`.

Résultats de la regex vous montre quelles étiquettes seront exclues quand le catalogue sera construit, basé sur les étiquettes dans votre base de données et sur le modèle regex entré. Les résultats sont modifiés lorsque vous modifiez le modèle regex.

10.7.6 Autres options



Couverture de Catalogue spécifie soit de générer une nouvelle couverture ou d'en utiliser une existante. Il est possible de créer une couverture personnalisée pour vos catalogues - voir *Couvertures de catalogue personnalisées* (page 259) pour plus d'information. Si vous avez créer une couverture personnalisée que vous voulez réutiliser, sélectionner *Utiliser une couverture existante*. Autrement, sélectionnez *Générer une nouvelle couverture*.

Note de Description supplémentaire spécifie que le contenu d'une colonne personnalisée est inséré dans la page de Description, juste après la vignette de couverture. Par exemple, vous pouvez vouloir afficher la date où vous avez lu un livre pour la dernière fois en utilisant la colonne personnalisée *Dernier lu*. Pour une utilisation avancée de la fonction note de Description, voir ce billet dans le forum calibre[99].

Largeur miniature spécifie une préférence de largeur pour les vignettes de couverure inclues avec les pages de Descriptions. Les vignettes sont mises en cache pour améliorer les performances. Pour expérimenter avec différentes largeurs, essayer en générant un catalogue avec juste quelques livres jusqu'à ce que vous ayez déterminé votre largeur préférée, générer alors votre catalogue complet. La première fois qu'un catalogue est généré avec une nouvelle largeur de vignette, les performances seront moindres, mais les constructions ultérieures du catalogue prendront avantage de la mise en cache des vignettes.

Fusionner avec les commentaires spécifie une colonne personnalisée dont le contenu sera fusionnée de manière non destructive avec la métadonnée commentaires pendant la génération du catalogue.. Par exemple,, vous pouvez avoir une colonne personnalisée *Bio Auteur* que vous aimeriez ajouter à la métadonnée commentaires. Vous pouvez choisir d'insérer le contenu de la colonne personnalisée *avant ou après* la section commentaires, et optionnellement séparer le contenu ajouté avec une ligne de séparation horizontale. Les types de colonnes personnalisées éligibles incluent `texte, commentaires, et composites`.

[99] https://www.mobileread.com/forums/showpost.php?p=1335767&postcount=395

10.7.7 Couvertures de catalogue personnalisées

Avec l'" extension Generate Cover[100] installée, vous pouvez créer des couvertures personnalisées pour votre catalogue. Pour installer l'extension, allez à *Préférences* > *Avancé*> *Extensions* > *Obtenir de nouvelles extensions*.

10.7.8 Ressources d'aide supplémentaires

Pour plus d'informations sur la fonction Catalogue de calibre, voir le forum MobileRead Creating Catalogs - Start here[101], où vous pouvez trouver de l'information sur comment personnaliser les modèles de catalogue, et comment soumettre un rapport de bogue.

Pour poser des questions ou discuter de la fonctionnalité Catalogue de calibre avec d'autres utilisateurs, visitez le forum MobileRead Library Management[102].

10.8 Bibliothèques virtuelles

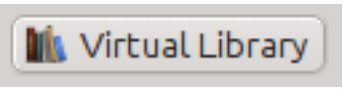
Dans calibre, une Bibliothèque virtuelle est une manière de dire à calibre d'ouvrir uniquement un sous-ensemble d'une bibliothèque normale. Par exemple, vous pouvez vouloir travailler uniquement avec les livres d'un certain auteur, ou les livres ayant seulement une certaine étiquette. Utiliser les Bibliothèques virtuelles est la manière préférable de partitionner votre grande collection de livres en de plus petites sous collections. C'est plus intéressant que de diviser votre bibliothèque en de multiples bibliothèques plus petites vu que, quand vous voulez rechercher dans votre collection entière, vous devez simplement revenir à la bibliothèque complète. Il n'y a pas de manière de rechercher simultanément au travers de multiples bibliothèques dans calibre.

Une Bibliothèque virtuelle est différente d'une simple recherche. Une recherche restreindra uniquement la liste des livres affichés dans votre liste de livres. Une Bibliothèque virtuelle fait cela, et en complément elle restreint les entrées affichées dans le *Navigateur d'étiquettes* sur la gauche. Le Navigateur d'étiquettes affichera uniquement les étiquettes, auteurs, séries,

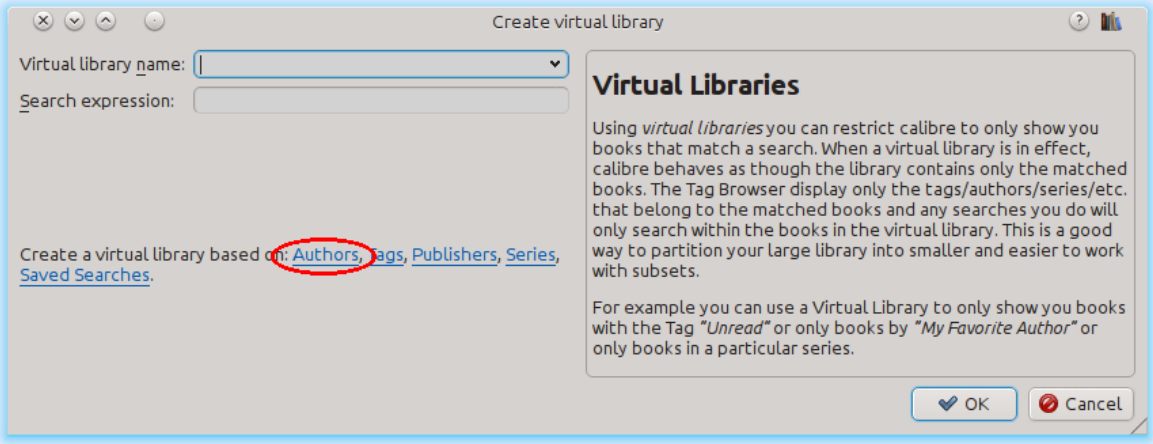
[100] https://www.mobileread.com/forums/showthread.php?t=124219
[101] https://www.mobileread.com/forums/showthread.php?t=118556
[102] https://www.mobileread.com/forums/forumdisplay.php?f=236

éditeurs, etc.. qui proviennent des livres dans la Bibliothèque virtuelle. Une Bibliothèque virtuelle se comporte donc comme si la bibliothèque réelle contenait uniquement l'ensemble restreint de livres.

10.8.1 Création de Bibliothèques virtuelles



Pour utiliser une Bibliothèque virtuelle cliquez sur le bouton *Bibliothèque virtuelle* localisé à la gauche de la Barre de recherche et sélectionnez l'option *Créer une Bibliothèque virtuelle*. Comme premier exemple, créons une Bibliothèque virtuelle qui nous affiche uniquement les livres d'un auteur particulier. Cliquez le lien :guilabel :`Auteurs`comme montré dans l'image ci-dessous et choisissez l'auteur que vous voulez utiliser et cliquez OK.



La boîte de dialogue Créer une Bibliothèque Virtuelle a été remplie pour vous. Cliquez OK et vous verrez qu'une nouvelle Bibliothèque virtuelle a été créée, et que vous y avez été automatiquement redirigé, celle-ci affiche uniquement les livres de l'auteur sélectionné. En ce qui concerne calibre, c'est comme si votre bibliothèque contenait seulement les livres de l'auteur sélectionné.

Vous pouvez revenir à la bibliothèque complète à n'importe quel moment en cliquant encore une fois sur *Bibliothèque virtuelle* et en sélectionnant l'entrée nommée <*None*>.

Les Bibliothèques virtuelles sont basées sur des *recherches*. Vous pouvez utilisez n'importe quelle recherche comme base à une Bibliothèque virtuelle. La Bibliothèque virtuelle contiendra uniquement les livres assortis à la recherche. Tout d'abord, entrez la recherche que vous voulez utiliser dans la Barre de recherche ou construisez une recherche en utilisant le *Navigateur d'étiquettes*. Quand vous êtes satisfait des résultats renvoyés, cliquez le bouton *Bibliothèque virtuelle*, choisissez guilabel :*Créer une Bibliothèque Virtuelle* et entrez un nom pour la nouvelle Bibliothèque virtuelle. La Bibliothèque virtuelle sera alors créée sur base de la recherche. Les recherches sont très puissantes, pour des exemples de ce que vous pouvez faire avec celles-ci, voir *L'interface de recherche* (page 13).

Exemples de Bibliothèques virtuelles utiles

- **Les livres ajoutés à calibre au cours du dernier jour :**
  date :>1daysago
- **Les livres ajoutés à calibre au cours du dernier mois :**
  date :>30daysago
- **Les livres avec une notation de 5 étoiles :**
  rating :5
- **Les livres avec une notation d'au moins 4 étoiles :**
  rating :>=4
- **Les livres sans notation :**
  rating :false
- **Les périodiques téléchargés par la fonction Récupérer des Actualités dans calibre :**
  tags :=News and author :=calibre
- **Les livres sans étiquettes :**
  tags :false
- **Les livres sans couvertures :**
  cover :false

10.8.2 Travailler avec des Bibliothèques virtuelles

Vous pouvez éditer une Bibliothèque virtuelle précédemment créée ou la supprimer, en cliquant *Bibliothèque virtuelle* et en choisissant l'action appropriée.

Vous pouvez dire à calibre que vous voulez toujours appliquer une bibliothèque virtuelle particulière quand la bibliothèque habituelle est ouverte, en allant dans *Préférences → Interface → Comportement*.

Vous pouvez rapidement utilisez la recherche actuelle comme une Bibliothèque virtuelle temporaire en cliquant le bouton *Bibliothèque virtuelle* et en choisissant l'entrée **recherche actuelle*

Vous pouvez afficher toutes les Bibliothèques virtuelles disponibles comme des onglets au dessus de la liste de livres. C'est particulièrement pratique si vous aimez très souvent basculer entre les bibliothèques virtuelles. Cliquez sur le bouton *Bibliothèque virtuelle* et sélectionnez *Afficher les Bibliothèques virtuelles sous forme d'onglets*. Vous pouvez réarranger les onglets par glisser et déposer et fermer ceux que vous ne voulez pas voir. Les onglets fermés peuvent être restaurés en faisant un clic droit sur sur la barre des onglets.

10.8.3 Utilisation des Bibliothèques virtuelles dans les recherches

Vous pouvez rechercher des livres qui sont dans une Bibliothèque virtuelle en utilisant le préfixe `vl:`. Par exemple, `vl:Lu` trouvera tous les livres dans la Bibliothèque virtuelle *Lu*. La recherche `vl:Lu and vl:"Science Fiction"` trouvera tous les livres qui sont tant dans les Bibliothèques virtuelles *Lu* que *Science Fiction*.

La valeur suivant `vl:` doit être le nom d'une Bibliothèque virtuelle. Si le nom de la Bibliothèque virtuelle contient des espaces entourez-le alors avec des guillemets.

Une utilisation d'une recherche Bibliothèque virtuelle se trouve dans le Serveur de contenu. Dans *Préférences → Partager sur le net → Exiger nom d'utilisateur et mot de passe* vous pouvez limiter les bibliothèques calibre visibles pour un utilisateur. Pour chaque bibliothèque visible vous pouvez spécifier une expression de recherche pour limiter par après les livres qui seront vus. Utilisez `vl:"Nom Bibliothèque virtuelle"` pour limiter les livres à ceux dans une Bibliothèque virtuelle.

10.8.4 Utilisation des restrictions supplémentaires

Vous pouvez limiter les livres montrés dans une Bibliothèque virtuelle à l'aide de :guilabel : *Restrictions supplémentaires*. Une restriction supplémentaire est recherche enregistrée que vous avez précédemment créée qui peut être appliqué à la Bibliothèque virtuelle actuelle pour limiter les livres montrés dans une bibliothèque virtuelle. Par exemple, disons que vous avez une Bibliothèque virtuelle pour des livres étiquetés comme *Fiction Historique* et une recherche enregistrée qui vous montre les livres non lus, vous pouvez cliquer sur le bouton *Bibliothèque virtuelle* et choisir *Restrictions supplémentaires* pour montrer uniquement les livres de fiction historique non lus. Pour se renseigner sur les recherches enregistrées, voir *Recherches sauvegardées* (page 19).

CHAPITRE 11

Le système d'URL calibre ://

calibre s'enregistre en tant que programme de gestion des URL de calibre ://. Vous pouvez donc les utiliser pour effectuer des actions telles que l'ouverture de livres, la recherche de livres, etc. à partir d'autres programmes/documents ou via la ligne de commande. Par exemple, en exécutant ce qui suit sur la ligne de commande

```
calibre calibre://switch-library/Some_Library
```

S'ouvrira dans calibre avec la bibliothèque nommée `Une certaine bibliothèque`. Les noms de bibliothèque sont le nom du dossier de la bibliothèque avec des espaces remplacés par des traits de soulignement. La valeur spéciale _ signifie la bibliothèque actuelle. Les différents types d'URL sont documentés ci-dessous.

Vous pouvez même placer ces liens à l'intérieur de fichiers HTML ou de documents Word ou similaires et le système d'exploitation exécutera automatiquement calibre pour effectuer l'action spécifiée.

- *Passer à une bibliothèque spécifique* (page 264)
- *Montrer un livre spécifique dans calibre* (page 264)
- *Ouvrir un livre spécifique dans la visionneuse de livre numérique à un emplacement spécifique* (page 264)
- *Recherche des livres* (page 265)
- *Ouvrir une fenêtre de détails du livre pour un livre dans une bibliothèque* (page 265)
- *Open the notes associated with an author/series/etc.* (page 265)
- *Codage hexadécimal des paramètres d'URL* (page 266)

11.1 Passer à une bibliothèque spécifique

La syntaxe de l'URL est

```
calibre://switch-library/Library_Name
```

Les noms des bibliothèques sont le nom du dossier de la bibliothèque avec des espaces remplacés par des traits de soulignement. La valeur spéciale _ signifie la bibliothèque actuelle. Vous pouvez aussi utiliser :ref :`hex encoding ` pour les noms de bibliothèques, utile si les noms de bibliothèques ont des caractères spéciaux qui nécessiteraient un autre encodage d'URL. Les noms de bibliothèques codés en hexadécimal ressemblent à

```
_hex_-AD23F4BC
```

Où la partie après le préfixe `_hex_-` est le nom de la bibliothèque encodé en UTF-8 et chaque octet représenté par deux caractères hexadécimaux.

11.2 Montrer un livre spécifique dans calibre

La syntaxe de l'URL est

```
calibre://show-book/Library_Name/book_id
```

Cela montrera le livre avec `book_id` (un numéro) dans calibre. Les identifiants des livres peuvent être vus dans l'interface de calibre en passant le curseur sur le lien *Cliquer pour ouvrir* dans le panneau *Détails du livre*, c'est le numéro entre parenthèses à la fin du chemin d'accès au dossier du livre.

Vous pouvez copier un lien vers le livre actuel affiché dans calibre en faisant un clic droit sur le panneau *Détails du livre* et en choisissant *Copier le lien vers le livre*.

Si une recherche est active et que le livre ne correspond pas à la recherche, la recherche est annulée.

Si une bibliothèque virtuelle est sélectionnée, calibre l'utilisera pour afficher le livre. Si le livre n'est pas trouvé dans cette bibliothèque virtuelle, celle-ci est effacée.

Si vous souhaitez passer à une bibliothèque virtuelle particulière lorsque vous montrez le livre, utilisez :

```
calibre://show-book/Library_Name/book_id?virtual_library=Library%20Name
or
calibre://show-book/Library_Name/book_id?encoded_virtual_library=hex_encoded_virtual_
↪library_name
```

en remplaçant les espaces dans le nom de la bibliothèque virtuelle par `%20`. Si le livre ne se trouve pas dans cette bibliothèque virtuelle, la bibliothèque virtuelle est ignoré.

11.3 Ouvrir un livre spécifique dans la visionneuse de livre numérique à un emplacement spécifique

La syntaxe de l'URL est

```
calibre://view-book/Library_Name/book_id/book_format?open_at=location
```

Ici, `book_format` est le format du livre, par exemple, `EPUB` ou `MOBI` et `location` est un emplacement optionnel à l'intérieur du livre. La façon la plus simple d'obtenir ces liens est d'ouvrir un livre dans la visionneuse, puis dans les commandes de la visionneuse, sélectionnez *Aller à → Emplacement* et là un tel lien sera donné que vous pourrez copier/coller ailleurs.

11.4 Recherche des livres

La syntaxe de l'URL est

```
calibre://search/Library_Name?q=query
calibre://search/Library_Name?eq=hex_encoded_query
```

Ici, toute requête est valide expression de recherche. Si l'expression de recherche est compliquée, code la comme une chaîne hexadécimale et utilise `eq` à la place. L'omission de la requête entraînera l'annulation de la recherche en cours.

Par défaut, si une Bibliothèque virtuelle est sélectionnée, calibre l'effacera avant de faire la recherche pour s'assurer que tous les livres sont trouvés. Si vous souhaitez préserver la bibliothèque virtuelle, utilisez

```
calibre://search/Library_Name?q=query&virtual_library=_
```

Si vous souhaitez passer à une Bibliothèque virtuelle particulière, utilisez

```
calibre://search/Library_Name?virtual_library=Library%20Name
or
calibre://search/Library_Name?encoded_virtual_library=hex_encoded_virtual_library_name
```

en remplaçant les espaces dans le nom de la bibliothèque virtuelle par `%20`.

Si vous effectuez une recherche dans calibre et souhaitez générer un lien pour celle-ci, vous pouvez le faire en cliquant avec le bouton droit de la souris sur la barre de recherche et en choisissant *Copier la recherche en tant qu'URL*.

11.5 Ouvrir une fenêtre de détails du livre pour un livre dans une bibliothèque

La syntaxe de l'URL est

```
calibre://book-details/Library_Name/book_id
```

Cette opération permet d'ouvrir une fenêtre de détails du livre spécifié dans la bibliothèque spécifiée sans modifier la bibliothèque actuelle ou le livre sélectionné.

11.6 Open the notes associated with an author/series/etc.

La syntaxe de l'URL est

```
calibre://book-details/Library_Name/Field_Name/id_Item_Id
```

This opens a window showing the notes of the specified item. The easiest way to create such URLs is to show the notes you want in calibre and click the *Copy URL* button to copy the URL to the clipboard and paste it wherever you need.

Here `Field_Name` is the name of the columns such as `authors` or `tags`. For user created columns, replace the leading `#` in the field name with an underscore, so `#mytags` becomes `_mytags`.

In addition to specifying items by id using `Item_Id` you can also specify them by name using either `val_Item_Name` or `hex_Hex_Encoded_Item_Name`. For example :

```
calibre://book-details/Library_Name/authors/val_John%20Doe
```

11.7 Codage hexadécimal des paramètres d'URL

Le codage hexadécimal des paramètres d'URL se fait en codant d'abord le paramètre en UTF-8 octets, puis en remplaçant chaque octet par deux caractères hexadécimaux représentant l'octet. Par exemple, la chaîne `abc` correspond aux octets `0x61 0x62 et 0x63` en UTF-8, donc la version codée est la chaîne : `616263`.

CHAPITRE 12

Personnaliser calibre

calibre a une conception hautement modulaire. Plusieurs de ses parties peuvent être personnalisées. Ici, vous apprendrez :

- comment utiliser les variables d'environnement et les *ajustements* pour personnaliser le comportement de calibre,
- comment spécifier vos propres ressources statiques, comme les icônes et les modèles, pour remplacer les valeurs par défaut
- comment utiliser les *plugins* pour ajouter des fonctionnalités à calibre.
- comment partager des thèmes d'icônes et des plugins avec d'autres utilisateurs de calibre.
- pour voir comment créer des *recettes* pour ajouter de nouvelles sources de contenu en ligne à calibre, visitez la Section *Ajouter votre site web favori d'actualités* (page 33).

Note : Notez que bien que les thèmes d'icônes et les plugins soient indexés et téléchargeables via le dispositif de mise à jour intégré de calibre, ils ne font pas partie de calibre, et leurs emplacements canoniques pour le support et le code source sont sur les forums Mobileread[103] dans leurs fils de discussion de support.

- *Variables d'environnement* (page 301)
- *Ajustements* (page 302)
- *Outrepasser les Icônes, les modèles, etc.* (page 313)
- *Créer votre propre thème d'icônes pour calibre* (page 313)
- *Personnalisation de calibre avec des extensions* (page 314)

[103] https://www.mobileread.com/forums/forumdisplay.php?f=166

12.1 Documentation API pour les extensions

Définit différentes classes abstraites de base qui peuvent être sous-classées pour créer de puissantes extensions. Les classes utiles sont :

- *Extension* (page 268)
- *FileTypePlugin* (page 270)
- *Les extensions métadonnées* (page 272)
- *Extension de catalogue* (page 273)
- *Extension de téléchargement de métadonnées* (page 274)
- *Extension de conversion* (page 278)
- *Pilotes de périphérique* (page 281)
- *Actions interface utilisateur* (page 295)
- *Extensions préférences* (page 299)

12.1.1 Extension

class calibre.customize.**Plugin**(*plugin_path*)

Une extension calibre. Les membres utiles incluent :

- `self.installation_type` : Enregistre la façon dont le plugin a été installé.
- **`self.plugin_path` : Enregistre le chemin vers le fichier ZIP le contenant** cette extension ou None si c'est une extension intégrée
- **`self.site_customization` : Enregistre une chaîne de personnalisation introduite** par l'utilisateur.

Les méthodes qui devraient être annulées dans les sous-classes :

- *`initialize()`* (page 269)
- *`customization_help()`* (page 270)

Méthodes utiles :

- *`temporary_file()`* (page 270)
- `__enter__()`
- *`load_resources()`* (page 269)

`supported_platforms = []`

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

`name = 'Trivial Plugin'`

Le nom de cette extension. Vous devez le fixer à autre chose que Trivial Plugin pour que cela fonctionne.

`version = (1, 0, 0)`

La version de cette extension en tant que 3-tuple (majeure, mineure, révision)

description = 'Ne fait absolument rien'

Une courte chaine décrivant ce que fait l'extension

author = 'Inconnu(e)'

L'auteur de cette extension

priority = 1

Quand plus d'une extension existe pour un type de fichier, les extensions sont exécutées dans l'ordre de priorité décroissant. Les extensions avec la priorité la plus haute seront exécutées en premier. La priorité la plus haute possible est `sys.maxsize`. La priorité par défaut est de 1.

minimum_calibre_version = (0, 4, 118)

La version la plus récente de calibre que nécessite cette extension

installation_type = None

La façon dont ce plugin est installé

can_be_disabled = True

Si False, l'utilisateur ne pourra pas désactiver cette extension. À utiliser avec soin.

type = 'Base'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

initialize()

Appelé une fois lorsque les extensions calibre sont initialisées. Les extensions sont ré-initialisées à chaque fois qu'une nouvelle extension est ajoutée. Notez également que si l'extension est en exécution dans un processus de travail, tel que pour ajouter des livres, alors l'extension sera initialisée pour tout nouveau processus de travail.

Effectuer n'importe quel initialisation spécifique d'extension ici, comme l'extraction de ressource à partir du fichier ZIP de l'extension. Le chemin vers le ZIP est disponible comme `self.plugin_path`.

Notez que `self.site_customization` n'est **pas** disponible à ce point-ci.

config_widget()

Implémenter cette méthode et *save_settings()* (page 269) dans votre extension pour utiliser une boîte de dialogue de configuration personnalisée, plutôt que d'utiliser la simple personnalisation par ligne par défaut.

Cette méthode, si implémentée, doit retourner un QWidget. Le widget peut avoir une méthode optionnelle validate() qui ne prend pas d'argument et qui est appelée dès que l'utilisateur clique sur OK. Les changements sont appliqués si et seulement si la méthode retourne True.

Si pour une raison quelconque, vous ne pouvez pas réaliser la configuration à ce moment, renvoie un tuple de deux chaines (message, détails), qui seront affichés comme un dialogue d'alerte à l'utilisateur et le processus sera interrompu.

save_settings(*config_widget*)

Sauve les paramètres spécifiés par l'utilisateur avec config_widget

Paramètres

config_widget – Le widget renvoyé par *config_widget()* (page 269).

do_user_config(*parent=None*)

Cette méthode montre une boîte de dialogue de configuration pour cette extension. Elle renvoie True si l'utilisateur clique OK, sinon False. Les changements sont appliqués automatiquement.

load_resources(*names*)

Si cette extension est donnée dans un fichier ZIP (extension ajoutée par l'utilisateur), cette méthode vous permettra de charger des ressources du fichier ZIP.

Par exemple pour charger une image

```
pixmap = QPixmap()
pixmap.loadFromData(self.load_resources(['images/icon.png'])['images/icon.png
↪'])
icon = QIcon(pixmap)
```

Paramètres

names – Liste de chemins vers les ressources dans le ZIP utilisant / comme séparateur

Renvoie

Un dictionnaire sous la forme `{name : file_contents}`. Tout nom qui n'est pas trouvé dans le fichier ZIP ne sera pas présent dans le dictionnaire.

customization_help(*gui=False*)

Retourne une chaîne offrant de l'aide sur comment personnalisé cette extension. Par défaut avance un `NotImplementedError`, qui indique que l'extension ne nécessite pas de personnalisation.

Si vous ré-implémentez cette méthode dans votre sous-classe, il sera demandé à l'utilisateur d'entrer une chaîne pour personnalisé cette extension. La chaîne de personnalisation sera disponible comme `self.site_customization`.

La personnalisation de site peut-être n'importe quoi, par exemple, le chemin vers un binaire nécessaire sur l'ordinateur de l'utilisateur.

Paramètres

gui – Si True renvoie l'aide HTML, sinon renvoie l'aide en texte simple.

temporary_file(*suffix*)

Renvoie un objet ressemblant à un fichier qui est un fichier temporaire dans les fichiers systèmes. Ce fichier reste disponible après avoir été fermé et sera supprimé uniquement après l'extinction de l'interpréteur. Utilisé le membre `name` de l'objet renvoyé pour accéder au chemin complet vers le fichier temporaire créé.

Paramètres

suffix – Le suffixe qu'aura le fichier temporaire.

cli_main(*args*)

Cette méthode est le point d'entrée principal pour les interfaces d'extensions en ligne de commande. Elle est appelée lorsque l'utilisateur fait : calibre-debug -r « Nom de l'Extension ». Tous les arguments sont présents dans les variables args.

12.1.2 FileTypePlugin

class calibre.customize.**FileTypePlugin**(*plugin_path*)

Bases : *Plugin* (page 268)

Une extension qui est associée à un ensemble particulier de types de fichiers.

file_types = {}

Ensemble de types de fichiers pour lesquels cette extension devrait fonctionner. Utilisez "*" pour tous les types de fichiers. Par exemple : `{'lit', 'mobi', 'prc'}`

on_import = False

Si True, cette extension est exécutée lorsque des livres sont ajoutés à la base de donnée

on_postimport = False

Si réglé sur True, cette extension est exécutée après que des livres aient été ajoutés à la base de donnée. Dans ce cas, les méthodes postimport et postadd de l'extension sont appelées.

on_postconvert = False

Si True, ce plugin est exécuté après la conversion d'un livre. Dans ce cas, la méthode postconvert du plugin est appelée.

on_postdelete = False

Si True, ce plugin est exécuté après qu'un fichier de livre a été supprimé de la base de données. Dans ce cas, la méthode postdelete du plugin est appelée.

on_preprocess = False

Si True, cette extension est exécutée juste avant une conversion

on_postprocess = False

Si True, cette extension est exécutée après conversion sur le fichier final produit par l'extension de conversion de sortie.

type = 'Type de fichier'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

run(*path_to_ebook*)

Exécute l'extension. Doit être implémenté dans les sous-classes. Cela devrait effectuer n'importe quelles modifications requises sur le livre numérique et renvoyer le chemin absolu vers le livre numérique modifié. Si aucune modifications ne sont nécessaire, cela devrait renvoyer le chemin vers le livre numérique original. Si une erreur est rencontrée, cela devrait amener une Exception. L'implémentation par défaut retourne simplement le chemin vers le livre numérique original. Notez que le chemin vers le fichier original (avant que n'importe quelle extension type de fichier ne soit exécutée, est disponible comme ceci self.original_path_to_file).

Le livre numérique modifié devrait être créer avec la méthode `temporary_file()`.

Paramètres
path_to_ebook – Chemin absolu vers le livre numérique.

Renvoie
Chemin absolu vers le livre numérique modifié.

postimport(*book_id*, *book_format*, *db*)

post import est appelée, c-à-d après que le fichier de livre ait été ajouté à la base de donnée. Notez que ceci est différent de *postadd()* (page 272) qui est appelée lorsque les données du livre sont créées pour la première fois. Cette méthode est appelée à chaque fois qu'un nouveau fichier est ajouté à un livre enregistré. Elle est utile pour modifier les données du livre se basant sur le contenu du fichier nouvellement ajouté.

Paramètres

- **book_id** – L'id du livre ajouté dans la base de donnée.
- **book_format** – Le type de fichier du livre qui a été ajouté.
- **db** – Base de donnée de la bibliothèque.

postconvert(*book_id*, *book_format*, *db*)

Appelé après la conversion, c'est-à-dire après que le fichier de sortie de la conversion a été ajouté à la base de données. Notez qu'elle n'est exécutée qu'après une conversion, et non après l'ajout d'un livre. Elle est utile pour modifier l'enregistrement du livre en fonction du contenu du fichier nouvellement ajouté.

Paramètres

- **book_id** – L'id du livre ajouté dans la base de donnée.
- **book_format** – Le type de fichier du livre qui a été ajouté.
- **db** – Base de donnée de la bibliothèque.

postdelete(*book_id*, *book_format*, *db*)

Appelé après la suppression, c'est-à-dire après que le fichier du livre a été supprimé de la base de données. Notez qu'il n'est pas exécuté lorsqu'un enregistrement de livre est supprimé, mais seulement lorsqu'un ou plusieurs formats du livre sont supprimés. Elle est utile pour modifier l'enregistrement du livre en fonction du format du fichier supprimé.

Paramètres

- **book_id** – L'id du livre ajouté dans la base de donnée.
- **book_format** – Le type de fichier du livre qui a été ajouté.
- **db** – Base de donnée de la bibliothèque.

postadd(*book_id*, *fmt_map*, *db*)

post add est appelée, c-à-d après qu'un livre ait été ajouté à la base de donnée. Notez que ceci est différent de *postimport()* (page 271), qui est appelée lorsque un fichier de livre est ajouté à un livre existant. postadd() est appelée uniquement lorsqu'un dossier entier de livre avec éventuellement plus d'un fichier de livre a été créé pour la première fois. Ceci est utile si vous souhaitez modifier les données d'un livre dans la base de donnée lorsque le livre est initialement ajouté à calibre.

Paramètres

- **book_id** – L'id du livre ajouté dans la base de donnée.
- **fmt_map** – Le chemin de l'emplacement du format du fichier à partir duquel le format a été ajouté. Notez que ceci pourrait ne pas indiquer un fichier existant, vu que parfois certains fichier sont ajoutés en tant que flux. Dans ce cas on pourrait avoir une valeur fausse ou un chemin inexistant.
- **db** – Base de donnée de la bibliothèque

12.1.3 Les extensions métadonnées

class calibre.customize.**MetadataReaderPlugin**(**args*, ***kwargs*)

Bases : *Plugin* (page 268)

Une extension qui implémente les métadonnées de lecture à partir d'un ensemble de types de fichier.

file_types = {}

Ensemble de types de fichier pour lesquels l'extension devrait être exécutée. Par exemple : `set(['lit', 'mobi', 'prc'])`

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

version = (7, 9, 0)

La version de cette extension en tant que 3-tuple (majeure, mineure, révision)

author = 'Kovid Goyal'

L'auteur de cette extension

type = 'Lecteur de métadonnées'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

get_metadata(*stream*, *type*)

Renvoie les métadonnées pour le fichier représenté par le flux (un objet-fichier qui supporte la lecture). Renvoie une exception lorsqu'il y a une erreur avec les données d'entrée.

Paramètres

type – Le type de fichier. Garanti d'être l'une des entrées dans *file_types* (page 272).

Renvoie

Un objet `calibre.ebooks.metadata.book.Metadata`

class calibre.customize.**MetadataWriterPlugin**(**args*, ***kwargs*)

Bases : *Plugin* (page 268)

Une extension qui implémente les métadonnées de lecture à partir d'un ensemble de types de fichier.

file_types = {}

Ensemble de types de fichier pour lesquels l'extension devrait être exécutée. Par exemple : `set(['lit', 'mobi', 'prc'])`

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

version = (7, 9, 0)

La version de cette extension en tant que 3-tuple (majeure, mineure, révision)

author = 'Kovid Goyal'

L'auteur de cette extension

type = 'Créateur de métadonnées'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

set_metadata(*stream*, *mi*, *type*)

Règle les métadonnées pour le fichier représenté par le flux (un objet-fichier qui supporte la lecture). Renvoie une exception lorsqu'il y a une erreur avec les données d'entrée.

Paramètres

- **type** – Le type de fichier. Garanti d'être l'une des entrées dans *file_types* (page 273).
- **mi** – Un objet `calibre.ebooks.metadata.book.Metadata`

12.1.4 Extension de catalogue

class calibre.customize.**CatalogPlugin**(*plugin_path*)

Bases : *Plugin* (page 268)

Une extension qui implémente un générateur de catalogue.

file_types = {}

Le type de fichier de sortie pour lesquels l'extension devrait être exécutée. Par exemple : "epub" ou "xml"

type = 'Générateur de catalogue'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

cli_options = []

Les options d'analyseur de CLI spécifiques pour cette extension, déclarée en tant qu"*Option namedtuple* :

from collections import namedtuple Option = namedtuple("Option", "option, default, dest, help") cli_options = [Option("–catalog-title", default = "My Catalog", dest = "catalog_title", help = (_("Title of generated catalog. nDefault :") + « "» + "%default" + « "»))] cli_options parsed in calibre.db.cli.cmd_catalog :option_parser()

initialize()

Si l'extension n'est pas native, copier les fichiers .ui et .py de l'extension à partir du fichier ZIP vers $TMPDIR. Un entrée sera dynamiquement générée et ajotée à la boîte de dialogue du Catalogue d'Options dans calibre.gui2.dialogs.catalog.py :Catalog

run(*path_to_output*, *opts*, *db*, *ids*, *notification=None*)

Exécute l'extension. Doit être implémenté dans les sous-classes. Cela devrait généré le catalogue dans le format spécifié dans file_types, renvoyant le chemin absolu vers le fichier catalogue généré. SI un erreur est rencontrée, cela devrait faire apparaître une Exception.

Le fichier catalogue généré devrait être créer avec la méthode `temporary_file()`.

Paramètres

- **path_to_output** – Chemin absolu vers le fichier catalogue généré.
- **opts** – Un dictionnaire d'arguments mot-clef
- **db** – Un objet BaseDeDonnéeLibrairie2

12.1.5 Extension de téléchargement de métadonnées

class calibre.ebooks.metadata.sources.base.**Source**(**args*, ***kwargs*)

Bases : *Plugin* (page 268)

type = 'Source de métadonnées'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

author = 'Kovid Goyal'

L'auteur de cette extension

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

capabilities = frozenset({})

Ensemble de capacités supportées par cet extension. Les capacités utiles sont : "identify", "cover"

touched_fields = frozenset({})

Liste de champ de métadonnée qui peuvent potentiellement être téléchargée par cette extension pendant la phase d'identification

has_html_comments = False

Fixer ceci à True si votre extension renvoie des commentaires formatés en HTML

supports_gzip_transfer_encoding = False

Paramétrer ceci à True veut dire que l'objet navigateur indiquera qu'il prend en charge l'encodage de transfert gzip. Ceci peu accélérer les téléchargements, soyez sûr que la source supporte effectivement les encodages de transfert gzip correctement.

ignore_ssl_errors = False

Paramétrer à TRUE pour ignorer les erreurs de certificat HTTPS lors de la connexion à cette source.

cached_cover_url_is_reliable = True

Des URLs de couverture dans le cache peuvent parfois être peu sûr (càd. que le téléchargement pourrait échouer ou que l'image pourrait ne pas fonctionner). Si c'est souvent le cas avec cette source, fixer à False

options = ()

Un liste d'objet `Option`. Ils seront utilisés pour construire automatiquement le widget de configuration pour cette extension

config_help_message = None

Une chaîne qui sera affichée au sommet du widget de configuration pour cette extension

can_get_multiple_covers = False

Si True cette source peut renvoyer de multiples couvertures pour une demande donnée

auto_trim_covers = False

Si fixer à True les couvertures téléchargées par cette extension seront automatiquement rognées.

prefer_results_with_isbn = True

Si réglé à True, et cette source retourne de multiples résultats pour une requête, certains de ceux-ci ont des ISBN et certains de ceux-ci n'en ont pas, le résultat sans ISBN sera ignoré.

is_configured()

Renvoie False si votre extension nécessite d'être configurée avant de pouvoir être utilisé. Par exemple, cela pourrait nécessiter un utilisateur/mot de passe/clef de greffon.

customization_help()

Retourne une chaîne offrant de l'aide sur comment personnalisé cette extension. Par défaut avance un `NotImplementedError`, qui indique que l'extension ne nécessite pas de personnalisation.

Si vous ré-implémentez cette méthode dans votre sous-classe, il sera demandé à l'utilisateur d'entrer une chaîne pour personnalisé cette extension. La chaîne de personnalisation sera disponible comme `self.site_customization`.

La personnalisation de site peut-être n'importe quoi, par exemple, le chemin vers un binaire nécessaire sur l'ordinateur de l'utilisateur.

Paramètres

gui – Si True renvoie l'aide HTML, sinon renvoie l'aide en texte simple.

config_widget()

Implémenter cette méthode et *save_settings()* (page 275) dans votre extension pour utiliser une boîte de dialogue de configuration personnalisée, plutôt que d'utiliser la simple personnalisation par ligne par défaut.

Cette méthode, si implémentée, doit retourner un QWidget. Le widget peut avoir une méthode optionnelle validate() qui ne prend pas d'argument et qui est appelée dès que l'utilisateur clique sur OK. Les changements sont appliqués si et seulement si la méthode retourne True.

Si pour une raison quelconque, vous ne pouvez pas réaliser la configuration à ce moment, renvoie un tuple de deux chaines (message, détails), qui seront affichés comme un dialogue d'alerte à l'utilisateur et le processus sera interrompu.

save_settings(*config_widget*)

Sauve les paramètres spécifiés par l'utilisateur avec config_widget

Paramètres

config_widget – Le widget renvoyé par *config_widget()* (page 275).

get_author_tokens (*authors*, *only_first_author=True*)

Prend une liste d'auteurs et retourne une liste de jetons utiles pour une demande de recherche AND. Cette fonction tente de renvoyer des jetons dans l'ordre prénom, second nom, nom de famille, en escomptant que si une virgule est dans le nom d'auteur, le nom est dans le nom de famille, sous forme des autres noms.

get_title_tokens (*title*, *strip_joiners=True*, *strip_subtitle=False*)

Prend un titre et retourne une liste de jetons utiles pour une requête de recherche AND. Exclus les conjonctions(optionnel) et la ponctuation.

split_jobs (*jobs*, *num*)

Divise une liste de travaux en sous groupe, aussi uniformément que possible

test_fields (*mi*)

Retourne le premier champs de self.touched_fields qui est nul sur le demi objet

clean_downloaded_metadata (*mi*)

Appelez cette méthode dans votre méthode d'identification de votre extension pour normaliser les métadonnées avant de rentrer l'objet Métadonnées dans le result_queue. Vous pouvez, bien sûr, utiliser un algorithme personnalisé adapté pour la source de métadonnée.

get_book_url (*identifiers*)

Renvoie un 3-tuple ou None. Le 3-tuple est sous la forme : (identifier_type, identifier_value, URL). L'URL est celle du livre identifié par les identifiants à la source. identifier_type, identifier_value, spécifiant l'identifiant correspondant à l'URL. Cet URL doit être consultable pour un être humain utilisant un navigateur. Elle est censée offrir un lien cliquable pour l'utilisateur pour visiter facilement les pages du livre à la source. Si aucun URL n'est trouvé, renvoie None. Cette méthode doit être rapide, et cohérente, à implémenter uniquement s'il est possible de construire l'URL à partir d'un schéma connu d'identifiants.

get_book_url_name (*idtype*, *idval*, *url*)

Renvoie un nom lisible par un humain pour la valeur de retour de get_book_url()

get_book_urls (*identifiers*)

Outrepassez cette méthode si vous souhaiteriez retourner de multiples URLs pour ce livre. Renvoie une liste de 3-tuples. Par défaut cette méthode appelle simplement *get_book_url()* (page 276).

get_cached_cover_url (*identifiers*)

Renvoie l'URL de la couverture mise en cache pour le livre identifié par le dictionnaire d'identifiants ou None si cette URL n'existe pas.

Noter que cette méthode doit uniquement renvoyer des URLs validées, càd. pas des URLs qui pourrait finir en une image de couverture générique ou une erreur « not found »

id_from_url (*url*)

Analyser une URL et retourner un tuple du formulaire : (identifier_type, identifier_value). Si l'URL ne correspond pas au modèle de la source de métadonnées, renvoyer None.

identify_results_keygen (*title=None*, *authors=None*, *identifiers={}*)

Renvoie une fonction qui est utilisée pour générer une clef qui peut trier les objets Métadonnées par leur pertinence en fonction d'une recherche donnée (titre, auteurs, identifiants).

Ces clefs sont utilisées pour trier les résultats d'un appel de *identify()* (page 276).

Pour plus de détails sur l'algorithme par défaut voir *InternalMetadataCompareKeyGen* (page 277). Ré-implémentez cette fonction dans votre extension si l'algorithme par défaut n'est pas adapté.

identify (*log*, *result_queue*, *abort*, *title=None*, *authors=None*, *identifiers={}*, *timeout=30*)

Identifie un livre par son Titre/Auteur/ISBN/etc.

Si des identifiants sont spécifiés et qu'aucune n'est trouvée et que cette source de métadonnée ne contient pas tous les identifiants demandé (par exemple, tous les ISBNs d'un livre), cette méthode devrait réessayer avec juste le titre et l'auteur (s'ils ont été spécifiés).

Si cette source de métadonnées fournit aussi des couvertures, l'URL de la couverture devrait être mis en cache pour qu'une demande suivant dans une API pour chercher les couvertures avec le même ISBN/identifiant spécial ne nécessite pas de rechercher l'URL de la couverture à nouveau. Utiliser le cache API pour ceci.

Chaque métadonnée introduite dans result_queue avec cette méthode doit avoir un attribut *source_relevance* qui est un entier indiquant l'ordre dans lequel les résultats seront rendus par la source de métadonnée pour cette recherche. Cet entier sera utilisé par `compare_identify_results()`. Si l'ordre n'a pas d'importance, fixer à zéro pour chaque résultat.

Assurez-vous que toutes les informations de mappage de couverture/ISBN sont mises en cache avant que l'objet Metadata ne soit placé dans result_queue.

Paramètres

- **log** – Un fichier log, utiliser le pour produire des informations de débogage/d'erreurs
- **result_queue** – Une File d'Attente de résultats, les résultats devraient y être mis. Chaque résultat est une métadonnée
- **abort** – Si abort.is_set() renvoie True, annule les futurs exécutions et retourne le plus vite possible
- **title** – Le titre du livre, peut être None
- **authors** – Une liste d'auteur du livre, peut être None
- **identifiers** – Un ensemble d'autres identifiants, le plus souvent {"isbn" :"1234…"}
- **timeout** – « Timeout » en seconde, aucune requête réseau ne devrait durer plus longtemps que « timeout ».

Renvoie

None si aucune erreur n'apparaît, sinon une représentation unicode de l'erreur adaptée pour le visionnage par l'utilisateur

download_cover(*log*, *result_queue*, *abort*, *title=None*, *authors=None*, *identifiers={}*, *timeout=30*, *get_best_cover=False*)

Télécharge une couverture et l'insère dans result_queue. Les paramètres auront tous le même sens que pour *identify()* (page 276). Insère (self, cover_data) dans result_queue.

Cette méthode devrait utiliser les URLs de couvertures mise en cache par efficience le plus souvent possible. Quand aucune donnée n'est présente en cache, la plupart des extensions appel simplement identify et utilise ces résultats.

Si le paramètre get_best_cover est True dans cette extension et que cette extension peut récupérer de multiples couvertures, elle ne devrait prendre que la « meilleure ».

class calibre.ebooks.metadata.sources.base.**InternalMetadataCompareKeyGen**(*mi*, *source_plugin*, *title*, *authors*, *identifiers*)

Génère une sorte de clef pour la comparaison de la pertinence des métadonnée, en fonction d'une recherche. Ceci est utilisé uniquement pour comparer des résultats d'une même source de métadonnées, pas à travers de différentes sources.

La clé de tri garanti que l'ordre croissant de tri soit trié par pertinence décroissante.

L'algorithme est :

- Préfère des résultats qui ont au moins un identificateur égal à celui de la requête
- Préfère des résultats avec une URL de couverture en cache
- Préfère des résultats avec tous les champs disponible remplis
- Préfère les résultats avec la même langue que la langue de l'interface utilisateur actuelle
- Préfère des résultats qui ont exactement le même titre que la recherche
- Préfère des résultats qui ont des plus longs commentaires (plus de 10% plus grand)
- **Utilise la pertinence du résultat comme reporté dans la recherche de source de métadonnées** machine

12.1.6 Extension de conversion

class calibre.customize.conversion.**InputFormatPlugin**(**args*)

Bases : *Plugin* (page 268)

Les InputFormatPlugins sont responsable de la conversion d'un document en HTML+OPF+CSS+etc. Le résultat de la conversion *doit* être encodé en UTF-8. L'action principale se passe dans *convert()* (page 279).

type = 'Entrée de conversion'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

can_be_disabled = False

Si False, l'utilisateur ne pourra pas désactiver cette extension. À utiliser avec soin.

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

file_types = {}

Ensemble de type de fichier pour lesquels l'extension devrait être exécuté. Par exemple : `set(['azw', 'mobi', 'prc'])`

is_image_collection = False

Si True, cette extension d'origine génère un ensemble d'image, une par fichier HTML. Ceci peut-être fait dynamiquement, dans la méthode de conversion si les fichiers d'origines peuvent être des ensembles d'images ou pas. Si vous fixer ceci à True, vous devez implémenter la méthode get_images() qui renvoie une liste d'images.

core_usage = 1

Nombre de cœur du CPU utilisés par cette extension. Une valeur de -1 signifie qu'elle utilise tous les cœurs disponibles

for_viewer = False

Si fixer à True, cette extension d'origine effectuera un exécution spéciale pour que son résultat soit approprié au visionnage

output_encoding = 'utf-8'

L'encodage dans lequel cette extension d'origine crée le fichier. Une valeur de None signifie que l'encodage est indéfini et doit être détecter individuellement

common_options = {<calibre.customize.conversion.OptionRecommendation object>}

Les options partagées par toutes les extensions de Format d'Entrée. N'outrepasses pas en sous-classe. Utiliser *options* (page 279) à la place. Chaque option doit être un exemple de `OptionRecommendation`.

options = {}

Des options pour personnaliser le comportement de cette extension. Chaque option doit être un exemple de `OptionRecommendation`.

recommendations = {}

Un ensemble de 3-tuples sous la forme (option_name, recommended_value, recommendation_level)

get_images()

Renvoie une liste de chemin absolu vers les images, si cette extension d'origine représente une collection d'image. La liste d'images est dans le même ordre que le manifeste et la TdM

convert(*stream*, *options*, *file_ext*, *log*, *accelerators*)

Cette méthode doit être implémentée en sous-classe. Elle doit renvoyer le chemin vers le fichier OPF créé ou un cas `OEBBook`. Tous les fichiers sortant devrait être contenu dans ce dossier. Si cette extension crée des fichiers en dehors du dossier ils doivent être supprimés/marqués pour suppression avant le résultat de cette méthode.

Paramètres

- **stream** – Un objet ressemblant à un fichier qui contient le fichier original.
- **options** – Des options pour personnalisés le processus de conversion. Garanti d'avoir des attributs correspondant à toutes les options déclarée dans cette extension. De plus, il y aura un attribut « verbose »(bavard) qui prend des valeurs entières à partir de zéro. Plus le nombre est haut, plus l'extension est bavarde. Un autre attribut utile est `input_profile` qui est un exemple de `calibre.customize.profiles.InputProfile`.
- **file_ext** – L'extension (sans le .) du fichier original. Garanti d'être un des *file_types* supporté par cette extension.
- **log** – Un objet `calibre.utils.logging.Log`. Tous les fichiers de sorties devraient utiliser cet objet.
- **accelarators** – Un dictionnaire d'informations variées que l'extension d'origine peut accéder facilement pour accélérer les étapes suivantes de la conversion.

postprocess_book(*oeb*, *opts*, *log*)

Appelé pour permettre l'extension d'origine d'effectuer du post-traitement après que le livre ait été divisé.

specialize(*oeb*, *opts*, *log*, *output_fmt*)

Appelé pour permettre à l'extension d'origine pour spécialiser le livre divisé pour un format de sortie particulier. Appelé après postprocess_book et avant toutes transformations effectuée sur le livre divisé.

gui_configuration_widget(*parent*, *get_option_by_name*, *get_option_help*, *db*, *book_id=None*)

Appelé pour créer un widget utilisé pour configurer cette extension dans le GUI de calibre. Le widget doit être une instance de la classe PluginWidget. Voir les d'extensions implémentées d'origine pour des exemples.

class calibre.customize.conversion.**OutputFormatPlugin**(**args*)

Bases : *Plugin* (page 268)

Les OutputFormatPlugins sont responsables de la conversion d'un document OEB (OPF+HTML) vers le livre numérique de sortie.

Le document OEB peut être supposé être encodé en UTF-8. L'action principale se passe dans `conversion()`.

type = 'Sortie de conversion'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

can_be_disabled = False

Si False, l'utilisateur ne pourra pas désactiver cette extension. À utiliser avec soin.

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

file_type = None

Le type de fichier (l'extension sans le point) que cette extension produit

common_options = {<calibre.customize.conversion.OptionRecommendation object>}

Les options partagées par toutes les extensions de Format d'Entrée. N'outrepasses pas en sous-classe. Utiliser *options* (page 280) à la place. Chaque option doit être un exemple de `OptionRecommendation`.

options = {}

Des options pour personnaliser le comportement de cette extension. Chaque option doit être un exemple de `OptionRecommendation`.

recommendations = {}

Un ensemble de 3-tuples sous la forme (option_name, recommended_value, recommendation_level)

property description

str(object="") -> str str(bytes_or_buffer[, encoding[, errors]]) -> str

Créez un nouvel objet chaîne à partir de l'objet donné. Si un encodage ou des erreurs sont spécifiés, l'objet doit alors exposer un tampon de données qui sera décodé en utilisant l'encodage et le gestionnaire d'erreurs donnés. Sinon, renvoie le résultat de object.__str__() (si défini) ou repr(object). encoding default to sys.getdefaultencoding(). errors default to "strict".

convert(*oeb_book*, *output*, *input_plugin*, *opts*, *log*)

Fournit le contenu de *oeb_book* (qui est une instance de `calibre.ebooks.oeb.OEBBook`) au fichier spécifié par la sortie.

Paramètres

- **output** – Soit un fichier soit une chaîne. Si c'est une chaîne, c'est le chemin vers un dossier qui peut exister ou pas. L'extension de sortie devrait écrire ses restitutions dans ce dossier. Si c'est un fichier, l'extension de sortie devrait écrire ces productions dans le fichier.
- **input_plugin** – L'extension d'origine qui a été utilisé au début de la pipeline de conversion.
- **opts** – Des options de conversions. Garantie d'avoir des attributs correspondant aux OptionRecommendations de cette extension.
- **log** – L'enregistreur. Sauver des messages d'info/débogage etc. en utilisant ceci.

specialize_options(*log*, *opts*, *input_fmt*)

Peut être utilisé pour modifier les valeurs des options de conversion, telles qu'elles sont utilisées par le pipeline de conversion.

specialize_css_for_output(*log*, *opts*, *item*, *stylizer*)

Peut-être utilisé pour effectuer des changements dans le CSS durant le processus d'aplanissement CSS.

Paramètres

- **item** – L'objet (fichier HTML) étant traité

- **stylizer** – Un objet Styliste contenant les styles aplani pour l'objet. Vous pouvez obtenir les styles pour n'importe quel élément par stylizer.style(élément).

gui_configuration_widget (*parent*, *get_option_by_name*, *get_option_help*, *db*, *book_id=None*)

Appelé pour créer un widget utilisé pour configurer cette extension dans le GUI de calibre. Le widget doit être une manipulation de la classe PluginWidget. Voir la création d'extensions d'origines pour des exemples.

12.1.7 Pilotes de périphérique

La classe de base pour tous les pilotes de périphérique est *DevicePlugin* (page 281). Cependant, si votre périphérique expose un port USBMS au système opérateur, vous devriez utiliser la classe USBMS à la place car elle implémente toute la logique nécessaire pour supporter ce genre de périphérique.

class calibre.devices.interface.**DevicePlugin** (*plugin_path*)

Bases : *Plugin* (page 268)

Définit l'interface qui devrait être implémentée par les derniers processus qui communiquent avec le lecteur de livre numérique.

type = 'Interface du périphérique'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

FORMATS = ['lrf', 'rtf', 'pdf', 'txt']

Liste ordonnée de formats supportés

VENDOR_ID = 0

VENDOR_ID peut être soit un entier, une liste d'entiers ou un dictionnaire. Si c'est un dictionnaire, cela doit être un dictionnaire de dictionnaires sous la forme :

```
{
 integer_vendor_id : { product_id : [list of BCDs], ... },
 ...
}
```

PRODUCT_ID = 0

Un entier ou une liste d'entiers

BCD = None

BCD peut être soit None pour ne pas distingué entre les périphériques basé sur BCD, ou il peut être une liste des nombres BCD de tous les périphériques supportés par ce pilote.

THUMBNAIL_HEIGHT = 68

Hauteur de l'aperçu sur le périphérique

THUMBNAIL_COMPRESSION_QUALITY = 75

Qualité de compression pour les vignettes. Paramétrer ceci proche de 100 pour obtenir une meilleure qualité d'étiquette avec moins d'artefacts de compression. Bien sûr, les vignettes grandiront en conséquence.

WANTS_UPDATED_THUMBNAILS = False

Fixer ceci à True si le périphérique supporte les mises à jours d'aperçus de couverture pendant sync_booklists. Le fixer à True demandera à device.py de rafraîchir les aperçus de couverture pendant la correspondance de livre.

CAN_SET_METADATA = ['title', 'authors', 'collections']

Soit les métadonnées des livres sont fixées via le GUI.

CAN_DO_DEVICE_DB_PLUGBOARD = False

Soit le périphérique peut gérer tableau de métadonnée device_db

path_sep = '/'

Le séparateur de chemin pour les chemins vers les livres sur le périphérique

icon = 'reader.png'

Icône de ce périphérique

UserAnnotation

alias de `Annotation`

OPEN_FEEDBACK_MESSAGE = None

Le GUI affiche ceci en tant que message dans la barre d'état si ce n'est pas None. Utile si l'ouverture peut prendre du temps

VIRTUAL_BOOK_EXTENSIONS = frozenset({})

Ensemble d'extensions qui sont des « livres virtuels » sur le périphérique et par conséquent ne peuvent être vus/sauvés/ajoutés à la bibliothèque. Par exemple : `frozenset(['kobo'])`

VIRTUAL_BOOK_EXTENSION_MESSAGE = None

Message à afficher à l'utilisateur pour les extensions de livre virtuel

NUKE_COMMENTS = None

S'il faut détruire les commentaires dans la copie du livre envoyé au périphérique . Si ce n'est pas None ce devrait être une courte chaîne qui remplacera le commentaire.

MANAGES_DEVICE_PRESENCE = False

Si True, indique que ce pilote gère entièrement la détection de périphérique, l'éjection etc. Si vous réglez à True, vous *devez* implémenter les méthodes detect_managed_devices et debug_managed_device_detection. Un pilote avec ceci réglé à True est responsable de la détection des périphériques, de la gestion d'une liste d'exclusion des périphérique, une liste de périphérique éjectés, etc. calibre utilisera périodiquement la méthode detect_managed_devices() et si elle retourne un périphérique détecté, calibre appellera open(). open() sera appelé à chaque fois qu'un périphérique est renvoyé même si des appels précédents à open() ont échoués, donc le pilote doit maintenir sa propre liste d'exclusion de périphériques qui ont échoués. Similairement, à l'éjection, calibre utilisera eject() et assume alors que le prochain appel à detect_managed_devices() renvoye None, il appellera post_yank_cleanup().

SLOW_DRIVEINFO = False

Si fixer à True, calibre utilisera la méthode *get_driveinfo()* (page 284) après que la liste de livre a été chargée pour obtenire le driveinfo.

ASK_TO_ALLOW_CONNECT = False

Si fixer à True, calibre demandera à l'utilisateur s'il veut gérer le périphérique avec calibre, la première fois qu'il est détecté. Si vous fixez ceci à True, vous devez implémenter *get_device_uid()* (page 287) et *ignore_connected_device()* (page 287) et *get_user_blacklisted_devices()* (page 287) et *set_user_blacklisted_devices()* (page 287)

user_feedback_after_callback = None

Fixer ceci à un dictionnaire de la forme {"title" :title, "msg" :msg, "det_msg" :detailed_msg} pour que calibre fasse surgir un message à l'utilisateur après que quelques rappels aient été fait (actuellement uniquement upload_books). Il faut faire attention à ne pas noyer l'utilisateur par trop de messages. Cette variable est vérifiée après *chaque* rappel, donc utilisez la uniquement lorsque c'est réellement nécessaire.

classmethod get_open_popup_message()

L'interface graphique affiche ceci comme une popup non modale. Devrait être une instance de OpenPopupMessage

is_usb_connected(*devices_on_system*, *debug=False*, *only_presence=False*)

Renvoye True , device_info si un périphérique géré par cette extension est actuellement connecté.

Paramètres

devices_on_system – Liste des périphériques actuellement connectés

detect_managed_devices(*devices_on_system*, *force_refresh=False*)

Appelé uniquement si MANAGES_DEVICE_PRESENCE est True.

Cherche des périphérique que ce pilote peut gérer. Devrait renvoyer un objet périphérique si un périphérique est trouvé. Cet objet sera passé à la méthode open() en tant que connected_device. Si aucun périphérique n'est trouvé, renvoie None. L'objet renvoyé peut être n'importe quoi, calibre ne l'utilise pas, il est uniquement passé dans open().

Cette méthode est appelée périodiquement par le GUI, donc soyez sûr qu'il n'utilise pas trop de ressource. Utilisez un cache pour pour éviter de scanner répétitivement le système.

Paramètres

- **devices_on_system** – Ensemble de périphériques USB trouvés sur le système.
- **force_refresh** – Si True et que le pilote utilise un cache pour prévenir les scan répétés, le cache doit être flushé.

debug_managed_device_detection(*devices_on_system*, *output*)

Appelé uniquement si MANAGES_DEVICE_PRESENCE est True.

Devrait écrire des informations sur les périphériques détectés par le système pour être affiché, qui devrait être un objet ressemblant à un fichier.

Devrait renvoyer True si un périphérique était détecter et effectivement ouvert, sinon False.

reset(*key='-1'*, *log_packets=False*, *report_progress=None*, *detected_device=None*)

Paramètres

- **key** – La clef pour débloquer le périphérique
- **log_packets** – Si True, le paquet envoyé ou reçu au périphérique est mis au registre
- **report_progress** – Une fonction qui est appelée avec un % de progression (un nombre en 0 et 100) pour différentes tâches. Si on utilise -1 cela veut dire que la tâche n'a aucune information de progression
- **detected_device** – Les informations de périphérique recueilient par l'analyse de périphérique

can_handle_windows(*usbdevice*, *debug=False*)

Une méthode optionnelle pour effectuer de plus amples vérifications sur un périphérique pour savoir si ce pilote est capable de le gérer. Sinon, cela devrait renvoyer False. Cette méthode est appelée uniquement après que le vendeur, l'identifiant du produit et le bcd sont appariés, donc elle peut faire des vérifications nécessitant pas mal de temps. L'implémentation par défaut renvoye True. Cette méthode est appelée uniquement sur Windows. voir aussi *can_handle()* (page 283).

Notez que pour les périphériques basés sur USBMS cette méthode par défaut délègue à *can_handle()* (page 283). Donc vous avez seulement besoin d'outrepasser *can_handle()* (page 283) dans votre sous-classe de USBMS.

Paramètres

usbdevice – Un périphérique USB est rapporté par `calibre.devices.winusb.scan_usb_devices()`

can_handle(*device_info*, *debug=False*)

Version Unix de *can_handle_windows()* (page 283).

Paramètres
device_info – Est un tuple de (vid, pid, bcd, manufacturer, product, serial number)

open(*connected_device*, *library_uuid*)

Effectue n'importe quel initialisation spécifique de périphérique. Appelé après que le périphérique soit détecté mais avant que n'importe quelle fonction qui communique avec le périphérique. Par exemple : Pour les périphériques qui se présentent comme des périphériques de stockage de masse USB, cette méthode devrait être responsable du montage du périphérique ou si le périphérique a été monté automatiquement, de trouver où il a été monté. La méthode *calibre.devices.usbms.device.Device.open()* (page 292) a une implémentation de cette fonction qui devrait servir comme un bon exemple pour les périphériques de stockage de masse USB.

Cette méthode peut créer une exception OpenFeedback pour afficher un message à l'utilisateur.

Paramètres

- **connected_device** – Le périphérique que nous essayons actuellement d'ouvrir. C'est un tuple (vendor id, product id, bcd, manufacturer name, product name, device serial number). Toutefois, certains périphériques n'on aucun numéro de séries et sur Windows seul les trois premier champs sont présent, les autres sont None
- **library_uuid** – L'UUID de la bibliothèque calibre actuelle. Peut être None s'il n'y a pas de bibliohèque (par exemple lorsqu'il est utilisé de l'invité de commande).

eject()

Démonte/éjecte le périphérique de l'OS. Ceci ne vérifie pas s'il y a des travaux GUI en attente qui nécessite communication avec le périphérique.

NOTEZ : Que cette méthode peut ne pas être appelée sur le même fil que le reste des méthodes périphériques.

post_yank_cleanup()

Appelé si l'utilisateur retire le périphériques sans l'avoir éjecter préalablement.

set_progress_reporter(*report_progress*)

Pose une fonction pour rapporter les information de progression.

Paramètres
report_progress – Une fonction qui est appelée avec un % de progression (un nombre en 0 et 100) pour différentes tâches. Si on utilise -1 cela veut dire que la tâche n'a aucune information de progression

get_device_information(*end_session=True*)

Demande ses informations au périphérique. Voir L{DeviceInfoQuery}.

Renvoie
(nom du périphérique, version du périphérique, version du programme sur le périphérique, type MIME) Le tuple peut parfois avoir un cinquième élément, qui est dictionnaire d'information sur le lecteur. Voir usbms.driver pour un exemple.

get_driveinfo()

Renvoie le dictionnaire driveinfo. Habituellement appélé de get_device_information(), mais si le chargement du driveinfo est lent pour ce lecteur, alors il devrait fixer SLOW_DRIVEINFO. Dans ce cas, cette méthode sera appelée par calibre après que la liste de livre ait été chargée. Notez que ceci n'est pas appelé sur le fil du périphérique, donc le pilote devrait cacher les informations du pilote dans la méthode books() et cette fonction devrait renvoyer les données cachées.

card_prefix(*end_session=True*)

Renvoie une liste de 2 éléments des préfixes vers les chemins sur les cartes. Si aucune carte n'est présent, None est fixé comme le préfixe de la carte. e.g. ("/place", "/place2") (None, "place2") ("place", None) (None, None)

total_space(*end_session=True*)

Récupère l'espace disponible sur les points de montages :

1. Mémoire principale
2. Carte Mémoire A
3. Carte Mémoire B

Renvoie
Une liste à 3 éléments avec des espaces entier en bit de (1, 2, 3). Si un périphérique particulier n'a pas l'une de ces locations, il devrait retourner à 0.

free_space(*end_session=True*)

Récupère l'espace disponible sur les points de montage :

1. Mémoire principale
2. Carte A
3. Carte B

Renvoie
Une liste à 3 éléments avec des espaces entier en bit de (1, 2, 3). Si un périphérique particulier n'a pas l'une de ces locations, il devrait retourner à -1.

books(*oncard=None*, *end_session=True*)

Renvoie une liste des livres numérique sur le périphérique.

Paramètres
oncard – Si "carda" ou "cardb", renvoie une liste de livres numériques sur la carte de stockage spécifique, sinon renvoie une liste de livres numériques dans la mémoire principale du périphérique. Si une carte est spécifiée et qu'aucun livre n'est dans la carte, renvoie une liste vide.

Renvoie
Une BookList

upload_books(*files*, *names*, *on_card=None*, *end_session=True*, *metadata=None*)

Envoie une liste de livres au périphérique. Si un fichier existe déjà sur le périphérique, il devrait être remplacé. Cette méthode devrait soulever une `FreeSpaceError` s'il n'y a pas d'espace sur le périphérique. Le texte dans la FreeSpaceError doit contenir le mot « carte » si « on_card » n'est pas None il doit contenir le mot « mémoire ».

Paramètres

- **files** – Une liste de chemins
- **names** – Une liste de noms que les livres devraient avoir une fois envoyé au périphérique. len(names) == len(files)
- **metadata** – Si n'est pas None, est une liste d'objets `Métadonnées`. L'idée est d'utiliser la métadonnée pour déterminer où mettre le livre sur le périphérique. len(metadata) == len(files). En plus de la couverture habituelle(chemin vers la couverture), il peut aussi y avoir

un attribut d'aperçu, qui devrait être utilisée préférentiellement. L'attribut d'aperçu est de la form (largeur, hauteur, cover_data en jpeg).

Renvoie
Une liste d'ensemble de tuples à 3 éléments. La liste est faite pour être passée à *add_books_to_metadata()* (page 286).

classmethod add_books_to_metadata(*locations*, *metadata*, *booklists*)

Ajoute des emplacements à la liste de livres. cette fonction ne doit pas communiquer avec le périphérique.

Paramètres

- **locations** – Résultat d'un appel à L{upload_books}
- **metadata** – Liste d'objets `Métadonénes`, même chose que *upload_books()* (page 285).
- **booklists** – En tuple contenant le résultat d'appels à (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard="cardb")`).

delete_books(*paths*, *end_session=True*)

Supprime les livre aux chemins sur le périphérique.

classmethod remove_books_from_metadata(*paths*, *booklists*)

Retire les livres de la liste de métadonnées. Cette fonction ne doit pas communiquer avec le périphérique.

Paramètres

- **paths** – les chemins vers les livres sur le périphérique.
- **booklists** – En tuple contenant le résultat d'appels à (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard="cardb")`).

sync_booklists(*booklists*, *end_session=True*)

Met à jour les métadonnées sur le périphérique.

Paramètres
booklists – En tuple contenant le résultat d'appels à (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard="cardb")`).

get_file(*path*, *outfile*, *end_session=True*)

Lit le fichier à `path` sur le périphérique et l'écrit sur un fichier extérieur.

Paramètres
outfile – un objet ressemblant à un fichier `sys.stdout` ou le résultat d'un appel à une *open()* (page 284)

classmethod config_widget()

Devrait renvoyer un QWdiget. Le QWidget contient les paramètres pour l'interface du périphérique.

classmethod save_settings(*settings_widget*)

Devrait sauver les paramètres sur le disque. Prend le widget créer dans *config_widget()* (page 286) et sauve tous les paramètres sur le disque.

classmethod settings()

Devrait renvoyer un objet opts. Les objets opts devrait au moins avoir un attribut *format_map* qui est une liste ordonnée de formats pour le périphérique.

set_plugboards(*plugboards*, *pb_func*)

fournit au pilote le l'ensemble actuel de tableaux de connexion et une fonction pour sélectionner un tableau de connexion spécifique. Cette méthode est appelée immédiatement avant add_books et sync_booklists.

pb_func est appelable avec la signature suivante :
def pb_func(device_name, format, tableaux de connexion)

Vous donnez le nom actuel du périphérique(soit le nom de classe du périphérique soit DEVICE_PLUGBOARD_NAME), le format qui vous intéresse (un "real" format ou "device_db"), et les tableaux de connexion (ceux donnés par set_plugboards, au même endroit qu'où vous avez eu cette méthode).

Renvoie
None ou unique tableau de connexion

set_driveinfo_name(*location_code*, *name*)

Fixe le nom du périphérique dans le driveinfo à "name". Ce Paramètre persistera jusqu'à ce que le fichier soit recréer ou que le nom soit changé à nouveau.

Des périphériques n'étant pas des disques devrait implémenter cette méthode basée sur les codes d'emplacement renvoyé par la méthode get_device_information().

prepare_addable_books(*paths*)

En fonction d'une liste de chemins, renvoie une autre liste de chemins. Ces chemins pointe vers des versions cumulable des livres.

S'il y a une erreur dans la préparation d'un livre, alors au lieu du chemin, la position dans la liste renvoyée pour ce livre devrait être un 3-tuple (original_path, l'exception, retraçage)

startup()

Appelé lorsque calibre démarre le périphérique. Fait toutes les initialisations requises. Notez que de multiples utilisation de la classe peuvent être instanciée, et donc __init__ peut être appelé de nombreuse fois, mais une seule instance utilisera cette méthode. Cette méthode est appelée sur le fil périphérique, pas celui du GUI.

shutdown()

Appelé lorsque calibre s'éteint, soit réellement, soit pour redémarrer. Fais n'importe quel nettoyage nécessaire. Cette méthode est appelée sur le fil périphérique, pas celui du GUI.

get_device_uid()

Doit rendre un identificateur unique pour le périphérique connecté (ceci est appelé immédiatement après un appel réussi à open()). Vous devez implémenter cette méthode si vous fixez ASK_TO_ALLOW_CONNECT = True

ignore_connected_device(*uid*)

Devrait ignorer le périphérique identifié par l'uid (le résultat d'un appel à get_device_uid()) dans le futur. Vous devez implémenter cette méthode si vous avez fixez ASK_TO_ALLOW_CONNECT = True.. Notez que cette fonction est appelée immédiatement après open(), donc si open() cache un état, le pilote devrait réinitialiser cet état.

get_user_blacklisted_devices()

Renvoie un carte d'uid de périphérique à des noms lisibles pour tous les périphériques que l'utilisateur a demandé d'ignorer.

set_user_blacklisted_devices(*devices*)

Fixe la liste d'uids de périphérique qui devraient être ignorés par ce pilote.

specialize_global_preferences(*device_prefs*)

Implémentez cette méthode si votre périphérique veut surpasser une préférence particulière. Vous devez vous assurer que tous les sites d'appel qui veulent la préférence qui peut être outrepassée utilise device_prefs["quelque_chose"] au lieu de prefs["quelque_chose"]. Votre méthode devrait appeler device_prefs.set_overrides(pref=val, pref=val, …). Acutellement utilisé pour : gestion de métadonnée (prefs["manage_device_metadata"])

set_library_info(*library_name*, *library_uuid*, *field_metadata*)

Implémentez cette méthode si vous voulez des informations sur la bibliothèque calibre actuelle. Cette méthode est appelée au démarrage et lorsque la bibliothèque calibre est changée un fois connecté.

is_dynamically_controllable()

Appelée par le gestionnaire de périphérique au lancement des extensions. Si cette méthode renvoie une chaîne, alors a) il supporte le contrôle dynamique d'interface du gestionnaire de périphérique, et b) ce nom doit être utiliser lorsqu'on communique avec l'extension.

Cette méthode peut être appelée dans le fil GUI. Un pilote qui implémente cette méthode doit être sûr du fil.

start_plugin()

Cette méthode est appelée au lancement de l'extension. Cette extension devrait commencer à accepter des connexions de périphériques qu'importe la manière dont ils le font. Si l'extension accepte déjà les connexions, alors ne fait rien.

Cette méthode peut être appelée dans le fil GUI. Un pilote qui implémente cette méthode doit être sûr du fil.

stop_plugin()

Cette méthode est appelée pour arrêter l'extension. Cette extension ne devrait plus accepter de connections et devrait nettoyer derrière lui. Il est probable que cette méthode appel une éteinte. Si cette extension n'accepte déjà plus de connections, alors ne fait rien.

Cette méthode peut être appelée dans le fil GUI. Un pilote qui implémente cette méthode doit être sûr du fil.

get_option(*opt_string*, *default=None*)

Renvoie la valeur de l'option indiquée par opt_string. Cette méthode peut être appelée lorsque l'extension n'est pas lancée. Renvoie None si cette option n'existe pas.

Cette méthode peut être appelée dans le fil GUI. Un pilote qui implémente cette méthode doit être sûr du fil.

set_option(*opt_string*, *opt_value*)

Fixe la valeur de l'option indiquée par opt_string. Cette méthode peut être appelée lorsque l'extension n'est pas lancée.

Cette méthode peut être appelée dans le fil GUI. Un pilote qui implémente cette méthode doit être sûr du fil.

is_running()

Renvoie True si l'extension est lancée, sinon False

Cette méthode peut être appelée dans le fil GUI. Un pilote qui implémente cette méthode doit être sûr du fil.

synchronize_with_db(*db*, *book_id*, *book_metadata*, *first_call*)

Appelé durant la correspondance de livre lorsqu'un livre sur un périphérique est appareillé avec une livre de la bdd de calibre. La méthode est responsable pour synchroniser les données du périphérique vers la bdd de calibre (si nécessaire).

La méthode doit renvoyer un tuple à deux valeurs. La première valeur est un ensemble des identifiants de livre calibre changés si la base de donnée calibre a été changée ou None si la base de donnée n'a pas été changée. Si la première valeur est un ensemble vide alors les métadonnée pour le livre sur le périphérique sont mises à jour avec les métadonnées de calibre et renvoyée au périphérique, mais aucun rafraîchissement GUI de ce livre n'est effectué. Ceci est pratique lorsque les données de calibre sont justes mais qu'elles doivent être envoyées au périphérique.

La seconde valeur est elle-même un tuple de 2 valeurs. La première valeur dans le tuple spécifie si un format du livre devrait être envoyé au périphérique. L'idée est de permettre de vérifier que le livre sur le périphérique est le même que le livre dans calibre. Cette valeur doit être None si aucun livre ne doit être envoyé, sinon renvoie le nom du fichier de base sur le périphérique (une chaîne comme foobar.epub). Soyez sûr d'inclure l'extension dans le nom. Le sous système du périphérique construira une tâche send_books pour tous les livres qui n'ont pas une valeur None . Note : à part pour récupérer l'extension par après, le nom est ignoré dans le

cas où le périphérique utilise un schéma pour générer le nom de fichier, ce que la plupart font. La seconde valeur dans le tuple renvoyé indique si le format est prévu. Renvoie True s'il l'est, sinon renvoie False. calibre affichera une boîte de dialogue à l'utilisateur pour afficher tous les livres prévus.

Extrêmement important : cette méthode est appelée sur le fil GUI. Elle doit être en accord avec le fil du gestionnaire de périphérique.

book_id : l'identificateur calibre pour le livre dans la base de données. book_metadata : la métadonnée pour le livre venant du périphérique. first_call : True si c'est le premier appel durant une synchronisation, sinon False.

class calibre.devices.interface.**BookList**(*oncard*, *prefix*, *settings*)

Bases : list

Un liste de livres. Chaque livre doit avoir les champs

1. titre
2. auteurs
3. taille (taille du fichier du livre)
4. date (un tuple temps UTC)
5. chemin (chemin vers le livre sur le périphérique)
6. aperçu (peut être None) l'aperçu est soit un str/bytes avec les données de l'image ou il devrait avoir un attribut image_path qui contient un chemin absolu (dépendant de la plateforme) vers le livre
7. identifiant (une liste de chaîne, peut être vide).

supports_collections()

Renvoie True si le périphérique supporte les collections de livre pour cette liste de livre.

add_book(*book*, *replace_metadata*)

Ajoute le livre dans la liste de livre. L'idée est de maintenir les métadonnées interne du périphérique. Renvoie True si la liste de livre doit être synchronisée.

remove_book(*book*)

Supprime un livre de la liste de livre. Corrige toutes les métadonnées sur le périphérique en même temps

get_collections(*collection_attributes*)

Renvoie un dictionnaire de collections créée avec collection_attributes. Chaque entrée dans le dictionnaire est sous la forme nom de collection :[liste de livres]

La liste de livre est triée par titre, sauf pour les collections créée à partir de séries, auquel cas series_index est utilisé.

Paramètres

collection_attributes – Un liste d'attribut du livre

Périphérique basé sur du stockage de masse USB

La classe de base pour ces périphériques est *calibre.devices.usbms.driver.USBMS* (page 293). Cette classe hérite certaines de ces fonctionnalité de ces bases, documentée plus bas. Un pilote basique typique USBMS (USB Mass Storage) ressemble à ceci :

```
from calibre.devices.usbms.driver import USBMS

class PDNOVEL(USBMS):
    name = 'Pandigital Novel device interface'
    gui_name = 'PD Novel'
    description = _('Communicate with the Pandigital Novel')
    author = 'Kovid Goyal'
    supported_platforms = ['windows', 'linux', 'osx']
    FORMATS = ['epub', 'pdf']

    VENDOR_ID   = [0x18d1]
    PRODUCT_ID  = [0xb004]
    BCD         = [0x224]

    THUMBNAIL_HEIGHT = 144

    EBOOK_DIR_MAIN = 'eBooks'
    SUPPORTS_SUB_DIRS = False

    def upload_cover(self, path, filename, metadata):
        coverdata = getattr(metadata, 'thumbnail', None)
        if coverdata and coverdata[2]:
            with open('%s.jpg' % os.path.join(path, filename), 'wb') as coverfile:
                coverfile.write(coverdata[2])
```

class calibre.devices.usbms.device.**Device**(*plugin_path*)

Bases : DeviceConfig, *DevicePlugin* (page 281)

Cette classe fournit une logique commune à tous les pilotes pour les périphériques qui s'exportent comme des périphériques de stockage de masse USB. Fournit des implémentation pour monter/éjecter un périphérique USBMS sur toute les plateformes.

VENDOR_ID = 0

VENDOR_ID peut être soit un entier, une liste d'entiers ou un dictionnaire. Si c'est un dictionnaire, cela doit être un dictionnaire de dictionnaires sous la forme :

```
{
 integer_vendor_id : { product_id : [list of BCDs], ... },
 ...
}
```

PRODUCT_ID = 0

Un entier ou une liste d'entiers

BCD = None

BCD peut être soit None pour ne pas distingué entre les périphériques basé sur BCD, ou il peut être une liste des nombres BCD de tous les périphériques supportés par ce pilote.

WINDOWS_MAIN_MEM = None

Une chaîne identifiant la mémoire principale du périphérique dans les chaines d'identifications PnP de Windows. Ceci peut être None, une chaîne, une liste de chaîne ou une expression régulière compilée

WINDOWS_CARD_A_MEM = None

Une chaîne identifiant la première carte du périphérique dans les chaines d'identifications PnP de Windows. Ceci peut être None, une chaîne, une liste de chaîne ou une expression régulière compilée

WINDOWS_CARD_B_MEM = None

Une chaîne identifiant la deuxième carte du périphérique dans les chaines d'identifications PnP de Windows. Ceci peut être None, une chaîne, une liste de chaîne ou une expression régulière compilée

OSX_MAIN_MEM_VOL_PAT = None

Utilisé par la détection du nouveau pilote pour dissocier la mémoire principale des cartes mémoires. Devrait-être une expression régulière qui correspond au point de montage de la mémoire principale assignée par macOS

BACKLOADING_ERROR_MESSAGE = None

MAX_PATH_LEN = 250

La longueur maximum des chemins créés sur le périphérique

NEWS_IN_FOLDER = True

Place les nouvelles dans son propre fichier

reset(*key='-1'*, *log_packets=False*, *report_progress=None*, *detected_device=None*)

Paramètres

- **key** – La clef pour débloquer le périphérique
- **log_packets** – Si True, le paquet envoyé ou reçu au périphérique est mis au registre
- **report_progress** – Une fonction qui est appelée avec un % de progression (un nombre en 0 et 100) pour différentes tâches. Si on utilise -1 cela veut dire que la tâche n'a aucune information de progression
- **detected_device** – Les informations de périphérique recueilient par l'analyse de périphérique

set_progress_reporter(*report_progress*)

Pose une fonction pour rapporter les information de progression.

Paramètres

report_progress – Une fonction qui est appelée avec un % de progression (un nombre en 0 et 100) pour différentes tâches. Si on utilise -1 cela veut dire que la tâche n'a aucune information de progression

card_prefix(*end_session=True*)

Renvoie une liste de 2 éléments des préfixes vers les chemins sur les cartes. Si aucune carte n'est présent, None est fixé comme le préfixe de la carte. e.g. ("/place", "/place2") (None, "place2") ("place", None) (None, None)

total_space(*end_session=True*)

Récupère l'espace disponible sur les points de montages :

1. Mémoire principale
2. Carte Mémoire A
3. Carte Mémoire B

Renvoie

Une liste à 3 éléments avec des espaces entier en bit de (1, 2, 3). Si un périphérique particulier n'a pas l'une de ces locations, il devrait retourner à 0.

free_space(*end_session=True*)

Récupère l'espace disponible sur les points de montage :

1. Mémoire principale
2. Carte A
3. Carte B

Renvoie
Une liste à 3 éléments avec des espaces entier en bit de (1, 2, 3). Si un périphérique particulier n'a pas l'une de ces locations, il devrait retourner à -1.

windows_sort_drives(*drives*)

Appelée pour différencier la mémoire principale et les cartes mémoires pour les périphériques qui ne peuvent les différencier sur base de *WINDOWS_CARD_NAME*. par ex. : L'EB600

can_handle_windows(*usbdevice*, *debug=False*)

Une méthode optionnelle pour effectuer de plus amples vérifications sur un périphérique pour savoir si ce pilote est capable de le gérer. Sinon, cela devrait renvoyer False. Cette méthode est appelée uniquement après que le vendeur, l'identifiant du produit et le bcd sont appariés, donc elle peut faire des vérifications nécessitant pas mal de temps. L'implémentation par défaut renvoye True. Cette méthode est appelée uniquement sur Windows. voir aussi `can_handle()`.

Notez que pour les périphériques basés sur USBMS cette méthode par défaut délègue à `can_handle()`. Donc vous avez seulement besoin d'outrepasser `can_handle()` dans votre sous-classe de USBMS.

Paramètres
usbdevice – Un périphérique USB est rapporté par `calibre.devices.winusb.scan_usb_devices()`

open(*connected_device*, *library_uuid*)

Effectue n'importe quel initialisation spécifique de périphérique. Appelé après que le périphérique soit détecté mais avant que n'importe quelle fonction qui communique avec le périphérique. Par exemple : Pour les périphériques qui se présentent comme des périphériques de stockage de masse USB, cette méthode devrait être responsable du montage du périphérique ou si le périphérique a été monté automatiquement, de trouver où il a été monté. La méthode *calibre.devices.usbms.device.Device.open()* (page 292) a une implémentation de cette fonction qui devrait servir comme un bon exemple pour les périphériques de stockage de masse USB.

Cette méthode peut créer une exception OpenFeedback pour afficher un message à l'utilisateur.

Paramètres

- **connected_device** – Le périphérique que nous essayons actuellement d'ouvrir. C'est un tuple (vendor id, product id, bcd, manufacturer name, product name, device serial number). Toutefois, certains périphériques n'on aucun numéro de séries et sur Windows seul les trois premier champs sont présent, les autres sont None
- **library_uuid** – L'UUID de la bibliothèque calibre actuelle. Peut être None s'il n'y a pas de bibliohèque (par exemple lorsqu'il est utilisé de l'invité de commande).

eject()

Démonte/éjecte le périphérique de l'OS. Ceci ne vérifie pas s'il y a des travaux GUI en attente qui nécessite communication avec le périphérique.

NOTEZ : Que cette méthode peut ne pas être appelée sur le même fil que le reste des méthodes périphériques.

post_yank_cleanup()

Appelé si l'utilisateur retire le périphériques sans l'avoir éjecter préalablement.

sanitize_callback(*path*)

Rappel pour permettre aux pilotes individuels de périphérique d'outrepasser l'expurgation des chemins utilisé par `create_upload_path()`.

filename_callback(*default*, *mi*)

Rappel pour permettre aux pilotes de changer le nom de fichier par défaut fixé par `create_upload_path()`.

sanitize_path_components(*components*)

Effectue n'importe quel expurgation de chemins spécifique au périphérique pour les fichiers devant être envoyés au périphérique

get_annotations(*path_map*)

Résous path_map pour annotation_map pour les fichier trouvés sur le périphérique

add_annotation_to_library(*db*, *db_id*, *annotation*)

Ajoute une annotation dans la bibliothèque calibre

class calibre.devices.usbms.cli.**CLI**

class calibre.devices.usbms.driver.**USBMS**(*plugin_path*)

Bases : *CLI* (page 293), *Device* (page 290)

Les classes de base pour tous les périphériques USBMS. Implémente la logique d'envoi/réception/màj/métadonnée/encachement de métadonnée/etc.

description = 'Communiquer avec une liseuse de livre numérique.'

Une courte chaine décrivant ce que fait l'extension

author = 'John Schember'

L'auteur de cette extension

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

booklist_class

alias de `BookList`

book_class

alias de `Book`

FORMATS = []

Liste ordonnée de formats supportés

CAN_SET_METADATA = []

Soit les métadonnées des livres sont fixées via le GUI.

get_device_information(*end_session=True*)

Demande ses informations au périphérique. Voir L{DeviceInfoQuery}.

Renvoie

(nom du périphérique, version du périphérique, version du programme sur le périphérique, type MIME) Le tuple peut parfois avoir un cinquième élément, qui est dictionnaire d'information sur le lecteur. Voir usbms.driver pour un exemple.

set_driveinfo_name (*location_code*, *name*)

Fixe le nom du périphérique dans le driveinfo à “name”. Ce Paramètre persistera jusqu'à ce que le fichier soit recréer ou que le nom soit changé à nouveau.

Des périphériques n'étant pas des disques devrait implémenter cette méthode basée sur les codes d'emplacement renvoyé par la méthode get_device_information().

books (*oncard=None*, *end_session=True*)

Renvoie une liste des livres numérique sur le périphérique.

Paramètres

oncard – Si “carda” ou “cardb”, renvoie une liste de livres numériques sur la carte de stockage spécifique, sinon renvoie une liste de livres numériques dans la mémoire principale du périphérique. Si une carte est spécifiée et qu'aucun livre n'est dans la carte, renvoie une liste vide.

Renvoie

Une BookList

upload_books (*files*, *names*, *on_card=None*, *end_session=True*, *metadata=None*)

Envoie une liste de livres au périphérique. Si un fichier existe déjà sur le périphérique, il devrait être remplacé. Cette méthode devrait soulever une `FreeSpaceError` s'il n'y a pas d'espace sur le périphérique. Le texte dans la FreeSpaceError doit contenir le mot « carte » si « on_card » n'est pas None il doit contenir le mot « mémoire ».

Paramètres

- **files** – Une liste de chemins
- **names** – Une liste de noms que les livres devraient avoir une fois envoyé au périphérique. len(names) == len(files)
- **metadata** – Si n'est pas None, est une liste d'objets `Métadonnées`. L'idée est d'utiliser la métadonnée pour déterminer où mettre le livre sur le périphérique. len(metadata) == len(files). En plus de la couverture habituelle(chemin vers la couverture), il peut aussi y avoir un attribut d'aperçu, qui devrait être utilisée préférentiellement. L'attribut d'aperçu est de la form (largeur, hauteur, cover_data en jpeg).

Renvoie

Une liste d'ensemble de tuples à 3 éléments. La liste est faite pour être passée à *add_books_to_metadata()* (page 294).

upload_cover (*path*, *filename*, *metadata*, *filepath*)

Envoie les couvertures de livre au périphérique. L'implémentation par défaut ne fait rien.

Paramètres

- **path** – Le chemin complet vers le dossier où se trouve le livre associé.
- **filename** – Le nom du fichier du livre sans l'extension.
- **metadata** – métadonnée appartenant au livre. Utilisez metadata.thumbnail pour les couvertures
- **filepath** – Le chemin absolu vers le livre numérique

add_books_to_metadata (*locations*, *metadata*, *booklists*)

Ajoute des emplacements à la liste de livres. cette fonction ne doit pas communiquer avec le périphérique.

Paramètres

- **locations** – Résultat d'un appel à L{upload_books}

- **metadata** – Liste d'objets `Métadonénes`, même chose que *upload_books()* (page 294).
- **booklists** – En tuple contenant le résultat d'appels à (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard="cardb")`).

delete_books (*paths*, *end_session=True*)

Supprime les livre aux chemins sur le périphérique.

remove_books_from_metadata (*paths*, *booklists*)

Retire les livres de la liste de métadonnées. Cette fonction ne doit pas communiquer avec le périphérique.

Paramètres

- **paths** – les chemins vers les livres sur le périphérique.
- **booklists** – En tuple contenant le résultat d'appels à (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard="cardb")`).

sync_booklists (*booklists*, *end_session=True*)

Met à jour les métadonnées sur le périphérique.

Paramètres

booklists – En tuple contenant le résultat d'appels à (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard="cardb")`).

classmethod normalize_path (*path*)

Renvoie le chemin avec les séparateurs de chemin natif à la platforme

12.1.8 Actions interface utilisateur

Si vous ajoutez votre propre extension dans un fichier ZIP, vous devriez sous-classer InterfaceActionBase et InterfaceAction. La méthode `load_actual_plugin()` de votre sous-classe InterfaceActionBase doit renvoyer un objet instancié de votre sous-classe InterfaceBase.

class `calibre.gui2.actions.`**InterfaceAction** (*parent*, *site_customization*)

Bases : `QObject`

Une extension représentant une « action » qui peut être effectuée dans le GUI. Tous les menus de la barre d'outil et les menus contextuels sont implémentés par ces extensions.

Notez que cette classe est la classe de base pour ces extensions, toutefois, pour intégrer l'extension avec le système d'extension de calibre, vous devez avoir une classe englobante qui référence la vrai extension.Voir le module `calibre.customize.builtins` pour des exemples.

Si deux objets *InterfaceAction* (page 295) ont le même nom, celui avec la priorité la plus haute à préséance.

Les sous-classes devrait implémenter les méthodes *library_changed()* (page 298), *location_selected()* (page 298), *shutting_down()* (page 298), *initialization_complete()* (page 298) et *tag_browser_context_action()* (page 298).

Une fois initialisé, cette extension à accès au Gui principal de calibre via le membre `gui`. Vous pouvez accéder d'autres extensions par nom, par exemple :

```
self.gui.iactions['Save To Disk']
```

Pour accéder à la réelle extension, utiliser l'attribut `interface_action_base_plugin`, cet attribut devient disponible uniquement après que l'extension ait été initialisée. Utile si vous souhaitez utiliser des méthodes de la classe extension comme do_user_config().

La QAction spécifiée par *action_spec* (page 296) est automatiquement créée et rendue disponible sous `self.qaction`.

name = 'Implement me'

Le nom de l'extension. Si deux extensions avec le même nom sont présentes, celle avec la plus haute priorité à préséance.

priority = 1

La priorité de l'extension. Si deux extensions avec le même nom sont présentes, celle avec la plus haute priorité a préséance.

popup_type = 1

Le type de menu contextuel pour lorsque cette extension est ajoutée à la barre d'outils

auto_repeat = False

Si cette action devrait être répétée automatiquement lorsque son raccourci est maintenant enfoncé.

action_spec = ('text', 'icon', None, None)

De la forme : (texte, icon_path, infobulle,raccourcis clavier) icône, infobulle et raccourcis clavier peuvent être None. Le raccourcis doit être une chaîne, None ou un tuple de raccourcis. Si None, aucune touche n'est enregistrée pour correspondre à l'action. Si vous entrez un tuple vide, alors le raccourcis est enregistré sans attache spécifique à une touche.

action_shortcut_name = None

Si ce n'est pas None, utilisé pour le nom affiché de l'utilisateur lors de la personnalisation des raccourcis clavier pour la spécification d'action ci-dessus au lieu de action_spec[0]

action_add_menu = False

Si True, un menu est automatiquement créé et ajouté à self.qaction

action_menu_clone_qaction = False

Si True, un clône de self.qaction est ajouté au menu de self.qaction. Si vous souhaitez que le texte de cette action soit différent de self.qaction, fixer cette variable aux nouveau texte

dont_add_to = frozenset({})

Ensemble d'emplacements vers lesquels cette action ne doit pas être ajoutée. Voir `all_locations` pour une liste des emplacements possible

dont_remove_from = frozenset({})

Ensemble d'emplacements d'où cette action ne doit pas être supprimé. Voir `all_locations` pour une liste des emplacements possible

action_type = 'global'

Type d'action. "current" signifie qu'il agit sur la vue courante, "global" représente une action qui ne change pas la vue courante, mais plutôt calibre tout entier

accepts_drops = False

Si True, alors cette InterfaceAction aura l'opportunité d'interagir avec les événements glisser-déposer. Voir les méthodes *accept_enter_event()* (page 296), :meth` :accept_drag_move_event`, *drop_event()* (page 296) pour plus de détails.

accept_enter_event(*event*, *mime_data*)

Cette méthode devrait renvoyer True ssi cette action d'interface est capable de gérer les événements glisser. Ne pas appeler accepter/ignorer sur l'événement, ceci sera géré par l'UI de calibre.

accept_drag_move_event(*event*, *mime_data*)

Cette méthode devrait renvoyer True ssi cette action d'interface est capable de gérer les événements glisser. Ne pas appeler accepter/ignorer sur l'événement, ceci sera géré par l'UI de calibre.

drop_event(*event*, *mime_data*)

Cette méthode devrait effectuer des actions utiles et renvoyer True ssi cette action interface est capable de gérer les événement déposer. Ne pas appeler accepter/ignorer sur cet événement, cela sera géré par l'UI de calibre. Vous ne devriez pas effectuer d'opérations longues/bloquantes avec cette fonction. À la place, émettez un signal ou utiliser un QTimer.singleShot et renvoyez rapidement. Voir les actions intégrées pour des exemples.

create_menu_action(*menu*, *unique_name*, *text*, *icon=None*, *shortcut=None*, *description=None*, *triggered=None*, *shortcut_name=None*, *persist_shortcut=False*)

Une méthode commode pour ajouter facilement des actions à un QMenu. Renvoie la QAction créée. Cette action a un attribut de plus, calibre_shortcut_unique_name qui s'il n'est pas None réfère à un nom unique sous lequel cette action est enregistrée dans le gestionnaire du clavier.

Paramètres

- **menu** – Le QMenu auquel la nouvelle action créée sera ajouté
- **unique_name** – Un nom unique pour cette action, ceci doit être globalement unique, donc faites en sorte qu'il soit le plus descriptif possible. Dans le doute, ajoutez-y un UUID
- **text** – Le texte de l'action.
- **icon** – Soit une QIcon ou un nom de fichier. Le nom de fichier est passé aux QIcon.ic() intégré, donc vous ne devez pas passer le chemin absolu vers le dossier d'images.
- **shortcut** – Une chaîne, un ensemble de chaînes, None ou False. Si False, aucun raccourci clavier ne sera enregistré pour cette action. Si None, un raccourci clavier sans touche par défaut est enregistré. La chaîne et la liste de chaîne enregistre un raccourci avec une touche par défaut comme spécifié.
- **description** – Une description pour cette action. Utilisée pour fixer les infobulles.
- **triggered** – Un appelable qui est connecté au signal déclenché de l'action créée.
- **shortcut_name** – Le texte affiché à l'utilisateur lors de la personnalisation des raccourcis clavier pour cette action. Par défaut il est fixer à une valeur `text`.
- **persist_shortcut** – Les raccourcis pour les actions qui n'apparaissent pas toujours, ou qui dépendent de la bibliothèque, peuvent disparaître lorsque d'autres raccourcis clavier sont édités, sauf si « Persist_shortcut » est réglé sur True.

load_resources(*names*)

Si cette extension est donnée dans un fichier ZIP (extension ajoutée par l'utilisateur), cette méthode vous permettra de charger des ressources du fichier ZIP.

Par exemple pour charger une image

```
pixmap = QPixmap()
pixmap.loadFromData(tuple(self.load_resources(['images/icon.png']).
↪values())[0])
icon = QIcon(pixmap)
```

Paramètres

names – Liste de chemins vers les ressources dans le ZIP utilisant / comme séparateur

Renvoie

Un dictionnaire sous la forme `{name : file_contents}`. Tout nom qui n'est pas trouvé dans le fichier ZIP ne sera pas présent dans le dictionnaire.

genesis()

Paramètre cette extension. Uniquement appelé une fois lors de l'initialisation. self.gui est disponible. L'action spécifiée par *action_spec* (page 296) est disponible à `self.qaction`.

location_selected(*loc*)

Appelé dès que la liste de livre étant affichée dans calibre change. Actuellement, les valeur pour loc sont : `library, main, card and cardb`.

Cette méthode devrait activer/désactiver cette action et ces sous-actions en fonction de l'emplacement

library_about_to_change(*olddb*, *db*)

Appelé dès que la bibliothèque actuelle est changée.

Paramètres

- **olddb** – La LibraryDatabase correspondant à la bibliothèque précédente
- **db** – La LibraryDatabase correspondant à la nouvelle bibliothèque.

library_changed(*db*)

Appelé dès que la bibliothèque actuelle est changée.

Paramètres
db – La LibraryDatabase correspondant à la bibliothèque actuelle

gui_layout_complete()

Appelée une fois par action lorsque l'affichage du Gui principal est complété. Si votre action nécessite de faire des changements dans l'affichage, elles devraient être faite ici, plutôt que dans *initialization_complete()* (page 298).

initialization_complete()

Appelée un fois par action lorsque l'initiation du GUI principal est complète.

tag_browser_context_action(*index*)

Appelé lors de l'affichage du menu contextuel dans le Navigateur d'étiquettes. `index` est le QModelIndex qui pointe vers l'élément du Navigateur d'étiquettes qui a été cliqué avec le bouton droit. Testez sa validité avec index.valid() et obtenez l'objet TagTreeItem sous-jacent avec index.data(Qt.ItemDataRole.UserRole). Tout objet d'action produit par cette méthode sera ajouté au menu contextuel.

shutting_down()

Appelée une fois par extension lorsque le GUI principal est en train de s'éteindre. Relâche toute les ressources utilisée, mais essaye de ne pas bloquer l'arrêt pour un longue période.

class calibre.customize.**InterfaceActionBase**(**args*, ***kwargs*)

Bases : *Plugin* (page 268)

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

author = 'Kovid Goyal'

L'auteur de cette extension

type = 'Action interface utilisateur'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

can_be_disabled = False

Si False, l'utilisateur ne pourra pas désactiver cette extension. À utiliser avec soin.

load_actual_plugin(*gui*)

Cette méthode doit renvoyer l'extension d'action interface correspondante.

12.1.9 Extensions préférences

class calibre.customize.**PreferencesPlugin**(*plugin_path*)

Bases : *Plugin* (page 268)

Une extension représentant un widget affiché dans la boîte de dialogue des préférences.

Cette extension n'a qu'un méthode importante *create_widget()* (page 299). Les différents champs de l'extension contrôlent comment c'est catégorisé dans l'UI.

supported_platforms = ['windows', 'osx', 'linux']

Liste des plateformes où fonctionne cette extension. Par exemple `['windows', 'osx', 'linux']`

author = 'Kovid Goyal'

L'auteur de cette extension

type = 'Préférences'

Le type de cette extension. Utilisé pour catégorisé les extensions dans le GUI

can_be_disabled = False

Si False, l'utilisateur ne pourra pas désactiver cette extension. À utiliser avec soin.

config_widget = None

Importe le chemin vers le module qui contient une classe nommée ConfigWidget qui implémente ConfigWidgetInterface. Utilisé par *create_widget()* (page 299).

category_order = 100

Où dans la liste de catégories la `catégorie` de cette extension devrait se trouver.

name_order = 100

Où dans la liste de noms d'une catégorie, le *gui_name* (page 299) de cette extension devrait se trouver

category = None

La catégorie où devrait se trouver cette extension

gui_category = None

Le nom de catégorie affiché à l'utilisateur pour cette extension

gui_name = None

Le nom affiché à l'utilisateur pour cette extension

icon = None

L'icône pour cette extension, devrait être un chemin absolu

description = None

La description pour les infobulles etc

create_widget(*parent=None*)

Crée et renvoie le Qt widget effectivement utilisé pour paramétrer ce groupe de préférence. Le widget doit implémenter la *calibre.gui2.preferences.ConfigWidgetInterface* (page 299).

L'implémentation par défaut utilise *config_widget* (page 299) pour instancié le widget.

class calibre.gui2.preferences.**ConfigWidgetInterface**

Cette classe définit l'interface que tous les widgets affichée dans la boîte de dialogues de Préférences doit implémenter. Voir *ConfigWidgetBase* (page 300) pour une classe de base qui implémente cette interface et définit aussi des méthodes pratiques.

changed_signal = None

Ce signal doit être émis dès que l'utilisateur change une valeur de ce widget

supports_restoring_to_defaults = True

Fixer à TRue ssi la méthode `ConfigWidgetBase` est implémentée.

restore_defaults_desc = 'Restaurer les réglages aux valeurs par défaut. Vous devez cliquer sur Appliquer pour enregistrer réellement les réglages par défaut.'

L'infobulle pour le bouton « Restauration des paramètres par défaut »

restart_critical = False

Si True, le la boite de dialogue de Préférence ne permettra pas à l'utilisateur d'ajouter d'autres préférences. A de l'effet uniquement si *commit()* (page 300) renvoie True.

genesis(*gui*)

Appelé une fois avant que le widget ne soit affiché, devrait effectuer les paramétrages nécessaire

Paramètres
gui – L'interface graphique utilisateur principale de calibre

initialize()

Devrait définir toutes les valeurs de configuration à leurs valeurs initiales (les valeurs stockées dans les fichiers de configuration). L'instruction « return » est facultative. Renvoie False si la boîte de dialogue ne doit pas être affichée.

restore_defaults()

Devrait fixer toutes les valeurs de configuration à leur valeur par défaut.

commit()

Sauve tout changement de paramètre. Renvoie True si les changements requière un redémarrage, sinon False. Crée un exception :class :`AbortCommit pour indiquer qu'une erreur est survenue. Vous êtes responsable de donner des informations à l'utilisateur sur ce qu'est l'erreur et comment la corriger.

refresh_gui(*gui*)

Appelée une fois que ce widget est engagé. Responsable la relecture par le GUI de tous les paramètres modifiés. Notez que par défaut le GUI réinitialise de toutes façon différents éléments, donc la plupart des widgets ne nécessitent pas cette méhode.

initial_tab_changed()

Appelé si l'onglet initialement affiché est modifié avant l'affichage du widget, mais après son initialisation.

class calibre.gui2.preferences.**ConfigWidgetBase**(*parent=None*)

Une classe de base qui contient du code pour ajouter facilement des widgets de configuration standards comme des cases, des listes déroulantes, des zones de textes, etc. Voir la méthode *register()* (page 301).

Cette classe gère automatiquement changement de notification, les remises aux paramètres par défaut, la traduction entre les objets GUI et les objets configuration, etc. pour l'enregistrement de paramètres.

Si votre widget de configuration hérite de cette classe mais inclus des paramètres qui ne sont pas enregistrés, vous devriez outrepasser les méthodes *ConfigWidgetInterface* (page 299) et appelé les méthodes de base à l'intérieur des dérogations.

changed_signal

Ce signal doit être émis dès que l'utilisateur change une valeur de ce widget

supports_restoring_to_defaults = True

Fixer à TRue ssi la méthode `ConfigWidgetBase` est implémentée.

restart_critical = False

Si True, le la boite de dialogue de Préférence ne permettra pas à l'utilisateur d'ajouter d'autres préférences. A de l'effet uniquement si `commit()` (page 301) renvoie True.

register(*name*, *config_obj*, *gui_name=None*, *choices=None*, *restart_required=False*, *empty_string_is_None=True*, *setting=<class 'calibre.gui2.preferences.Setting'>*)

Enregistrer un paramètre.

Paramètres

- **name** – Le nom du paramètre
- **config_obj** – Un configuration qui lit/écrit le paramètre
- **gui_name** – Le nom de l'objet GUI qui présente une interface à changer par le paramètre. Par défaut, il est présumé être `'opt_' + name`.
- **choices** – Si le paramètre est un paramètre de base à plusieurs choix (une liste déroulante), la liste de choix. La liste est une liste de tuples sous la forme `[(nom GUI, valeur), ...]`
- **setting** – La classe responsable de la gestion de ce paramètre. La classe par défaut gère presque ous les cas, donc ce paramètre est rarement utilisé.

initialize()

Devrait définir toutes les valeurs de configuration à leurs valeurs initiales (les valeurs stockées dans les fichiers de configuration). L'instruction « return » est facultative. Renvoie False si la boîte de dialogue ne doit pas être affichée.

commit(**args*)

Sauve tout changement de paramètre. Renvoie True si les changements requière un redémarrage, sinon False. Crée un exception :class :`AbortCommit pour indiquer qu'une erreur est survenue. Vous êtes responsable de donner des informations à l'utilisateur sur ce qu'est l'erreur et comment la corriger.

restore_defaults(**args*)

Devrait fixer toutes les valeurs de configuration à leur valeur par défaut.

12.2 Variables d'environnement

- `CALIBRE_CONFIG_DIRECTORY` - paramètre le dossier où les fichiers de configuration sont stockés/lus.
- `CALIBRE_TEMP_DIR` - paramètre le dossier temporaire utilisé par calibre
- `CALIBRE_CACHE_DIRECTORY` - défini le dossier utilisé pour un cache persistant des données entre les sessions
- `CALIBRE_OVERRIDE_DATABASE_PATH` - vous permet de spécifier le chemin de fichier complet vers metadata.db. En utilisant cette variable vous pouvez avoir metadata.db dans un autre emplacement que le répertoire de la bibliothèque. Utile si votre répertoire de bibliothèque est sur un lecteur réseau qui ne supporte pas le verrouillage de fichier.
- `CALIBRE_DEVELOP_FROM` - utilisé pour exécuter à partir d'un environnement de développement calibre. Voir *Mise en place d'un environnement de développement calibre* (page 357).
- `CALIBRE_OVERRIDE_LANG` - utilisé pour forcer la langue utilisée par l'interface(code de langue ISO 639)
- `CALIBRE_TEST_TRANSLATION` - utilisé pour tester une traduction d'un fichier .po (devrait être le chemin vers le fichier .po)
- `CALIBRE_NO_NATIVE_FILEDIALOGS` - force calibre à ne pas se servir des fichiers de dialogues natifs pour la sélection des fichiers/dossiers.

- `CALIBRE_NO_NATIVE_MENUBAR` - entraîne le fait que calibre ne crée pas de menu natif (global) sur Ubuntu Unity et les environnements de bureau Linux similaires. A la place, le menu est placé dans la fenêtre, comme d'habitude.
- `CALIBRE_USE_SYSTEM_THEME` - par défaut, sous Linux, calibre utilise son propre style Qt intégré. Ceci pour éviter des plantages et des blocages par incompatibilités entre la version Qt sur laquelle est basée calibre et la version Qt du système. La contrepartie est que calibre ne suit pas l'apparence du système. Si vous paramétrez cette variable sous Linux, cela entraînera que calibre utilisera le thème du système – prendre garde aux plantages et blocages.
- `CALIBRE_SHOW_DEPRECATION_WARNINGS` - entraîne l'impression par calibre d'avertissements de dépréciation vers stdout. Utile pour les développeurs calibre.
- `CALIBRE_NO_DEFAULT_PROGRAMS` - empêche calibre d'enregistrer automatiquement les types de fichiers qu'il est capable de gérer sous Windows.
- `QT_QPA_PLATFORM` - Sous Linux, définissez ce paramètre à *wayland*` pour forcer calibre à utiliser Wayland et à `xcb` pour forcer l'utilisation de X11.
- `SYSFS_PATH` - Utiliser si sysfs est monté autre part que /sys
- `http_proxy`, `https_proxy` - utilisés sous Linux pour spécifier un proxy HTTP(S)

Voir Comment définir les variables d'environnement dans Windows[104]. Si vous êtes sous macOS, vous pouvez définir des variables d'environnement en créant le fichier `~/Library/Preferences/calibre/macos-env.txt` et en y mettant les variables d'environnement une par ligne, par exemple

```
CALIBRE_DEVELOP_FROM=$HOME/calibre-src/src
CALIBRE_NO_NATIVE_FILEDIALOGS=1
CALIBRE_CONFIG_DIRECTORY=~/.config/calibre
```

12.3 Ajustements

Les ajustements sont de petits changements que vous pouvez spécifier pour contrôler divers aspects du comportement de calibre. Vous pouvez les changer en allant dans Préférences->Avancé->Ajustements. Les valeurs par défaut pour les ajustements sont reproduits ci-dessous

```
#!/usr/bin/env python
# vim:fileencoding=UTF-8:ts=4:sw=4:sta:et:sts=4:ai
# License: GPLv3 Copyright: 2010, Kovid Goyal <kovid at kovidgoyal.net>


# Contains various tweaks that affect calibre behavior. Only edit this file if
# you know what you are doing. If you delete this file, it will be recreated from
# defaults.

#: Auto increment series index
# The algorithm used to assign a book added to an existing series a series number.
# New series numbers assigned using this tweak are always integer values, except
# if a constant non-integer is specified.
# Possible values are:
#   next - First available integer larger than the largest existing number
#   first_free - First available integer larger than 0
#   next_free - First available integer larger than the smallest existing number
#   last_free - First available integer smaller than the largest existing number.␣
↪Return largest existing + 1 if no free number is found
```

(suite sur la page suivante)

[104] https://www.computerhope.com/issues/ch000549.htm

(suite de la page précédente)

```
#   const - Assign the number 1 always
#   no_change - Do not change the series index
#   a number - Assign that number always. The number is not in quotes. Note that 0.0␣
↪can be used here.
# Examples:
#   series_index_auto_increment = 'next'
#   series_index_auto_increment = 'next_free'
#   series_index_auto_increment = 16.5
#
# Set the use_series_auto_increment_tweak_when_importing tweak to True to
# use the above values when importing/adding books. If this tweak is set to
# False (the default) then the series number will be set to 1 if it is not
# explicitly set during the import. If set to True, then the
# series index will be set according to the series_index_auto_increment setting.
# Note that the use_series_auto_increment_tweak_when_importing tweak is used
# only when a value is not provided during import. If the importing regular
# expression produces a value for series_index, or if you are reading metadata
# from books and the import plugin produces a value, then that value will
# be used irrespective of the setting of the tweak.
series_index_auto_increment = 'next'
use_series_auto_increment_tweak_when_importing = False

#: Add separator after completing an author name
# Set this if the completion separator should be appended to the end of the
# completed text to automatically begin a new completion operation for authors.
# It can be either True or False
authors_completer_append_separator = False

#: Author sort name algorithm
# The algorithm used to copy author to author_sort.
# Possible values are:
#  invert: use "fn ln" -> "ln, fn"
#  copy  : copy author to author_sort without modification
#  comma : use 'copy' if there is a ',' in the name, otherwise use 'invert'
#  nocomma : "fn ln" -> "ln fn" (without the comma)
# When this tweak is changed, the author_sort values stored with each author
# must be recomputed by right-clicking on an author in the left-hand tags
# panel, selecting 'Manage authors', and pressing
# 'Recalculate all author sort values'.
#
# The author_name_suffixes are words that are ignored when they occur at the
# end of an author name. The case of the suffix is ignored and trailing
# periods are automatically handled.
#
# The same is true for author_name_prefixes.
#
# The author_name_copywords are a set of words which, if they occur in an
# author name, cause the automatically generated author sort string to be
# identical to the author's name. This means that the sort for a string like
# "Acme Inc." will be "Acme Inc." instead of "Inc., Acme".
#
# If author_use_surname_prefixes is enabled, any of the words in
# author_surname_prefixes will be treated as a prefix to the surname, if they
# occur before the surname. So for example, "John von Neumann" would be sorted
# as "von Neumann, John" and not "Neumann, John von".
author_sort_copy_method = 'comma'
author_name_suffixes = ('Jr', 'Sr', 'Inc', 'Ph.D', 'Phd',
```

(suite sur la page suivante)

(suite de la page précédente)

```
                        'MD', 'M.D', 'I', 'II', 'III', 'IV',
                        'Junior', 'Senior')
author_name_prefixes = ('Mr', 'Mrs', 'Ms', 'Dr', 'Prof')
author_name_copywords = (
    'Agency', 'Corporation', 'Company', 'Co.', 'Council',
    'Committee', 'Inc.', 'Institute', 'National', 'Society', 'Club', 'Team',
    'Software', 'Games', 'Entertainment', 'Media', 'Studios',
)
author_use_surname_prefixes = False
author_surname_prefixes = ('da', 'de', 'di', 'la', 'le', 'van', 'von')

#: Splitting multiple author names
# By default, calibre splits a string containing multiple author names on
# ampersands and the words "and" and "with". You can customize the splitting
# by changing the regular expression below. Strings are split on whatever the
# specified regular expression matches, in addition to ampersands.
# Default: r'(?i),?\s+(and|with)\s+'
authors_split_regex = r'(?i),?\s+(and|with)\s+'

#: Use author sort in Tag browser
# Set which author field to display in the Tag browser (the list of authors,
# series, publishers etc on the left hand side). The choices are author and
# author_sort. This tweak affects only what is displayed under the authors
# category in the Tag browser and Content server. Please note that if you set this
# to author_sort, it is very possible to see duplicate names in the list because
# although it is guaranteed that author names are unique, there is no such
# guarantee for author_sort values. Showing duplicates won't break anything, but
# it could lead to some confusion. When using 'author_sort', the tooltip will
# show the author's name.
# Examples:
#   categories_use_field_for_author_name = 'author'
#   categories_use_field_for_author_name = 'author_sort'
categories_use_field_for_author_name = 'author'

#: Control partitioning of Tag browser
# When partitioning the Tag browser, the format of the subcategory label is
# controlled by a template: categories_collapsed_name_template if sorting by
# name, categories_collapsed_rating_template if sorting by average rating, and
# categories_collapsed_popularity_template if sorting by popularity. There are
# two variables available to the template: first and last. The variable 'first'
# is the initial item in the subcategory, and the variable 'last' is the final
# item in the subcategory. Both variables are 'objects'; they each have multiple
# values that are obtained by using a suffix. For example, first.name for an
# author category will be the name of the author. The sub-values available are:
#  name: the printable name of the item
#  count: the number of books that references this item
#  avg_rating: the average rating of all the books referencing this item
#  sort: the sort value. For authors, this is the author_sort for that author
#  category: the category (e.g., authors, series) that the item is in.
# Note that the "r'" in front of the { is necessary if there are backslashes
# (\ characters) in the template. It doesn't hurt anything to leave it there
# even if there aren't any backslashes.
categories_collapsed_name_template = r'{first.sort:shorten(4,,0)} - {last.
↪sort:shorten(4,,0)}'
categories_collapsed_rating_template = r'{first.avg_rating:4.2f:ifempty(0)} - {last.
↪avg_rating:4.2f:ifempty(0)}'
categories_collapsed_popularity_template = r'{first.count:d} - {last.count:d}'
```

(suite sur la page suivante)

(suite de la page précédente)

```
#: Specify columns to sort the booklist by on startup
# Provide a set of columns to be sorted on when calibre starts.
# The argument is None if saved sort history is to be used
# otherwise it is a list of column,order pairs. Column is the
# lookup/search name, found using the tooltip for the column
# Order is 0 for ascending, 1 for descending.
# For example, set it to [('authors',0),('title',0)] to sort by
# title within authors.
sort_columns_at_startup = None

#: Control how dates are displayed
# Format to be used for publication date and the timestamp (date).
#  A string controlling how the publication date is displayed in the GUI
#  d     the day as number without a leading zero (1 to 31)
#  dd    the day as number with a leading zero (01 to 31)
#  ddd   the abbreviated localized day name (e.g. 'Mon' to 'Sun').
#  dddd  the long localized day name (e.g. 'Monday' to 'Sunday').
#  M     the month as number without a leading zero (1-12)
#  MM    the month as number with a leading zero (01-12)
#  MMM   the abbreviated localized month name (e.g. 'Jan' to 'Dec').
#  MMMM  the long localized month name (e.g. 'January' to 'December').
#  yy    the year as two digit number (00-99)
#  yyyy  the year as four digit number
#  h     the hours without a leading 0 (0 to 11 or 0 to 23, depending on am/pm) '
#  hh    the hours with a leading 0 (00 to 11 or 00 to 23, depending on am/pm) '
#  m     the minutes without a leading 0 (0 to 59) '
#  mm    the minutes with a leading 0 (00 to 59) '
#  s     the seconds without a leading 0 (0 to 59) '
#  ss    the seconds with a leading 0 (00 to 59) '
#  ap    use a 12-hour clock instead of a 24-hour clock, with "ap" replaced by the␣
↪localized string for am or pm
#  AP    use a 12-hour clock instead of a 24-hour clock, with "AP" replaced by the␣
↪localized string for AM or PM
#  iso   the date with time and timezone. Must be the only format present
#  For example, given the date of 9 Jan 2010, the following formats show
#  MMM yyyy ==> Jan 2010    yyyy ==> 2010       dd MMM yyyy ==> 09 Jan 2010
#  MM/yyyy ==> 01/2010      d/M/yy ==> 9/1/10   yy ==> 10
#
# publication default if not set: MMM yyyy
# timestamp default if not set: dd MMM yyyy
# last_modified_display_format if not set: dd MMM yyyy
gui_pubdate_display_format = 'MMM yyyy'
gui_timestamp_display_format = 'dd MMM yyyy'
gui_last_modified_display_format = 'dd MMM yyyy'

#: Control sorting of titles and series in the library display
# Control title and series sorting in the library view. If set to
# 'library_order', the title sort field will be used instead of the title.
# Unless you have manually edited the title sort field, leading articles such as
# The and A will be ignored. If set to 'strictly_alphabetic', the titles will be
# sorted as-is (sort by title instead of title sort). For example, with
# library_order, The Client will sort under 'C'. With strictly_alphabetic, the
# book will sort under 'T'.
# This flag affects calibre's library display. It has no effect on devices. In
# addition, titles for books added before changing the flag will retain their
# order until the title is edited. Editing a title and hitting Enter
```

(suite sur la page suivante)

(suite de la page précédente)

```
# without changing anything is sufficient to change the sort. Or you can use
# the 'Update title sort' action in the Bulk metadata edit dialog to update
# it for many books at once.
title_series_sorting = 'library_order'

#: Control formatting of title and series when used in templates
# Control how title and series names are formatted when saving to disk/sending
# to device. The behavior depends on the field being processed. If processing
# title, then if this tweak is set to 'library_order', the title will be
# replaced with title_sort. If it is set to 'strictly_alphabetic', then the
# title will not be changed. If processing series, then if set to
# 'library_order', articles such as 'The' and 'An' will be moved to the end. If
# set to 'strictly_alphabetic', the series will be sent without change.
# For example, if the tweak is set to library_order, "The Lord of the Rings"
# will become "Lord of the Rings, The". If the tweak is set to
# strictly_alphabetic, it would remain "The Lord of the Rings". Note that the
# formatter function raw_field will return the base value for title and
# series regardless of the setting of this tweak.
save_template_title_series_sorting = 'library_order'

#: Set the list of words considered to be "articles" for sort strings
# Set the list of words that are to be considered 'articles' when computing the
# title sort strings. The articles differ by language. By default, calibre uses
# a combination of articles from English and whatever language the calibre user
# interface is set to. In addition, in some contexts where the book language is
# available, the language of the book is used. You can change the list of
# articles for a given language or add a new language by editing
# per_language_title_sort_articles. To tell calibre to use a language other
# than the user interface language, set, default_language_for_title_sort. For
# example, to use German, set it to 'deu'. A value of None means the user
# interface language is used. The setting title_sort_articles is ignored
# (present only for legacy reasons).
per_language_title_sort_articles = {
        # English
        'eng'  : (r'A\s+', r'The\s+', r'An\s+'),
        # Esperanto
        'epo': (r'La\s+', r"L'", 'L´'),
        # Spanish
        'spa'  : (r'El\s+', r'La\s+', r'Lo\s+', r'Los\s+', r'Las\s+', r'Un\s+',
                  r'Una\s+', r'Unos\s+', r'Unas\s+'),
        # French
        'fra'  : (r'Le\s+', r'La\s+', r"L'", u'L´', u'L’', r'Les\s+', r'Un\s+', r'Une\
↪s+',
                  r'Des\s+', r'De\s+La\s+', r'De\s+', r"D'", r'D´', r'D’'),
        # Polish
        'pol': (),
        # Italian
        'ita': ('Lo\\s+', 'Il\\s+', "L'", 'L´', 'La\\s+', 'Gli\\s+',
                'I\\s+', 'Le\\s+', 'Uno\\s+', 'Un\\s+', 'Una\\s+', "Un'",
                'Un´', 'Dei\\s+', 'Degli\\s+', 'Delle\\s+', 'Del\\s+',
                'Della\\s+', 'Dello\\s+', "Dell'", 'Dell´'),
        # Portuguese
        'por'  : (r'A\s+', r'O\s+', r'Os\s+', r'As\s+', r'Um\s+', r'Uns\s+',
                  r'Uma\s+', r'Umas\s+', ),
        # Romanian
        'ron'  : (r'Un\s+', r'O\s+', r'Nişte\s+', ),
        # German
```

(suite sur la page suivante)

(suite de la page précédente)

```
        'deu'  : (r'Der\s+', r'Die\s+', r'Das\s+', r'Den\s+', r'Ein\s+',
                  r'Eine\s+', r'Einen\s+', r'Dem\s+', r'Des\s+', r'Einem\s+',
                  r'Eines\s+'),
        # Dutch
        'nld'  : (r'De\s+', r'Het\s+', r'Een\s+', r"'n\s+", r"'s\s+", r'Ene\s+',
                  r'Ener\s+', r'Enes\s+', r'Den\s+', r'Der\s+', r'Des\s+',
                  r"'t\s+"),
        # Swedish
        'swe'  : (r'En\s+', r'Ett\s+', r'Det\s+', r'Den\s+', r'De\s+', ),
        # Turkish
        'tur'  : (r'Bir\s+', ),
        # Afrikaans
        'afr'  : (r"'n\s+", r'Die\s+', ),
        # Greek
        'ell'  : (r'O\s+', r'I\s+', r'To\s+', r'Ta\s+', r'Tus\s+', r'Tis\s+',
                  r"'Enas\s+", r"'Mia\s+", r"'Ena\s+", r"'Enan\s+", ),
        # Hungarian
        'hun'  : (r'A\s+', r'Az\s+', r'Egy\s+',),
}
default_language_for_title_sort = None
title_sort_articles=r'^(A|The|An)\s+'

#: Specify a folder calibre should connect to at startup
# Specify a folder that calibre should connect to at startup using
# connect_to_folder. This must be a full path to the folder. If the folder does
# not exist when calibre starts, it is ignored.
# Example for Windows:
#     auto_connect_to_folder = 'C:/Users/someone/Desktop/testlib'
# Example for other operating systems:
#     auto_connect_to_folder = '/home/dropbox/My Dropbox/someone/library'
auto_connect_to_folder = ''

#: Specify renaming rules for SONY collections
# Specify renaming rules for SONY collections. This tweak is only applicable if
# metadata management is set to automatic. Collections on SONYs are named
# depending upon whether the field is standard or custom. A collection derived
# from a standard field is named for the value in that field.
#
# For example, if the standard 'series' column contains the value 'Darkover', then the
# collection name is 'Darkover'. A collection derived from a custom field will
# have the name of the field added to the value. For example, if a custom series
# column named 'My Series' contains the name 'Darkover', then the collection
# will by default be named 'Darkover (My Series)'. For purposes of this
# documentation, 'Darkover' is called the value and 'My Series' is called the
# category. If two books have fields that generate the same collection name,
# then both books will be in that collection.
#
# This set of tweaks lets you specify for a standard or custom field how
# the collections are to be named. You can use it to add a description to a
# standard field, for example 'Foo (Tag)' instead of the 'Foo'. You can also use
# it to force multiple fields to end up in the same collection.
#
# For example, you could force the values in 'series', '#my_series_1', and
# '#my_series_2' to appear in collections named 'some_value (Series)', thereby
# merging all of the fields into one set of collections.
#
# There are two related tweaks. The first determines the category name to use
```

(suite sur la page suivante)

(suite de la page précédente)

```
# for a metadata field.  The second is a template, used to determines how the
# value and category are combined to create the collection name.
# The syntax of the first tweak, sony_collection_renaming_rules, is:
# {'field_lookup_name':'category_name_to_use', 'lookup_name':'name', ...}
#
# The second tweak, sony_collection_name_template, is a template. It uses the
# same template language as plugboards and save templates. This tweak controls
# how the value and category are combined together to make the collection name.
# The only two fields available are {category} and {value}. The {value} field is
# never empty. The {category} field can be empty. The default is to put the
# value first, then the category enclosed in parentheses, it isn't empty:
# '{value} {category:|(|)}'
#
# Examples: The first three examples assume that the second tweak
# has not been changed.
#
# 1) I want three series columns to be merged into one set of collections. The
# column lookup names are 'series', '#series_1' and '#series_2'. I want nothing
# in the parenthesis. The value to use in the tweak value would be:
#    sony_collection_renaming_rules={'series':'', '#series_1':'', '#series_2':''}
#
# 2) I want the word '(Series)' to appear on collections made from series, and
# the word '(Tag)' to appear on collections made from tags. Use:
#   sony_collection_renaming_rules={'series':'Series', 'tags':'Tag'}
#
# 3) I want 'series' and '#myseries' to be merged, and for the collection name
# to have '(Series)' appended. The renaming rule is:
#   sony_collection_renaming_rules={'series':'Series', '#myseries':'Series'}
#
# 4) Same as example 2, but instead of having the category name in parentheses
# and appended to the value, I want it prepended and separated by a colon, such
# as in Series: Darkover. I must change the template used to format the category name
#
# The resulting two tweaks are:
#    sony_collection_renaming_rules={'series':'Series', 'tags':'Tag'}
#    sony_collection_name_template='{category:||: }{value}'
sony_collection_renaming_rules={}
sony_collection_name_template='{value}{category:| (|)}'

#: Specify how SONY collections are sorted
# Specify how SONY collections are sorted. This tweak is only applicable if
# metadata management is set to automatic. You can indicate which metadata is to
# be used to sort on a collection-by-collection basis. The format of the tweak
# is a list of metadata fields from which collections are made, followed by the
# name of the metadata field containing the sort value.
# Example: The following indicates that collections built from pubdate and tags
# are to be sorted by the value in the custom column '#mydate', that collections
# built from 'series' are to be sorted by 'series_index', and that all other
# collections are to be sorted by title. If a collection metadata field is not
# named, then if it is a series- based collection it is sorted by series order,
# otherwise it is sorted by title order.
# [(['pubdate', 'tags'],'#mydate'), (['series'],'series_index'), (['*'], 'title')]
# Note that the bracketing and parentheses are required. The syntax is
# [ ( [list of fields], sort field ) , ( [ list of fields ] , sort field ) ]
# Default: empty (no rules), so no collection attributes are named.
sony_collection_sorting_rules = []
```

(suite sur la page suivante)

(suite de la page précédente)

```
#: Control how tags are applied when copying books to another library
# Set this to True to ensure that tags in 'Tags to add when adding
# a book' are added when copying books to another library
add_new_book_tags_when_importing_books = False

#: Set the maximum number of sort 'levels'
# Set the maximum number of sort 'levels' that calibre will use to resort the
# library after certain operations such as searches or device insertion. Each
# sort level adds a performance penalty. If the database is large (thousands of
# books) the penalty might be noticeable. If you are not concerned about multi-
# level sorts, and if you are seeing a slowdown, reduce the value of this tweak.
maximum_resort_levels = 5

#: Choose whether dates are sorted using visible fields
# Date values contain both a date and a time. When sorted, all the fields are
# used, regardless of what is displayed. Set this tweak to True to use only
# the fields that are being displayed.
sort_dates_using_visible_fields = False

#: Fuzz value for trimming covers
# The value used for the fuzz distance when trimming a cover.
# Colors within this distance are considered equal.
# The distance is in absolute intensity units.
cover_trim_fuzz_value = 10

#: Control behavior of the book list
# You can control the behavior of double clicks and pressing Enter on the books
# list. Choices: open_viewer, do_nothing, show_book_details,
# show_locked_book_details, edit_cell, edit_metadata. Selecting anything other
# than open_viewer, show_book_details, or show_locked_book_details has the side
# effect of disabling editing a field using a single click.
# Default: open_viewer.
# Example: doubleclick_on_library_view = 'do_nothing'
# You can also control whether the book list scrolls per item or
# per pixel. Default is per item.
doubleclick_on_library_view = 'open_viewer'
enter_key_behavior = 'do_nothing'
horizontal_scrolling_per_column = False
vertical_scrolling_per_row = False

#: Language to use when sorting
# Setting this tweak will force sorting to use the
# collating order for the specified language. This might be useful if you run
# calibre in English but want sorting to work in the language where you live.
# Set the tweak to the desired ISO 639-1 language code, in lower case.
# You can find the list of supported locales at
# https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes
# Default: locale_for_sorting = '' -- use the language calibre displays in
# Example: locale_for_sorting = 'fr' -- sort using French rules.
# Example: locale_for_sorting = 'nb' -- sort using Norwegian rules.
locale_for_sorting =  ''

#: The number of seconds to wait before sending emails
# The number of seconds to wait before sending emails when using a
# public email server like GMX/Hotmail/Gmail. Default is: 5 minutes
# Setting it to lower may cause the server's SPAM controls to kick in,
# making email sending fail. Changes will take effect only after a restart of
```

(suite sur la page suivante)

(suite de la page précédente)

```
# calibre. You can also change the list of hosts that calibre considers
# to be public relays here. Any relay host ending with one of the suffixes
# in the list below will be considered a public email server.
public_smtp_relay_delay = 301
public_smtp_relay_host_suffixes = ['gmail.com', 'live.com', 'gmx.com', 'outlook.com']

#: The maximum width and height for covers saved in the calibre library
# All covers in the calibre library will be resized, preserving aspect ratio,
# to fit within this size. This is to prevent slowdowns caused by extremely
# large covers
maximum_cover_size = (1650, 2200)

#: Where to send downloaded news
# When automatically sending downloaded news to a connected device, calibre
# will by default send it to the main memory. By changing this tweak, you can
# control where it is sent. Valid values are "main", "carda", "cardb". Note
# that if there isn't enough free space available on the location you choose,
# the files will be sent to the location with the most free space.
send_news_to_device_location = "main"

#: Unified toolbar on macOS
# If you enable this option and restart calibre, the toolbar will be 'unified'
# with the titlebar as is normal for macOS applications. However, doing this has
# various bugs, for instance the minimum width of the toolbar becomes twice
# what it should be and it causes other random bugs on some systems, so turn it
# on at your own risk!
unified_title_toolbar_on_osx = False

#: Save original file when converting/polishing from same format to same format
# When calibre does a conversion from the same format to the same format, for
# example, from EPUB to EPUB, the original file is saved, so that in case the
# conversion is poor, you can tweak the settings and run it again. By setting
# this to False you can prevent calibre from saving the original file.
# Similarly, by setting save_original_format_when_polishing to False you can
# prevent calibre from saving the original file when polishing.
save_original_format = True
save_original_format_when_polishing = True

#: Number of recently viewed books to show
# Right-clicking the "View" button shows a list of recently viewed books. Control
# how many should be shown, here.
gui_view_history_size = 15

#: Change the font size of the Book details panel in the interface
# Change the font size at which book details are rendered in the side panel and
# comments are rendered in the metadata edit dialog. Set it to a positive or
# negative number to increase or decrease the font size.
change_book_details_font_size_by = 0

#: What format to default to when using the "Unpack book" feature
# The "Unpack book" feature of calibre allows direct editing of a book format.
# If multiple formats are available, calibre will offer you a choice
# of formats, defaulting to your preferred output format if it is available.
# Set this tweak to a specific value of 'EPUB' or 'AZW3' to always default
# to that format rather than your output format preference.
# Set to a value of 'remember' to use whichever format you chose last time you
# used the "Unpack book" feature.
```

(suite sur la page suivante)

(suite de la page précédente)

```
# Examples:
#   default_tweak_format = None       (Use output format)
#   default_tweak_format = 'EPUB'
#   default_tweak_format = 'remember'
default_tweak_format = None

#: Do not preselect a completion when editing authors/tags/series/etc.
# This means that you can make changes and press Enter and your changes will
# not be overwritten by a matching completion. However, if you wish to use the
# completions you will now have to press Tab to select one before pressing
# Enter. Which technique you prefer will depend on the state of metadata in
# your library and your personal editing style.
#
# If preselect_first_completion is False and you want Tab to accept what you
# typed instead of the first completion then set tab_accepts_uncompleted_text
# to True. If you do this then to select from the completions you must press
# the Down or Up arrow keys. The tweak tab_accepts_uncompleted_text is ignored
# if preselect_first_completion is True
preselect_first_completion = False
tab_accepts_uncompleted_text = False

#: Completion mode when editing authors/tags/series/etc.
# By default, when completing items, calibre will show you all the candidates
# that start with the text you have already typed. You can instead have it show
# all candidates that contain the text you have already typed. To do this, set
# completion_mode to 'contains'. For example, if you type asi it will match both
# Asimov and Quasimodo, whereas the default behavior would match only Asimov.
completion_mode = 'prefix'

#: Sort the list of libraries alphabetically
# The list of libraries in the Copy to library and Quick switch menus are
# normally sorted by most used. However, if there are more than a certain
# number of such libraries, the sorting becomes alphabetic. You can set that
# number here. The default is ten libraries.
many_libraries = 10

#: Choose available output formats for conversion
# Restrict the list of available output formats in the conversion dialogs.
# For example, if you only want to convert to EPUB and AZW3, change this to
# restrict_output_formats = ['EPUB', 'AZW3']. The default value of None causes
# all available output formats to be present.
restrict_output_formats = None

#: Set the thumbnail image quality used by the Content server
# The quality of a thumbnail is largely controlled by the compression quality
# used when creating it. Set this to a larger number to improve the quality.
# Note that the thumbnails get much larger with larger compression quality
# numbers.
# The value can be between 50 and 99
content_server_thumbnail_compression_quality = 75

#: Image file types to treat as e-books when dropping onto the "Book details" panel
# Normally, if you drop any image file in a format known to calibre onto the
# "Book details" panel, it will be used to set the cover. If you want to store
# some image types as e-books instead, you can set this tweak.
# Examples:
#    cover_drop_exclude = {'tiff', 'webp'}
```

(suite sur la page suivante)

(suite de la page précédente)

```
cover_drop_exclude = ()

#: Exclude fields when copy/pasting metadata
# You can ask calibre to not paste some metadata fields when using the
# Edit metadata->Copy metadata/Paste metadata actions. For example,
# exclude_fields_on_paste = ['cover', 'timestamp', '#mycolumn']
# to prevent pasting of the cover, Date and custom column, mycolumn.
# You can also add a shortcut in Preferences->Shortcuts->Edit metadata
# to paste metadata ignoring this tweak.
exclude_fields_on_paste = []

#: Skip internet connected check
# Skip checking whether the internet is available before downloading news.
# Useful if for some reason your operating systems network checking
# facilities are not reliable (for example NetworkManager on Linux).
skip_network_check = False

#: Tab stop width in the template editor
# Sets the width of the tab stop in the template editor in "average characters".
# For example, a value of 1 results in a space with the width of one average␣
↪character.
template_editor_tab_stop_width = 4

#: Value for undefined numbers when sorting
# Sets the value to use for undefined numbers when sorting.
# For example, the value -10 sorts undefined numbers as if they were set to -10.
# Use 'maximum' for the largest possible number. Use 'minimum' for the smallest
# possible number. Quotes are optional if entering a number.
# Examples:
#   value_for_undefined_numbers_when_sorting = -100
#   value_for_undefined_numbers_when_sorting = '2'
#   value_for_undefined_numbers_when_sorting = -0.01
#   value_for_undefined_numbers_when_sorting = 'minimum'
#   value_for_undefined_numbers_when_sorting = 'maximum'
value_for_undefined_numbers_when_sorting = 0

#: Allow template database functions in composite columns
# If True then the template database functions book_values() and book_count()
# can be used in composite custom columns. Note: setting this tweak to True and
# using these functions in composites can be very slow.
# Default: False
allow_template_database_functions_in_composites = False


#: Change the programs that are run when opening files/URLs
# By default, calibre passes URLs to the operating system to open using
# whatever default programs are configured there. Here you can override
# that by specifying the program to use, per URL type. For local files,
# the type is "file" and for web links it is "http*". For example:
# openers_by_scheme = { "http*": "firefox %u" } will make calibre run Firefox
# for https://whatever URLs. %u is replaced by the URL to be opened. The scheme
# takes a glob pattern allowing a single entry to match multiple URL types.
openers_by_scheme = {}

#: Set the first day of the week for calendar popups
# It must be one of the values Default, Sunday, Monday, Tuesday, Wednesday,
# Thursday, Friday, or Saturday, all in English, spelled exactly as shown.
```

(suite sur la page suivante)

(suite de la page précédente)

```
calendar_start_day_of_week = 'Default'
```

12.4 Outrepasser les Icônes, les modèles, etc.

Note : calibre a un support direct pour les thèmes d'icônes, il y a plusieurs thèmes d'icônes disponibles pour calibre, que vous pouvez utiliser en allant dans *Préférences → Interface → Apparence et présentation → Changer le thème d'Icône*. Il est préférable d'utiliser les thèmes d'icônes plutôt que de remplacer les icônes individuelles.

calibre vous permet d'outrepasser les ressources statiques, comme les icônes, le Javascript et les modèles pour les jaquettes de métadonnées, catalogues, etc. avec des versions personnalisées que vous préférez. Toutes les ressources statiques sont stockées dans le sous-répertoire ressources de l'emplacement d'installation de calibre. Sous Windows c'est habituellement `C:\Program Files\app\Calibre2\resources`. Sous macOS, `/Applications/calibre.app/Contents/Resources/resources/`. Sous Linux, si vous utilisez l'installateur de binaires depuis le site web de calibre ce sera `/opt/calibre/resources`. Ces chemins peuvent changer dépendant de où vous choisissez d'installer calibre.

Vous ne devriez pas changer les fichiers de ce répertoire ressources, vu que vos changements seront écrasés lors de votre prochaine mise à jour de calibre. A la place, allez dans *Préférences → Avancé → Divers* et cliquez *Ouvrir le répertoire de configuration de calibre*. Dans ce répertoire de configuration, créer un sous dossier appelé resources et placez y les fichiers que vous voulez remplacer. Placer les fichiers dans le sous dossier approprié, par exemple placez les images dans `resources/images`, etc. calibre utilisera automatiquement votre fichier personnalisé en préférence à celui intégré la prochaine fois qu'il est démarré.

Par exemple, si vous désirez changer l'icône pour l'action *Supprimer des livres*, vous devrez d'abord regarder dans le répertoire resources intégré et voir que le ficher pertinent est `resources/images/remove_books.png`. Supposons que vous avez une icône alternative appelée `my_remove_books.png` vous devriez la sauver dans le dossier de configuration comme file *:resources/images/remove_books.png*. Toutes les icônes de l'interface utilisateur calibre sont dans `resources/images` et ses sous.répertoires. Placer un fichier de remplacement ici aura une priorité encore plus élevée qu'un thème d'icône personnalisé.

12.5 Créer votre propre thème d'icônes pour calibre

Si vous avez créé un superbe ensemble d'icônes et que vous souhaitez le partager avec les autres utilisateurs de calibre via le support de thème intégré, vous pouvez facilement les empaqueter dans un thème. Pour ce faire, aller à *Préférences → Divers → Créer un thème d'icônes*, sélectionner le dossier où vous avez posé vos icônes. Entrez alors les métadonnées du thème et cliquez sur OK. Ceci créera un fichier ZIP contenant vos icônes à thème. Vous pouvez téléverser cela sur le forum calibre à l'adresse Mobileread[105] et je rendrai votre thème disponible via le système de thème d'icônes intégré de calibre. Par défaut, le thème d'icônes que vous venez de créer sera également installé comme thème courant dans calibre. Si vous testez votre thème, pensez à supprimer les images du dossier `resources/images` afin que les icônes du thème soient utilisées.

A partir de calibre 6, vous pouvez avoir des icônes personnalisées pour le mode clair et le mode sombre. Il suffit de créer deux versions de l'icône et de nommer les fichiers avec le suffixe `-for-dark-theme` et `-for-light-theme`'. Par exemple, ``modified-for-dark-theme.png` et `modified-for-light-theme.png`. calibre utilisera alors automatiquement l'icône appropriée en fonction du thème actuel.

[105] https://www.mobileread.com/forums/forumdisplay.php?f=166

12.6 Personnalisation de calibre avec des extensions

calibre a un design hautement modulaire. Presque toutes les fonctionnalités dans calibre viennent sous la forme d'extensions. Les extensions sont utilisées pour la conversion, pour télécharger des actualités (bien que celles-ci s'appellent des recettes), pour divers composants de l'interface utilisateur, pour connecter différents périphériques, pour traiter les fichiers lorsque vous les ajouter à calibre. Vous pouvez obtenir une liste complète de toutes les extensions intégrées dans calibre en allant à *Préférences → Avancé → Extensions*.

Vous pouvez écrire vos propres extensions pour personnaliser et améliorer le coportement de calibre. L'architecture extension dans calibre est vraiment simple, voir le tutoriel *Écrire vos propres extensions pour étendre les fonctionnalités de calibre* (page 236).

Une fois que vous avez écrit un plugin, vous pouvez le télécharger sur le forum des plugins de calibre à l'adresse Mobileread[106] et il sera mis à disposition via le dispositif de mise à jour des plugins intégré à calibre.

[106] https://www.mobileread.com/forums/forumdisplay.php?f=237

CHAPITRE 13

Interface en ligne de commande

Note : Sous macOS, les outils en de ligne de commande sont à l'intérieur de la suite logicielle calibre, par exemple, si vous installez calibre dans `/Applications` les outils de ligne de commande sont dans `/Applications/calibre.app/Contents/MacOS/`. Donc, par exemple `/Applications/calibre.app/Contents/MacOS/ebook-convert`.

13.1 Commandes documentées

13.1.1 `calibre`

```
calibre [options] [path_to_ebook ou url calibre ...]
```

Lancez l'interface graphique principale de **`calibre`** et ajoutez éventuellement le livre numérique à path_to_ebook vers la base de données. Vous pouvez également spécifier des URL **`calibre`** pour effectuer différentes actions, tel que le simple ajout de livres. Par exemple :

`calibre` ://view-book/test_library/1842/epub

ouvrira le livre avec le numéro d'identification 1842 au format EPUB de la bibliothèque « test_library » dans la visionneuse de livre numérique **`calibre`**. Les noms des bibliothèques sont les noms de dossiers des bibliothèques dont les espaces sont remplacés par des caractères de soulignement. Une description complète des diverses actions basées sur l'URL se trouve dans le Manuel de l'utilisateur.

Chaque fois que vous passez à **`calibre`** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--detach

Détacher du terminal de contrôle, s'il y en a un (linux uniquement)

--help, -h

affiche ce message d'aide et quitte

--ignore-plugins

Ignorer les extensions personnalisées, utile si vous installez une extension qui empêche calibre de démarrer.

--no-update-check

Ne pas vérifier les mises à jour

--shutdown-running-calibre, -s

Provoque la fermeture de la session de calibre en cours (si elle existe). Remarquez que s'il y a des tâches en cours, elles seront annulées sur le champ. À utiliser avec prudence.

--start-in-tray

Démarrer en mode minimisé dans la zone de notification.

--verbose, -v

Ignoré, ne pas utiliser. Présent uniquement pour d'anciennes raisons

--version

affiche le numéro de version du logiciel et quitte

--with-library

Utiliser la bibliothèque présente dans le dossier spécifié.

13.1.2 calibre-customize

```
options calibre-customize
```

Personnaliser calibre en chargeant des extensions externes.

Chaque fois que vous passez à **calibre-customize** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--add-plugin, -a

Ajouter une extension en précisant le chemin de l'archive ZIP qui la contient.

--build-plugin, -b

Pour les développeurs de plugin : chemin vers le dossier où vous développez le plugin. Cette commande va automatiquement compresser le plugin et le mettre à jour dans calibre.

--customize-plugin

Customize plugin. Specify name of plugin and customization string separated by a comma. The customization string is the same as you would enter when customizing the plugin in the main calibre GUI.

--disable-plugin

Désactiver l'extension nommée

--enable-plugin

Activer l'extension nommée

--help, -h

affiche ce message d'aide et quitte

--list-plugins, -l

Lister toutes les extensions installées

--remove-plugin, -r

Supprimer une extension personnalisée par son nom. N'a pas d'effet sur les extensions intégrées

--version

affiche le numéro de version du logiciel et quitte

13.1.3 calibre-debug

```
calibre-debug [options]
```

Diverses interfaces de ligne de commande utiles pour le débogage de calibre. Sans option, cette commande lance un interpréteur Python embarqué. Vous pouvez également exécuter l'interface principale calibre, la visionneuse de livres numériques calibre et l'éditeur calibre en mode débogage.

Il contient également des interfaces pour divers éléments de calibre qui ne disposent pas d'outils dédiés en ligne de commande, comme le sous-ensemble de polices, l'outil de comparaison de livres numériques, etc.

Vous pouvez également utiliser **calibre-debug** pour exécuter des scripts autonomes. Pour ce faire, utilisez-le comme suit :

calibre-debug -e myscript.py -- --option1 --option2 file1 file2 …

Tout ce qui suit le -- est transmis au script. Vous pouvez aussi utiliser **calibre-debug** comme shebang dans les scripts, comme ceci :

#!/usr/bin/env -S **calibre-debug** -e -- –

Chaque fois que vous passez à **calibre-debug** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--add-simple-plugin

Ajouter une extension simple (par ex. une extension qui consiste en un unique fichier .py), en spécifiant le chemin vers le fichier py contenant le code de l'extension.

--command, -c

Exécuter du code Python.

--debug-device-driver, -d

Déboguer la détection du périphérique

--default-programs

(Dés)enregistrer calibre des Programmes par Défaut de Windows *--default-programs* (page 317) = (register|unregister)

--diff

Exécuter l'outil calibre diff. Par exemple : calibre-debug *--diff* (page 317) fichier1 fichier2

--edit-book

Lancez l'Éditeur de livre calibre en mode débogage.

--exec-file, -e

Exécuter le code Python dans le fichier

--explode-book, -x

Décompacte le livre dans le dossier spécifié. Utilisation : -x file.epub output_dir Exporte le livre comme une collection de fichiers HTML et de métadonnées, qui peuvent être édités en utilisant des outils standards d'édition HTML. Fonctionne avec des fichiers EPUB, AZW3, HTMLZ et DOCX

--export-all-calibre-data

Exporter toutes les données calibre (livres/réglages/plugins). Normalement, on vous demandera le dossier d'exportation et les bibliothèques à exporter. Vous pouvez également les spécifier en tant qu'arguments de ligne de commande pour sauter les questions. Utilisez des chemins absolus pour le dossier d'exportation et les bibliothèques. Le mot-clé spécial "all" peut être utilisé pour exporter toutes les bibliothèques. Exemples : calibre-debug *--export-all-calibre-data* (page 318) # pour l'utilisation interactive calibre-debug *--export-all-calibre-data* (page 318) /path/to/empty/export/folder /path/to/library/folder1 /path/to/library2 calibre-debug *--export-all-calibre-data* (page 318) /export/folder all # exporter toutes les bibliothèques connues

--fix-multiprocessing

Pour usage interne

--gui, -g

Exécuter le GUI avec le débogage activé. La sortie de débogage est inscrite dans stdout et stderr.

--gui-debug

Exécuter le GUI avec une console de débogage, inscrite à l'emplacement spécifié. Pour usage interne uniquement, utiliser l'option -g pour exécuter le GUI en mode débogage.

--help, -h

affiche ce message d'aide et quitte

--implode-book, -i

Re-compacter un livre précédemment décompacté. Utilisation : -i output_dir file.epub Importe le livre depuis les fichiers dans output_dir qui avaient été créé par une un appel précédent à call to *--explode-book* (page 318). Assurez-vous de spécifier le même type de fichier que celui utilisé lors du décompactage.

--import-calibre-data

Importer les données calibre précédemment exportées

--inspect-mobi, -m

Inspecter le(s) fichier(s) MOBI à/aux (l')emplacement(s) spécifié(s)

--paths

Sort les emplacements nécessaires pour configurer l'environnement de calibre

--run-plugin, -r

Exécuter une extension qui fournit une interface en ligne de commande. Par exemple : calibre-debug -r "Plugin name" -- file1 --option1 Tout ce qui se trouve après le -- sera passé à l'extension en tant qu'arguments.

--run-test, -t

Exécuter le(s) test(s) nommé(s). Utilisez la valeur spéciale "all" pour exécuter tous les tests. Si le nom du test commence par un point, il est supposé être un nom de module. Si le nom du test commence par @, il s'agit d'un nom de catégorie.

--run-without-debug

Ne pas exécuter avec l'option DEBUG activée

--shutdown-running-calibre, -s

Provoque la fermeture de la session de calibre en cours (si elle existe). Remarquez que s'il y a des tâches en cours, elles seront annulées sur le champ. À utiliser avec prudence.

--subset-font, -f

Créer un sous-ensemble avec les polices spécifiées. Utilisez -- après cette option pour passer l'option au programme de sous-ensembles de police.

--test-build

Tester les modules binaires en construction

--version

affiche le numéro de version du logiciel et quitte

--viewer, -w

Exécuter la visionneuse de livre numérique en mode débogage

13.1.4 calibre-server

```
calibre-server [options] [path to library folder...]
```

Démarre le Serveur de contenu calibre. Le Serveur de contenu calibre expose vos bibliothèques calibre sur internet. Vous pouvez spécifier le chemin vers les dossier de bibliothèque comme argument à **calibre-server**. Si vous ne spécifiez aucun chemin, toutes les bibliothèques connues par calibre seront utilisées.

Chaque fois que vous passez à **calibre-server** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--access-log

Chemin vers le fichier journal d'accès. Ce journal contient des informations au sujet des clients se connectant au serveur et faisant des requêtes. Par défaut aucun accès connecté n'est fini.

--ajax-timeout

Délai d'attente d'une réponse du serveur (en secondes) lors d'une requête.

--auth-mode

Choisir le type d'authentification utilisé. Paramétrer le mode d'authentification HTTP utilisé par le serveur. Paramétrer à "basic" si vous mettez le serveur derrière un proxy SSL. Autrement, laisser à "auto", lequel utilisera "basic" si SSL est configuré sinon il utilisera "digest".

--auto-reload

Recharge automatiquement le serveur lors de changements du code source. Utile pour le développement. Vous devriez aussi spécifier une valeur basse pour la temporisation d'arrêt.

--ban-after

Nombre d'échecs de connexion pour bannir. Le nombre d'échecs de connexion après lesquels une adresse IP est bannie

--ban-for

Bannir l'adresse IP qui a des échecs répétés de connexion. Bannir temporairement l'accès aux adresses IP qui ont des échecs répétés de connexion pour le nombre spécifié de minutes. Utile pour empêcher des tentatives sur des mots de passe invités. Si défini à zéro, aucun bannissement n'est effectué.

--book-list-mode

Choisir le mode liste de livres par défaut. Définir le mode liste de livres par défaut qui sera utilisé pour les nouveaux utilisateurs. Chaque utilisateur peut remplacer la valeur par défaut dans ses propres paramètres. La valeur par défaut est d'utiliser une grille de couverture.

--compress-min-size

Taille minimale pour laquelle les réponses utilisent la compression de données (en bytes).

--custom-list-template

Chemin vers un fichier JSON contenant un modèle pour le mode modèle de liste de livres. La manière la plus simple de créer un tel modèle est d'aller dans Préférences->Partager sur le net->Modèle de liste de livres, créer un modèle et l'exporter.

--daemonize

Exécute le processus en tâche de fond comme un daemon (Linux uniquement)

--displayed-fields

Restreindre l'affichage des champs définis par l'utilisateur. Liste de champs de métadonnées définis par l'utilisateur séparée par des virgules qui seront montrés par le Serveur de contenu dans les affichages /opds et /mobile. Si vous spécifiez cette option, tous les champs qui ne sont pas dans cette liste ne seront pas affichés. Par exemple : my_rating,my_tags

--enable-allow-socket-preallocation, --disable-allow-socket-preallocation

Pré-allocation d'un socket, par exemple, avec activation du socket systemd. Par défaut, cette option est activée.

--enable-auth, --disable-auth

Authentification basée sur un mot de passe pour accéder au serveur. Normalement, le serveur est sans restriction, permettant à n'importe qui d'y accéder. Vous pouvez restreindre l'accès à des utilisateurs prédéfinis avec cette option. Par défaut, cette option est désactivée.

--enable-fallback-to-detected-interface,
--disable-fallback-to-detected-interface

Retour à l'auto-détection de l'interface. Si pour une raison quelconque le serveur est incapable de se lier à l'interface dans l'option listen_on, alors il essayera de détecter une interface qui se connecte avec le monde extérieur et se liera à celle-là. Par défaut, cette option est activée.

--enable-local-write, --disable-local-write

Autoriser les connexions locales non authentifiées à apporter des modifications. Normalement, si vous n'activez pas l'authentification, le serveur fonctionne en mode lecture seule afin d'empêcher les utilisateurs anonymes d'apporter des modifications à vos bibliothèques calibre. Cette option permet aux utilisateurs se connectant à partir de l'ordinateur où le serveur s'exécute d'effectuer des modifications. Ceci permet d'exécuter le serveur sans authentification tout en utilisant calibredb pour apporter des modifications à vos bibliothèques calibre. Notez que l'activation de cette option signifie que tout programme exécuté sur l'ordinateur peut modifier vos bibliothèques calibre. Par défaut, cette option est désactivée.

--enable-log-not-found, --disable-log-not-found

Enregistrer les requêtes HTTP 404 (Not Found). Normallement, le serveur inscrit toutes les requêtes HTTP pour

les ressources qui ne sont pas trouvées. Ceci peut occasionné des inscriptions indésirables, si votre serveur est ciblé par des robots, Utiliser cette option pour l'arrêter. Par défaut, cette option est activée.

--enable-use-bonjour, --disable-use-bonjour

Annoncer les flux OPDS via BonJour. Annoncer les flux OPDS via le service BonJour, faisant que les applications de lecture basées OPDS puissent détecter et se connecter automatiquement au serveur. Par défaut, cette option est activée.

--enable-use-sendfile, --disable-use-sendfile

Zéro copie de fichiers de transfert pour augmenter la performance. Ceci utilisera zero-copy dans les transferts in-kernel lors de l'envoi de fichiers sur le réseau, augmentant les performances. Cependant, il peut causer la corruption de transferts de fichiers sur certains systèmes de fichier endommagés. Si vous faites l'expérience de transferts de fichiers corrompus, désactivez-le. Par défaut, cette option est activée.

--help, -h

affiche ce message d'aide et quitte

--ignored-fields

Champs de métadonnées définis par l'utilisateur ignorés. Liste de champs de métadonnées définis par l'utilisateur séparés par des virgules qui ne seront pas montrés par le Serveur de contenu dans les affichages /opds et /mobile. Par exemple : my_rating,my_tags

--listen-on

L'interface sur laquelle écouter pour les connexions. Le comportement par défaut est d'écouter sur toutes les interfaces IPv6 et IPv4 disponibles. Vous pouvez changer cela en, par exemple, "127.0.0.1" pour écouter uniquement des connections IPv4 à partir de la machine locale, ou en "0.0.0.0" pour toutes les connexions IPv4 entrantes.

--log

Chemin vers le fichier journal pour le journal du serveur. Ce journal contient les informations et les erreurs serveur, pas les journaux d'accès. Il est écrit par défaut sur stdout.

--manage-users

Gérer la base de données des utilisateurs autorisés à se connecter à ce serveur. Vous pouvez l'utiliser en mode automatique en ajoutant un –. Voir calibre-server *--manage-users* (page 321) -- help pour plus de détails. Voir aussi l'option *--userdb* (page 322).

--max-header-line-size

Taille max. d'une seule en-tête HTTP (en KB).

--max-job-time

Temps maximum pour les processus de travail. Quantité maximale de temps que les processus de travail sont autorisés à s'exécuter (en minutes). Régler à zéro pour aucune limite.

--max-jobs

Nombre maximum de processus de travail. Les processus de travail sont lancés en cas d'utilité et utilisés pour de volumineux travaux comme préparer un livre à la visualisation, ajouter des livres, convertir, etc. Normalement, le nombre maximal de tels processus est basé sur le nombre de cœurs du CPU. Vous pouvez le contrôler par ce réglage.

--max-log-size

Taille max. du fichier journal (en MB). La taille maximale des fichiers journaux, générés par le serveur. Quand le journal devient plus grand que cette taille, il est automatiquement remplacé. Régler à zéro pour désactiver la rotation de journal.

--max-opds-items

Nombre maximum de livres dans les flux OPDS. Le nombre maximum de livres que serveur renverra dans un seul flux d'acquisition OPDS.

--max-opds-ungrouped-items

Le nombre maximum d'éléments dégroupés dans les flux OPDS. Groupe les éléments en catégories telles que auteur/étiquettes à l'aide de la première lettre quand il y a plus que ce nombre d'articles. Réglez à zéro pour désactiver.

--max-request-body-size

Taille max. allouée pour les fichiers téléchargés sur le serveur (en MB).

--num-per-page

Nombre de livres à afficher sur une seule page. Le nombre de livres à afficher sur une seule page dans le navigateur.

--pidfile

Écrire le PID du processus dans le fichier spécifié

--port

Le port sur lequel écouter pour les connexions.

--search-the-net-urls

Chemin d'accès à un fichier JSON contenant les URL de la fonction "Rechercher sur Internet". La façon la plus simple de créer un tel fichier est d'aller dans Préférences-> Partager sur le net->Rechercher sur Internet dans calibre, créer les URLs et les exporter.

--shutdown-timeout

Temps total en secondes à attendre pour un arrêt approprié.

--ssl-certfile

Chemin vers le fichier de certificat SSL.

--ssl-keyfile

Chemin vers le fichier de clé privée SSL.

--timeout

Temps (en secondes) après lequel une connexion inactive est fermée.

--trusted-ips

Autoriser les connexions non authentifiées à partir d'adresses IP spécifiques pour effectuer des modifications. Normalement, si vous n'activez pas l'authentification, le serveur fonctionne en mode lecture seule, afin de ne pas permettre à des utilisateurs anonymes d'apporter des modifications à vos bibliothèques calibre. Cette option permet à toute personne se connectant à partir des adresses IP spécifiées d'effectuer des modifications. Il doit s'agir d'une liste d'adresses ou de spécifications de réseau séparées par des virgules. Cette option est utile si vous voulez faire fonctionner le serveur sans authentification mais que vous utilisez tout de même calibredb pour modifier vos bibliothèques de calibre. Notez que l'activation de cette option signifie que toute personne se connectant à partir des adresses IP spécifiées peut apporter des modifications à vos bibliothèques calibre.

--url-prefix

Un préfixe à ajouter à toutes les URLs. Utile si vous souhaitez exécuter ce serveur derrière un proxy inversé. Par exemple, utilisez /calibre comme préfixe d'URL.

--userdb

Chemin vers la base de données utilisateur à utiliser pour l'authentification. La base de données est un fichier SQLite. Pour le créer utilisez *--manage-users* (page 321). Vous pouvez en lire plus à propos de la gestion des utilisateurs sur : https://manual.calibre-ebook.com/fr/server.html#managing-user-accounts-from-the-command-line-only

--version

affiche le numéro de version du logiciel et quitte

--worker-count

Nombre de processus de travail utilisés pour effectuer les requêtes.

13.1.5 calibre-smtp

```
calibre-smtp [options] [from to text]
```

Envoyer des courriels en utilisant le protocole SMTP. **calibre-smtp** a deux modes d'opération. Dans le mode composer vous spécifiez l'expéditeur, le destinataire et le texte, ceux-ci sont utilisés pour élaborer et envoyer un message électronique. Dans le mode filtre, **calibre-smtp** lit un message électronique complet depuis STDIN et l'envoie.

le texte est le corps du message. Si le texte n'est pas spécifié, un message complet est lu depuis STDIN. de est l'adresse courriel de l'expéditeur et à est l'adresse courriel du destinataire. Quand un courriel complet est lu depuis STDIN, de et à sont uniquement utilisés dans la négociation SMTP, l'en-tête de message n'est pas modifié.

Chaque fois que vous passez à **calibre-smtp** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--fork, -f

Redirige et délivre le message en tâche de fond. Si vous utilisez cette option, vous devriez aussi aussi utiliser *--outbox* (page 323) pour traiter les erreurs de livraison.

--help, -h

affiche ce message d'aide et quitte

--localhost, -l

Nom d'hôte de l'hôte local. Utilisé lors d'une connexion à un serveur SMTP.

--outbox, -o

Chemin vers le dossier maildir où enregistrer les courriels qui ont échoué.

--timeout, -t

Délai d'attente pour la connexion

--verbose, -v

Soyez plus détaillé

--version

affiche le numéro de version du logiciel et quitte

COMPOSE MAIL

Options pour rédiger un courriel. Ignoré si le texte n'est pas spécifié

--attachment, -a
Fichier à attacher au courriel

--subject, -s
Sujet du courriel

SMTP RELAY

Options pour utiliser un serveur de relais SMTP pour envoyer un courriel. calibre essayera d'envoyer un courriel directement à moins que –relay est spécifié.

--cafile
Chemin d'accès à un fichier de certificats CA concaténés au format PEM, utilisé pour vérifier le certificat du serveur lors de l'utilisation de TLS. Par défaut, les certificats CA du système sont utilisés.

--dont-verify-server-certificate
Ne vérifiez pas le certificat de serveur lors de la connexion à l'aide de TLS. C'était le comportement par défaut dans les versions de calibre antérieures à 3.27. Si vous utilisez un relais avec un certificat auto-signé ou autre certificat invalide, vous pouvez utiliser cette option pour restaurer le comportement d'avant la version 3.27.

--encryption-method, -e
Méthode de cryptage à utiliser lors de la connection à un relais. Les choix sont TLS, SSL et AUCUN. L'option par défaut est TLS. ATTENTION : choisir AUCUN est très risqué

--password, -p
Mot de passe pour le relais

--port
Port pour se connecter à un serveur relais. L'option par défaut est d'utiliser le 465 si la méthode de cryptage est le SSL et 25 autrement.

--relay, -r
Un serveur de relais SMTP à utiliser pour envoyer un courriel.

--username, -u
Nom d'utilisateur pour le relais

13.1.6 calibredb

```
calibredb command [options] [arguments]
```

calibredb est l'interface en ligne de commande de la base de données de calibre. Elle a plusieurs sous-commandes, documentées ci-dessous.

La commande :*calibredb* peut être utilisée pour manipuler soit une base de données de calibre spécifiée par le chemin ou calibre *Serveur de contenu* fonctionnant soit en local soit sur Internet. Vous pouvez lancer calibre *Serveur de contenu* en utilisant soit la commande **calibre-server**. ou dans le programme principal de calibre, cliquer sur *Connecter/partager → Démarrer le serveur de contenu*. Puisque **calibredb** peut apporter des modifications à vos bibliothèques vous devez d'abord configurer l'authentification sur le serveur. Là il y a deux façons de le faire :

- Si vous prévoyez de vous connecter uniquement à un serveur fonctionnant sur le même ordinateur,

vous pouvez simplement utiliser l'option « –enable-local-write » du Serveur de contenu pour permettre à tout programme, y compris calibredb, de fonctionner sur l'ordinateur local pour apporter des modifications aux données de calibre. Lors de l'exécution le serveur du programme principal de calibre, cette option se trouve sous *Préférences → Partage sur le net → Avancé*.

- Si vous souhaitez activer l'accès à Internet, vous devez le configurer

sur le serveur et utilisez les options `--username` (page 326) et `--password` (page 326). pour donner accès à **calibredb**. Vous pouvez configurer l'authentification de l'utilisateur pour **calibre-server** en utilisant la commande `--enable-auth`. et en utilisant l'option « –manage-users » pour créer les comptes utilisateurs. Si vous exécutez le serveur à partir du programme principal de calibre, utilisez l'option *Préférences → Partage sur le net → Nom d'utilisateur/mot de passe requis*.

Pour vous connecter à un serveur de contenu en cours d'exécution, passez l'URL du serveur l'option `--with-library` (page 326), voir la documentation de cette option pour des détails et exemples.

- *OPTIONS GLOBALES* (page 326)
- *list* (page 326)
- *add* (page 327)
  - *AJOUTER A PARTIR DES DOSSIERS* (page 328)
- *remove* (page 329)
- *add_format* (page 329)
- *remove_format* (page 329)
- *show_metadata* (page 329)
- *set_metadata* (page 330)
- *export* (page 330)
- *catalog* (page 331)
  - *EPUB OPTIONS* (page 332)
- *saved_searches* (page 333)
- *add_custom_column* (page 334)
- *custom_columns* (page 334)
- *remove_custom_column* (page 334)
- *set_custom* (page 335)
- *restore_database* (page 335)
- *check_library* (page 335)
- *list_categories* (page 336)
- *backup_metadata* (page 336)
- *clone* (page 337)
- *embed_metadata* (page 337)
- *search* (page 337)
- *fts_index* (page 338)
- *fts_search* (page 338)

OPTIONS GLOBALES

--help, -h

affiche ce message d'aide et quitte

--library-path, --with-library

Chemin vers la bibliothèque calibre. Par défaut utilise le chemin enregistré dans les paramètres. Vous pouvez vous connecter à un Serveur de contenu calibre pour effectuer des actions sur des bibliothèques distantes. Pour faire cela utiliser une URL de la forme : http://hostname:port/#library_id par exemple, http://localhost:8080/#mylibrary. library_id est l'identifiant de la bibliothèque à laquelle vous vous voulez cous connecter sur le Serveur de Contenu. Vous pouvez utiliser la la valeur spéciale library_id - pour obtenir une liste des bibliothèques disponibles sur le serveur. Pour plus de détails sur comment paramétrer l'accès via un Serveur de Contenu, voir https://manual.calibre-ebook.com/fr/generated/fr/calibredb.html.

--password

Mot de passe pour la connexion à un Serveur de contenu calibre. Pour lire le mot de passe en entrée standard, utilisez la valeur spéciale : <stdin>. Pour lire le mot de passe à partir d'un fichier, utilisez : <f :/path/to/file>. (c.a.d <f : suivi du chemin complet du fichier et ajouter >). Les chevrons (< ou >) ci-dessus sont obligatoires, n'oubliez pas de les échapper ou d'utiliser des guillemets pour votre shell.

--timeout

Le délai d'attente, en secondes, lors de la connexion à une bibliothèque calibre sur le réseau. La valeur par défaut est de deux minutes.

--username

Nom d'utilisateur pour la connexion au Serveur de contenu calibre

--version

affiche le numéro de version du logiciel et quitte

list

```
calibredb list [options]
```

Lister les livres disponibles dans la base Calibre.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--ascending

Trier les résultats dans l'ordre ascendant

--fields, -f

Les champs à afficher quand on liste les livres dans la base de données. Devrait être une liste de champs séparés par une virgule. Champs disponibles : author_sort, authors, comments, cover, formats, identifiers, isbn, languages, last_modified, pubdate, publisher, rating, series, series_index, size, tags, template, timestamp, title, uuid Par défaut : title,authors. Le champ spécial "all" peut être utiliser pour sélectionner tous les champs. En supplément des champs intégrés ci-dessous, les champs personnalisés sont également disponibles en tant que *field_name, par exemple, pour un champ personnalisé #notation, utiliser le nom : *notation

--for-machine

Générer une sortie au format JSON, qui est plus commode pour une analyse machine. Entraîne les options largeur de ligne et séparateur à être ignorés.

--limit

Le maximum de résultats à afficher. Par défaut : tous

--line-width, -w

Largeur maximum pour une seule ligne en sortie. Par défaut détecte la taille de l'écran.

--prefix

Préfixe pour tous les chemins vers des fichiers. Par défaut : chemin absolu du dossier de la bibliothèque.

--search, -s

Filtrer les résultats à l'aide de la requête de recherche. Pour le format de la requête de recherche, veuillez consulter la section recherche dans le manuel utilisateur. Par défaut n'effectue aucun filtrage

--separator

La chaîne utilisée pour séparer des champs. Par défaut : un espace

--sort-by

Le champ par lequel les résultats doivent être triés. Vous pouvez spécifier plusieurs champs en les séparant par des virgules. Champs disponibles : author_sort, authors, comments, cover, formats, identifiers, isbn, languages, last_modified, pubdate, publisher, rating, series, series_index, size, tags, template, timestamp, title, uuid Défaut : id

--template

Le modèle à exécuter si "template" figure dans la liste des champs. Default : None

--template_file, -t

Chemin d'accès à un fichier contenant le modèle à exécuter si "template" figure dans la liste des champs. Default : None

--template_heading

Intitulé de la colonne du modèle. Valeur par défaut : template. Cette option est ignorée si l'option *--for-machine* (page 326) est définie.

add

```
calibredb add [options] fichier1 fichier2 fichier3 …
```

Ajouter les fichiers spécifiés comme livre numérique à la base de données. Vous pouvez aussi spécifier des dossiers, voir ci-dessous les options relatives aux dossiers.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--authors, -a

Définissez les auteurs du(es) livre(s) ajouté(s)

--automerge, -m

Si des livres dont les titres et les auteurs sont similaires sont trouvés, fusionnez automatiquement les formats (fichiers) entrants dans les fiches de livres existantes. Une valeur "ignore" signifie que les formats en double sont éliminés. Une valeur "overwrite" signifie que les formats en double dans la bibliothèque sont écrasés avec les fichiers nouvellement ajoutés. Une valeur de "new_record" signifie que les formats dupliqués sont placés dans une nouvelle fiche de livre.

--cover, -c

Chemin à utiliser pour la couverture du livre ajouté

--duplicates, -d

Ajouter des livres à la base de données même s'ils existent déjà. La comparaison se fait sur la base des titres des livres et des auteurs. Notez que l'option *--automerge* (page 327) est prioritaire.

--empty, -e

Ajouter un livre vide (un livre sans formats)

--identifier, -I

Paramètres les identifiants pour ce livre, par ex. -I asin :XXX -I isbn :YYY

--isbn, -i

Définir l'ISBN du(es) livre(s) ajouté(s)

--languages, -l

Une liste de langages séparés par une virgule (mieux d'utiliser les codes de langue IS0639, ainsi certains noms de langues peuvent aussi être reconnus)

--series, -s

Définir les séries du(es) livre(s) ajouté(s)

--series-index, -S

Assigne le numéro de la série de(s) livre(s) ajoutés

--tags, -T

Définir les étiquettes du(es) livre(s) ajouté(s)

--title, -t

Définir le titre du(es) livre(s) ajouté(s)

AJOUTER A PARTIR DES DOSSIERS

Options pour contrôler l'ajout de livres à partir de dossiers. Par défaut seuls les fichiers qui ont des extensions de type livre numérique connues sont ajoutés.

--add

Un motif de nom de fichier (glob), les fichiers correspondant à ce motif seront ajoutés lors de la recherche de fichiers dans les dossiers, même s'ils ne sont pas d'un type de fichier de livre numérique connu. Peut être spécifié plusieurs fois pour plusieurs motifs.

--ignore

Un motif de nom de fichier (glob), les fichiers correspondant à ce motif seront ignorés lors de la recherche de fichiers dans les dossiers. Peut être spécifié plusieurs fois pour plusieurs motifs. Par exemple : *.pdf ignorera tous les fichiers PDF

--one-book-per-directory, -1

Suppose que chaque dossier dispose d'un seul livre logique et que tous les fichiers dans celui-ci sont des formats différents de ce même livre.

--recurse, -r

Traite les dossiers récursivement

remove

```
calibredb supprime les ids
```

Supprime les livres identifiés par leurs ids de la base de données. Les ids devraient être une liste de nombres id séparés par des virgules (vous pouvez obtenir les numéros id en utilisant la commande search). Par exemple, 23,34,57-85 (quand un intervalle est précisé, le dernier nombre dans l'intervalle n'est pas inclus).

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--permanent

N'utilisez pas les Corbeille

add_format

```
calibredb add_format [options] id ebook_file
```

Ajouter le livre numérique dans ebook_file aux formats disponibles pour le livre logique identifié par l'id. Vous pouvez obtenir l'id à l'aide de la commande search. Si le format existe déjà, il est remplacé, à moins que l'option ne pas remplacer est spécifiée.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--as-extra-data-file

Ajouter le fichier en tant que fichier de données supplémentaire au e-livre, et non en tant que format ebook

--dont-replace

Ne pas remplacer le format s'il existe déjà

remove_format

```
calibredb remove_format [options] id fmt
```

Supprime le format fmt du livre logique identifié par l'id. Vous pouvez obtenir l'identifiant à l'aide de la commande search. fmt devrait être une extension de fichier comme LRF, TXT ou EPUB. Si le fichier logique n'a pas de format fmt disponible, aucune action n'est effectuée.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

show_metadata

```
calibredb show_metadata [options] id
```

Afficher les métadonnées stockées dans la base de données calibre pour le livre identifié par l'id. id est un identifiant obtenu à l'aide de la commande search.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--as-opf

Imprimer les métadonnées dans un formulaire OPF (XML)

set_metadata

```
calibredb set_metadata [options] book_id [/path/to/metadata.opf]
```

Place les métadonnées stockées dans la base de données de calibre pour le livre identifié par
book_id à partir du fichier OPF metadata.opf. book_id est un numéro d'id issu de la commande rechercher. Vous pouvez obtenir un aperçu rapide du format OPF à

l'aide du commutateur –as-opf de la commande show_metadata. Vous pouvez également placer les métadonnées de champs individuels avec l'option –field. Si vous utilisez l'option –field, il n'est pas nécessaire de spécifier un fichier OPF.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--field, -f

Le champ à définir. Le format est nom_champ :valeur, par exemple : *--field* (page 330) tags :tag1,tag2. Utilisez *--list-fields* (page 330) pour récupérer une liste de tous les noms de champs. Vous pouvez spécifier cette option plusieurs fois pour définir plusieurs champs. Note : pour le champs "languages" (langues), vous devez utiliser les codes de langues ISO639 (par exemple, en pour Anglais, fr pour Français, et ainsi de suite). Pour les identifiants (identifiers), la syntaxe est *--field* (page 330) identifiers :isbn :XXXX,doi :YYYYY. Pour les champs booléens (oui/non), utilisez "true" (vrai) ou "false" (faux) ou "yes" (oui) et "no" (non).

--list-fields, -l

Lister les champs de métadonnées pouvant être utilisés avec l'option *--field* (page 330)

export

```
calibredb export [options] ids
```

Exporter les livres spécifiés par ids (une liste séparée par des virgules) vers le système de fichiers. L'opération d'**export** enregistre tous les formats du livre, la couverture et les métadonnées (dans n'importe quel un fichier OPF). Tous les fichiers de données supplémentaires associés au livre sont également enregistrés. Vous pouvez obtenir les numéros id à partir de la commande search.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--all

Exporter tous les livres dans la base, en ignorant la liste des ids.

--dont-asciiize

Faire en sorte que le calibre convertisse tous les caractères non anglais en équivalents anglais pour les noms de fichiers. Ceci est utile si vous enregistrez sur un ancien système de fichiers qui ne prend pas entièrement en charge les noms de fichiers Unicode. Spécifier cette bascule désactivera ce comportement.

--dont-save-cover

Normalement, calibre enregistrera la couverture dans un fichier séparé avec les fichiers du livre numérique. Spécifier cette bascule désactivera ce comportement.

--dont-save-extra-files

Enregistrez tous les fichiers de données associés au livre lors de l'enregistrement du livre Spécifier cette bascule désactivera ce comportement.

--dont-update-metadata

En temps normal calibre mettra à jour les métadonnées des fichiers sauvegardés à partir du contenu de la bibliothèque calibre. L'enregistrement prendra davantage de temps. Spécifier cette bascule désactivera ce comportement.

--dont-write-opf

Normalement, calibre écrira les métadonnées dans un fichier OPF séparé avec les fichiers de livre numérique actuels. Spécifier cette bascule désactivera ce comportement.

--formats

Liste séparée par des virgules de formats à enregistrer pour chaque livre. Par défaut tous les formats disponibles sont sauvegardés.

--progress

Signaler l'avancement

--replace-whitespace

Remplacer les espaces par des underscores.

--single-dir

Exporter tous les livres vers un dossier unique

--template

Modèle pour contrôler le nom des fichier et la structure des dossiers d'enregistrement des fichiers. Par défaut : "{author_sort}/{title}/{title} - {authors}" qui va sauvegarder les livres dans un sous-dossier par auteur avec des noms de fichiers contenant le titre et l'auteur. Les variables disponibles sont : {author_sort, authors, id, isbn, languages, last_modified, pubdate, publisher, rating, series, series_index, tags, timestamp, title}

--timefmt

Le format suivant lequel afficher les dates. %d - jour, %b - mois, %m - numéro de mois, %Y - année. Par défaut : %b, %Y

--to-dir

Exporter les livres vers le dossier spécifié. Par défaut : .

--to-lowercase

Convertir les chemins en minuscule.

catalog

```
calibredb catalog /path/to/destination.(csv|epub|mobi|xml...) [options]
```

Export a **catalog** in format specified by path/to/destination extension. Options control how entries are displayed in the generated **catalog** output. Note that different **catalog** formats support different sets of options. To see the different options, specify the name of the output file and then the –help option.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--ids, -i

Liste séparée par des virgules d'identifiants de base de données à cataloguer. Si déclaré, *--search* (page 331) est ignoré. Par défaut : all

--search, -s

Filtrer les résultats par la requête de recherche. Pour le format de la requête de recherche, veuillez consulter la section recherche dans le Manuel Utilisateur. Par défaut : Pas de filtre

--verbose, -v

Affiche des informations détaillées. Utile pour le débogage

EPUB OPTIONS

--catalog-title

Titre du catalogue généré utilisé comme titre dans les métadonnées. Par défaut : 'My Books' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--cross-reference-authors

Crée des références croisées dans la section Auteurs pour les livres avec des auteurs multiples. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB et MOBI

--debug-pipeline

Enregistre la sortie à différentes étapes du processus de conversion vers le dossier spécifié. Utile si vous n'êtes pas sûr de l'étape de la conversion à laquelle se produit le bogue. Par défaut : 'None' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--exclude-genre

Expression régulière décrivant les étiquettes à exclure comme genres. Par défaut : '[.+]|^+$' exclut les étiquettes entre crochets, par ex. '[Project Gutenberg]', et '+', l'étiquette par défaut pour les livres lus. S'applique aux formats de sortie : AZW3, EPUB, MOBI

--exclusion-rules

Indique les règles à appliquer pour exclure des livres du catalogue généré. Le modèle pour une règle d'exclusion est ('<nom de la règle>','Étiquettes','<liste d'étiquettes séparées par des virgules>') or ('<nom de la règle>','<nom de colonne>','<valeur>'). Par exemple : (('Livres archivés','#statut','Archivé'),) exclura les livres qui ont la valeur 'Archivé' dans la colonne 'statut'. Quand plusieurs règles sont définies, toutes les règles seront appliquées. Par défaut : "(('Catalogs','Tags','Catalog'),)" S'applique aux formats de sortie AZW3, EPUB, MOBI

--generate-authors

Inclut la section 'Auteurs' dans le catalogue. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--generate-descriptions

Inclut la section 'Descriptions' dans le catalogue. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--generate-genres

Inclut la section 'Genres' dans le catalogue. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--generate-recently-added

Inclut la section 'Ajoutés récemment' dans le catalogue. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--generate-series

Inclut la section 'Series' dans le catalogue. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--generate-titles

Inclut la section 'Titres' dans le catalogue. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--genre-source-field

Champ source pour la section 'Genres'. Par défaut : 'Étiquettes' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--header-note-source-field

Champ personnalisé contenant une note de texte à insérer dans l'en-tête de Description. Par défaut : '' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--merge-comments-rule

#<custom field> : spécifier [before|after] :[True|False] : <custom field> Champ personnalisé contenant des notes à fusionner avec les commentaires [before|after] Placement des notes par rapport au commentaires [True|False] - Une ligne horizontale est insérée entre les notes et les commentaires Par défaut : ' ::' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--output-profile

Spécifie le profil de sortie. Dans certains cas, un profil de sortie est requis pour optimiser le rendu en fonction du périphérique. Par exemple, 'kindle' ou 'kindle_dx' créent une table des matières structurée avec des sections et des articles. Par défaut : 'None' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--prefix-rules

Indique les règles utilisées pour inclure des préfixes qui indiquent qu'un livre est lu, qu'il est sur votre liste d'envies et d'autres indications personnalisées. Le modèle pour une règle de préfixe est : ('<rule name>','<source field>','<pattern>','<prefix>'). Quand plusieurs règles sont définies, toutes les règles s'appliquent. Par défaut : "(('Read books','tags','+','✓'),('Wishlist item','tags','Wishlist','×'))" S'applique aux formats de sortie AZW3, EPUB, MOBI

--preset

Employe un préréglage appelé créé avec le constructeur de Catalogue GUI. Un préréglage spécifie tous les paramètres pour construire un catalogue. Par défaut : 'None' S'applique aux formats de sortie AZW3, EPUB, MOBI

--thumb-width

Indice de taille (en pouces) pour les couvertures de livres dans le catalogue. Gamme : 1.0 - 2.0 Par défaut : '1.0' S'applique aux formats de sortie : AZW3, EPUB, MOBI

--use-existing-cover

Remplace la couverture existante lorsque le catalogue est généré. Par défaut : 'False' S'applique aux formats de sortie : AZW3, EPUB, MOBI

saved_searches

```
calibredb saved_searches [options] (list|add|remove)
```

Gère les recherches sauvegardées stockées dans la base de données. Si vous essayez d'ajouter une requête avec un nom préexistant, elle sera remplacée.

Syntaxe pour l'ajout :

calibredb **saved_searches** add search_name search_expression

Syntaxe pour la suppression :

calibredb **saved_searches** remove search_name

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

add_custom_column

```
calibredb add_custom_column [options] label name datatype
```

Créer une colonne personnalisée. label correspond au nom de la colonne pour le programme. Ne doit pas contenir d'espace ou de deux-points. name est le nom de la colonne pour l'utilisateur. datatype est à choisir parmi : bool, comments, composite, datetime, enumeration, float, int, rating, series, text

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--display

Un dictionnaire des options pour personnaliser la manière dont les données dans cette colonne seront interprétées. C'est une chaîne JSON. Pour des colonnes d'énumération, employez *--display* (page 334)"{\"enum_values\" :[\"val1\", \"val2\"]}" Il y a beaucoup d'options qui peuvent entrer dans la variable d'affichage.Les options par type de colonne sont : composite : composite_template, composite_sort, make_category, contains_html, use_decorations datetime : date_format enumeration : enum_values, enum_colors, use_decorations int, float : number_format text : is_names, use_decorations La meilleure manière de trouver des combinaisons légitimes est de créer dans le GUI une colonne personnalisée du type approprié et d'ensuite de regarder dans la sauvegarde de l'OPF d'un livre (assurez-vous qu'un nouvel OPF a été créé depuis que la colonne a été ajoutée). Vous verrez le JSON pour l'« affichage » pour la nouvelle colonne dans l'OPF.

--is-multiple

Cette colonne stocke les étiquettes comme des données (par exemple des valeurs multiples séparées par des virgules). Appliqué seulement si le type de donnée est du texte.

custom_columns

```
calibredb custom_columns [options]
```

Liste des colonnes personnalisées disponibles. Affiche les entêtes de colonne et les ids.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--details, -d

Afficher les détails pour chaque colonne.

remove_custom_column

```
calibredb remove_custom_column [options] label


Enlever la colonne personnalisée identifiée par le libellé. Vous pouvez afficher
les colonnes disponibles avec la commande custom_columns.
```

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--force, -f

Ne pas demander de confirmation

set_custom

```
calibredb set_custom [options] valeur de l'id de la colonne
```

Définir la valeur d'une colonne personnalisée pour le livre identifié par l'id. Vous pouvez obtenir une liste d'ids en utilisant la commande search. Vous pouvez obtenir une liste des noms de colonnes personnalisés en utilisant la commande custom_columns.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--append, -a

Si la colonne stocke des valeurs multiples, ajouter les valeurs spécifiées à celles existantes, au lieu de les remplacer.

restore_database

```
calibredb restore_database [options]
```

Restaure la base de données à partir des métadonnées stockées dans les fichiers OPF présents dans chaque dossier de la bibliothèque calibre. Ceci est utile si votre fichier metadata.db a été corrompu.

ATTENTION : cette commande régénère complètement votre base de données. Vous perdrez toutes les recherches sauvegardées, les catégories utilisateur, les tableaux de connexions, les réglages de conversion par livre et les recettes personnalisées. Les métadonnées restaurées seront aussi précises que ce qui sera trouvé dans les fichiers OPF.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--really-do-it, -r

Effectuer réellement la récupération. La commande ne sera pas exécutée si cette option n'est pas spécifiée.

check_library

```
calibredb check_library [options]
```

Réalise des vérifications sur le système de fichiers représentant une bibliothèque. Les rapports sont invalid_titles, extra_titles, invalid_authors, extra_authors, missing_formats, extra_formats, extra_files, missing_covers, extra_covers, failed_folders

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--csv, -c

Sortie en CSV

--ignore_extensions, -e

Liste d'extensions à ignorer séparées par des virgules. Par défaut : toutes

--ignore_names, -n

Liste de noms à ignorer séparés par des virgules. Par défaut : tous

--report, -r

Liste de rapports séparés par des virgules. Par défaut : tous

--vacuum-fts-db

Nettoyer la base de données de recherche en texte intégral. Cette opération peut être très lente et gourmande en mémoire, en fonction de la taille de la base de données.

list_categories

```
calibredb list_categories [options]
```

Produit un rapport sur les informations des catégories dans la base de données. Les informations sont les équivalents de ce qui est affiché dans le Navigateur d'étiquettes.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--categories, -r

Liste de catégories de noms de recherche séparées par des virgules. Par défaut : tous

--csv, -c

Sortie en CSV

--dialect

Le type de fichier CSV à produire. Choix : excel, excel-tab, unix

--item_count, -i

Sort seulement le nombre d'articles dans une catégorie au lieu du nombre d'occurrences par article à l'intérieur de la catégorie

--width, -w

Largeur maximum pour une seule ligne en sortie. Par défaut détecte la taille de l'écran.

backup_metadata

```
calibredb backup_metadata [options]
```

Sauvegarde les métadonnées stockées dans la base de données dans les fichiers OPF individuels dans chaque dossier de livres. Ceci se fait normalement automatiquement, mais vous pouvez lancer cette commande pour forcer la régénération des fichiers OPF, avec l'option –all.

Notez qu'il y a normalement aucun besoin de faire ceci, car les fichiers OPF sont mis à jour automatiquement, chaque fois que les métadonnées sont changées.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--all

Normalement, cette commande opère seulement sur les livres qui ont des fichiers OPF périmés. Cette option la fait opérer sur tous les livres.

clone

```
calibredb clone path/to/new/library
```

Crée un **clone** de la bibliothèque active. Vous obtiendrez une bibliothèque vide qui a exactement les même colonnes personnalisées, Bibliothèques virtuelles et autres paramétrages de la bibliothèque active.

Cette bibliothèque clonée ne contiendra aucun livre. Si vous voulez avoir une copie complète, contenu inclus, le plus simple est de copier manuellement le dossier contenant cette bibliothèque.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

embed_metadata

```
calibredb embed_metadata [options] book_id
```

Met à jour les métadonnées dans les fichiers de livre stockées dans la bibliothèque calibre depuis les métadonnées de la base de données calibre. Normalement, les métadonnées sont mises à jour uniquement quand des fichiers sont exportés depuis calibre, cette commande est utile si vous voulez que les fichiers soient mis à jour sur place. Notez que les différents formats de fichiers supportent différentes quantités de métadonnées. Vous pouvez utiliser la valeur spéciale "all" de book_id pour mettre à jour les métadonnées dans tous les livres. Vous pouvez aussi spécifier beaucoup d'ids de livres séparés par des espaces et des ensembles d'ids séparés par des traits d'union : calibredb **embed_metadata** 1 2 10-15 23

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--only-formats, -f

Met uniquement à jour les métadonnées dans les fichiers au format spécifié. Spécifier le plusieurs fois pour de multiples formats. Par défaut, tous les formats sont mis à jour.

search

```
calibredb search [options] expression de recherche
```

Recherche dans la bibliothèque après le terme spécifié dans la recherche, renvoie une liste séparée par une virgule de livres dont les ids correspondent à l'expression de recherche. Le format de sortie est utilisé pour alimenter d'autres commandes qui acceptent une liste d'ids comme entrée.

L'expression de recherche peut être n'importe quoi à partir du puissant langage de requête de recherche de calibre, par exemple : calibredb **search** author :asimov "title :»i robot »"

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--limit, -l

Le nombre maximum de résultats à renvoyer. Par défaut tous les résultats

fts_index

```
calibredb fts_index [options] enable/disable/status/reindex
```

Contrôle le processus d'indexation de la recherche intégrale du texte.

enable Active l'indexation RIT disable Désactive l'indexation RIT pour cette bibliothèque. status Affiche le statut actuel de l'indexation reindex Peut être utilisé pour réindexer soit des livres particuliers ou la bibliothèque entière. Pour réindexer des livres particuliers spécifiez les identifiants des livres comme arguments supplémentaires après la commande reindex. Si aucun identifiant de livre n'est spécifié, la bibliothèque entière est réindexée.

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--indexing-speed

La vitesse d'indexation. Utiliser rapide pour une indexation rapide utilisant toutes les ressources de votre ordinateur et lent pour une indexation moins gourmande en ressources. Noter que la vitesse est réinitialisée à lent après chaque demande.

--wait-for-completion

Attendre que tous les livres soient indexés, en montrant périodiquement la progression de l'indexation

fts_search

```
calibredb fts_search [options] expression de recherche
```

Effectue une recherche intégrale du texte sur l'ensemble de la bibliothèque ou sur un sous-ensemble de celle-ci

Chaque fois que vous passez à calibredb des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

--do-not-match-on-related-words

La correspondance se fait uniquement sur les mots exacts et non sur les mots apparentés. La correction ne correspondra donc pas à corrigé.

--include-snippets

Inclure des extraits du texte entourant chaque correspondance. Noter que cela rend la recherche beaucoup plus lente.

--indexing-threshold

Pourcentage de la bibliothèque qui doit être indexée avant que la recherche ne soit autorisée. La valeur par défaut est 90

--match-end-marker

Le marqueur utilisé pour indiquer la fin d'un mot correspondant à l'intérieur d'un extrait

--match-start-marker

Le marqueur utilisé pour indiquer le début d'un mot correspondant dans un extrait

--output-format

Le format de sortie des résultats de la recherche. Soit "text" pour du texte brut, soit "json" pour une sortie JSON.

--restrict-to

Restreindre les livres recherchés, en utilisant une expression de recherche ou des identifiants. Par exemple : ids :1,2,3 pour restreindre par ids ou search :tag :foo pour restreindre aux livres ayant le tag foo.

13.1.7 ebook-convert

```
ebook-convert input_file output_file [options]
```

Convertir un livre numérique d'un format à un autre.

input_file est l'entrée output_file est la sortie. Chacun des deux doit être spécifié comme premiers arguments de la commande.

Le format de livre numérique de sortie est défini par l'extension du fichier output_file. output_file peut également être un format spécial .EXT où EXT est l'extension du fichier de sortie. Dans ce cas, le nom du fichier de sortie est dérivé du nom du fichier d'entrée. Notez que les noms de fichier ne doivent pas commencer par un trait d'union. Pour finir, si output_file n'a pas d'extension, il est traité comme un dossier et un « open e-book » (OEB) se composant des fichiers HTML est écrit dans ce dossier. Ces fichiers sont les fichiers qui auraient été normalement traité par l'extension de sortie.

Après avoir spécifié le fichier d'entrée et de sortie vous pouvez adapter la conversion en spécifiant diverses options. Les options disponibles dépendent des types de fichier d'entrée et de sortie. Pour obtenir de l'aide sur celles-ci, spécifiez le fichier d'entrée et de sortie et puis employez l'option -h.

Pour la documentation complète du système de conversion voir *Conversion de livre numérique* (page 61)

Chaque fois que vous passez à **ebook-convert** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

Les options et les valeurs par défaut pour les options changent selon les formats d'entrée et de sortie, aussi vous devriez toujours vérifier avec

```
ebook-convert myfile.input_format myfile.output_format -h
```

Ci-dessous se trouvent les options communes à toutes les conversions, suivies par les options spécifiques à tous les formats d'entrée et de sortie.

- *OPTIONS D'ENTRÉE* (page 340)
- *OPTIONS DE SORTIE* (page 340)
- *ASPECT ET COMPORTEMENT* (page 341)
- *TRAITEMENT HEURISTIQUE* (page 344)
- *RECHERCHER ET REMPLACER* (page 345)
- *DÉTECTION DE LA STRUCTURE* (page 345)
- *TABLE DES MATIERES* (page 346)
- *METADONNEES* (page 347)
- *DÉBOGAGE* (page 348)

--help, -h

affiche ce message d'aide et quitte

--input-profile

Spécifier le profil d'entrée. Ce profil d'entrée indique au système de conversion comment interpréter les diverses informations dans le document d'entrée. Par exemple, la résolution dépend des longueurs. (c.-à-d. longueurs en pixels). Les choix sont : cybookg3, cybook_opus, default, hanlinv3, hanlinv5, illiad, irexdr1000, irexdr800, kindle, msreader, mobipocket, nook, sony, sony300, sony900

--list-recipes

Liste les noms des recettes intégrées. Vous pouvez créer un livre numérique depuis une recette intégrée de la manière suivante : ebook-convert "Nom Recette.recipe" sortie.epub

--output-profile

Spécifiez le profil de sortie. Le profil de sortie explique au système de conversion comment optimiser les documents créés pour un périphérique donné. Dans certains cas, un profil de sortie peut être utilisé pour optimiser la sortie pour un périphérique bien particulier, mais cela est rarement nécessaire. Les choix sont :cybookg3, cybook_opus, default, generic_eink, generic_eink_hd, generic_eink_large, hanlinv3, hanlinv5, illiad, ipad, ipad3, irexdr1000, irexdr800, jetbook5, kindle, kindle_dx, kindle_fire, kindle_oasis, kindle_pw, kindle_pw3, kindle_scribe, kindle_voyage, kobo, msreader, mobipocket, nook, nook_color, nook_hd_plus, pocketbook_inkpad3, pocketbook_lux, pocketbook_hd, pocketbook_900, pocketbook_pro_912, galaxy, sony, sony300, sony900, sony-landscape, sonyt3, tablet

--version

affiche le numéro de version du logiciel et quitte

OPTIONS D'ENTRÉE

Options pour contrôler le traitement du fichier d'entrée mobi

--input-encoding

Spécifier l'encodage des caractères du document d'entrée. Si réglée cette option outrepassera tout encodage déjà déclaré par le document lui-même. Particulièrement utile pour des documents ne déclarant pas d'encodage ou ayant des déclarations d'encodage incorrectes.

OPTIONS DE SORTIE

Options pour contrôler le traitement de la sortie epub

--dont-split-on-page-breaks

Désactiver le sectionnement aux sauts de page. En temps normal, le fichier d'entrée est sectionné à chaque saut de page pour former deux fichiers ; vous bénéficiez ainsi d'un livre numérique susceptible d'être analysé plus rapidement et avec moins de ressources. Toutefois, sectionner les fichiers est un processus lent, donc il est recommandé de désactiver le sectionnement si le fichier d'entrée contient de nombreux sauts de page.

--epub-flatten

Cette option est nécessaire si vous avez l'intention d'utiliser l'EPUB avec FBReaderJ. Elle va aplatir la structure du système de fichiers à l'intérieur de l'EPUB, mettant tous les fichiers sur le niveau supérieur.

--epub-inline-toc

Insérer une Table des Matières intégrée qui apparaîtra en tant qu'élément du contenu principal du livre.

--epub-max-image-size

Taille maximale de l'image (largeur x hauteur). Une valeur de none signifie que sera utilisé la taille d'écran du profil de sortie. Une valeur de profile signifie qu'aucune taille maximale n'est spécifiée. Par exemple, une valeur de 100x200 entraînera le redimensionnement de toutes les images de manière à ce que leur largeur ne dépasse pas 100 pixels et leur hauteur 200 pixels. Notez que cela n'affecte que la taille des fichiers d'image eux-mêmes. Une image donnée peut être rendue à une taille différente en fonction du style qui lui est appliqué dans le document.

--epub-toc-at-end

Mettre la Table des Matières intégrée à la fin du livre plutôt qu'au début.

--epub-version

La version du fichier EPUB à générer. EPUB 2 est le plus largement compatible, n'utilisez EPUB 3 que si vous savez que vous en avez réellement besoin.

--extract-to

Extrait le contenu du fichier EPUB généré vers le dossier spécifié. Le contenu du dossier sera d'abord effacé, donc faites attention.

--flow-size

Diviser tous les fichiers HTML plus grand que cette taille (en Ko). Ceci est nécessaire car la plupart des lecteurs EPUB ne prennent pas en charge des fichiers de grande taille. Par défaut 260 Ko est la taille requise par Adobe Digital Editions. Paramétrer à 0 pour désactiver la division basée sur la taille.

--no-default-epub-cover

Normalement, si le fichier d'entrée n'a pas de couverture et que vous n'en avez pas spécifié une, une couverture par défaut est générée avec le titre, les auteurs, etc. Cette option désactive la génération de cette couverture.

--no-svg-cover

Ne pas utiliser le format SVG pour la couverture du livre. Utiliser cette option si votre EPUB va être utilisé sur un périphérique qui ne prend pas en charge SVG, comme l'iPhone ou le JetBook Lite. Sans cette option, ce type périphérique affichera une page blanche comme couverture.

--preserve-cover-aspect-ratio

Lors de l'utilisation d'une image SVG en couverture, cette option va entraîner une mise à l'échelle permettant de couvrir tout l'écran, mais va toujours garder les proportions (ratio hauteur/largeur) de l'image d'origine. Ceci signifie qu'il peut y avoir des bordures blanches sur les cotés, en haut ou en bas de l'image, mais que celle-ci ne sera jamais distordue. Sans cette option l'image peut être légèrement distordue, mais il n'y aura pas de bordures.

--pretty-print

Si spécifié, l'extension de sortie essaiera de créer une sortie qui sera aussi lisible que possible pour un être humain. Peut être sans effet avec certaines extensions de sortie.

--toc-title

Titre pour toutes les Tables des Matières intégrées générées.

ASPECT ET COMPORTEMENT

Options pour contrôler l'apparence de la sortie

--asciiize

Translittère les caractères Unicode en une représentation ASCII. A utiliser avec précaution car cela remplacera les caractères Unicode par des caractères ASCII. Par exemple, elle remplacera "Pelé" par "Pele". Notez également que dans les cas où il existe plusieurs représentations d'un caractère (caractères partagés par le chinois et le japonais par exemple), la représentation basée sur la langue de l'interface actuelle de calibre sera utilisée.

--base-font-size

La taille de base pour la police de caractère en pts. Toutes les tailles de police dans le livre produit seront redimensionnées par rapport à cette taille. En choisissant une taille large, vous pouvez obtenir des polices plus grandes en sortie et vice versa. Par défaut, quand la valeur est zéro, la taille de base pour la police est choisie par rapport au profil de sortie que vous avez choisi.

--change-justification

Changer la justification du texte. Une valeur à "left" convertit tout le texte justifié dans la source vers un alignement de texte à gauche (c.à.d non justifié). Une valeur à "justify" convertit le texte non justifié en justifié. Une valeur à "original" (par défaut) ne change pas la justification du fichier source. Notez que seuls certains formats prennent en charge la justification.

--disable-font-rescaling

Désactiver tous les redimensionnements des tailles de polices.

--embed-all-fonts

Incorporer toutes les polices qui sont référencées dans le document mais pas encore intégrées. Une recherche sera effectuée sur votre ordinateur et si les polices sont retrouvées, elles seront intégrées dans le document. L'intégration fonctionnera uniquement si le format vers lequel vous convertissez supporte les polices intégrées, telles que EPUB, AZW3, DOCX et PDF. Veuillez vous assurer que vous possédez la licence appropriée pour incorporer les polices utilisées dans ce document.

--embed-font-family

Inclut la famille de polices spécifiée dans le livre. Ceci précise la police de "base" utilisée pour le livre. Si le document d'entrée spécifie ses propres polices, elles peuvent primer sur la police de "base". Vous pouvez utilisez l'option de filtre de style pour enlever les polices du document d'entrée. Notez que l'inclusion de polices fonctionne seulement avec certains formats de sortie, principalement l'EPUB,AZW3 et DOCX.

--expand-css

Par défaut, calibre va utiliser la forme abrégée des diverses propriétés CSS comme les marges, les bordures, etc. Cette option permet d'utiliser la forme étendue à la place. Notez que le CSS sera toujours sous sa forme étendue lors de la génération de fichiers EPUB avec le profil de sortie définit pour Nook car le Nook ne sait pas gérer la forme abrégée du CSS.

--extra-css

Soit le chemin d'une feuille de style CSS ou un CSS brut. Ce CSS sera ajouté aux règles de style du fichier source, ainsi il pourra être utilisé pour outrepasser ces règles.

--filter-css

Une liste de propriétés CSS, séparées par des virgules, qui seront supprimées de toutes les règles de style CSS. Ceci est utile si la présence de certaines informations de style les empêche d'être ignorées par votre périphérique. Par exemple : font-family,color,margin-left,margin-right

--font-size-mapping

Correspondance à partir de noms de police CSS vers des tailles de polices en pts. Un exemple de paramétrage est : 12,12,14,16,18,20,22,24. Ce sont les correspondances pour les tailles de xx-small à xx-large, la taille finale est utilisée pour les très grandes polices. Le redimensionnement de la police utilise ces tailles pour redimensionner intelligemment les polices. Le comportement par défaut est d'utiliser une correspondance basée sur le profil de sortie que vous avez choisi.

--insert-blank-line

Insérer une ligne blanche entre les paragraphes. Ne fonctionnera pas si le fichier source n'utilise pas de paragraphes. (balises <p> ou)

--insert-blank-line-size

Définir la hauteur des lignes blanches insérées (en em). La hauteur des lignes entre deux paragraphes sera de deux fois la valeur spécifiée ici.

--keep-ligatures

Conserver les ligatures présentes dans le document d'entrée. Une ligature est d'une paire de caractères comme ff, fi, fl, œ, æ et cætera. La plupart des polices par défaut des lecteurs ne prennent pas en charge les ligatures, aussi un rendu correct de celles-ci semble improbable sur le lecteur. Par défaut, calibre va transformer une ligature en la paire de caractères correspondants. A l'opposé, cette option va conserver la ligature.

--line-height

La hauteur de ligne en pts. Contrôle l'espacement entre les lignes consécutives du texte. Ne s'applique qu'aux éléments qui ne définissent pas leur propre hauteur de ligne. Dans la plupart des cas, l'option hauteur de ligne minimum est plus utile. Par défaut, aucune modification de hauteur de ligne n'est effectuée.

--linearize-tables

Certains documents mal conçus utilisent les tableaux pour contrôler l'affichage du texte dans la page. Quand ils sont convertis, ces documents ont souvent le texte qui déborde de la page et d'autres dysfonctionnements. Cette option extraira le contenu des tableaux et le présentera dans un mode linéaire.

--margin-bottom

Régler la marge du bas en pts. Le réglage par défaut est 5.0. Aucune marge ne sera définie si le nombre est inférieur à zéro (le réglage des marges dans le document d'origine est préservé). Notez : les formats de page orientés tels que PDF et DOCX ont leurs propres paramétrages de marges qui ont la priorité.

--margin-left

Régler la marge gauche en pts. Le réglage par défaut est 5.0. Aucune marge ne sera définie si le nombre est inférieur à zéro (le réglage des marges dans le document d'origine est préservé). Notez : les formats de page orientés tels que PDF et DOCX ont leurs propres paramétrages de marge qui ont la priorité.

--margin-right

Régler la marge droite en pts. Le réglage par défaut est 5.0. Aucune marge ne sera définie si le nombre est inférieur à zéro (le réglage des marges dans le document d'origine est préservé). Notez : les formats de page orientés tels que PDF et DOCX ont leurs propres paramétrages de marge qui ont la priorité.

--margin-top

Régler la marge haute en pts. Le réglage par défaut est 5.0. Aucune marge ne sera définie si le nombre est inférieur à zéro (le réglage des marges dans le document d'origine est préservé). Notez : les formats de page orientés tels que PDF et DOCX ont leurs propres paramétrages de marge qui ont la priorité.

--minimum-line-height

La hauteur de ligne minimum, en pourcentage de la taille de la police de l'élément calculé. calibre s'assurera que chaque élément a une hauteur de ligne correspondant au moins à ce réglage, quelle que soit la spécification du document d'entrée. Fixer à 0 pour désactiver. Par défaut 120%. Utiliser cette option préférentiellement à une spécification de ligne directe, à moins de savoir ce que vous faites. Par exemple, vous pouvez obtenir un texte en "double interligne" en fixant cette valeur à 240%.

--remove-paragraph-spacing

Supprimer l'interligne entre les paragraphes. Paramètre aussi une indentation sur les paragraphes de 1.5em. La suppression de l'interligne ne fonctionnera pas si le fichier source n'utilise pas les paragraphes (balises <p> ou).

--remove-paragraph-spacing-indent-size

Lorsque calibre supprime les lignes blanches entre les paragraphes, il ajoute automatiquement une indentation de paragraphe, afin de s'assurer que les paragraphes puissent être facilement distingués. Cette option contrôle la largeur de cette indentation (en em). Si vous fixez une valeur négative, alors l'indentation spécifiée dans le document d'entrée est utilisée, c'est à dire que calibre ne modifie pas l'indentation.

--smarten-punctuation

Convertit les guillemets, les tirets et les points de suspension en leurs équivalents typographiques corrects. Pour plus de détails, voir https://daringfireball.net/projects/smartypants.

--subset-embedded-fonts

Créer un sous-ensemble de toutes les polices intégrées. Chaque police intégrée est réduite pour ne contenir que les caractères utilisés dans le document. Cela réduit la taille des fichiers associés aux polices. Utile si vous souhaitez intégrer des polices particulièrement grandes dont peu de glyphes sont utilisés.

--transform-css-rules

Chemin vers un fichier contenant les règles pour transformer les styles CSS dans ce livre. La manière la plus facile de créer un tel fichier est d'utiliser l'assistant pour la création de règles dans l'interface graphique utilisateur de calibre. Accédez-y dans la section de la boite de dialogue conversion sous "Apparence->Transformer les styles".

Une fois fois que vous avez créé les règles, vous pouvez utiliser le bouton Exporter pour les sauvegarder en un fichier.

--transform-html-rules

Chemin vers un fichier contenant les règles pour transformer du HTML dans ce livre. La manière la plus facile de créer un tel fichier est d'utiliser l'assistant pour la création de règles dans l'interface graphique utilisateur de calibre. Accédez-y dans la section de la boite de dialogue conversion sous "Apparence->Transformer du HTML". Une fois fois que vous avez créé les règles, vous pouvez utiliser le bouton "Exporter" pour les sauvegarder en un fichier.

--unsmarten-punctuation

Convertir les guillemets fantaisie, les tirets et les points de suspension en leurs équivalents en texte brut.

TRAITEMENT HEURISTIQUE

Modifier le texte et la structure du document selon le modèle courant. Désactivé par défaut. Utiliser –enable-heuristics pour l'activer. Les actions individuelles peuvent être désactivées avec les options –disable-*.

--disable-dehyphenate

Analyser les mots contenant des traits d'union dans le document. Le document lui-même est utilisé comme dictionnaire pour déterminer si les traits d'union doivent être conservés ou retirés.

--disable-delete-blank-paragraphs

Supprimer les paragraphes vides du document quand ils se trouvent entre d'autres paragraphes

--disable-fix-indents

Transformer les retraits de ligne créées par de multiples espaces insécables en retraits CSS.

--disable-format-scene-breaks

Les marqueurs de rupture de scène alignés à gauche sont alignés au centre. Les ruptures de scène qui utilisent des lignes blanches multiples sont remplacées par des traits horizontaux.

--disable-italicize-common-cases

Rechercher les mots et les motifs usuels qui dénotent la présence de mots en italiques et les mettre en italique.

--disable-markup-chapter-headings

Détecter les titre de chapitre et les sous-titres sans mise en forme et les transforme en balises h2 et h3. Ce réglage ne va pas créer de Table des Matières, mais peut être utilisé dans la partie Détection de structure afin d'en créer une.

--disable-renumber-headings

Recherche les occurrences de séquences de balises <h1> ou <h2>. Les balises sont renumérotées afin de d'éviter que les titres de chapitres ne soient coupés en deux.

--disable-unwrap-lines

Redéfinir la longueur des lignes en se basant sur la ponctuation et autres indices de formatage.

--enable-heuristics

Activer le traitement heuristique. Cette option doit être activée pour que tout traitement heuristique puisse être effectué.

--html-unwrap-factor

Échelle utilisée pour déterminer la longueur à partir de laquelle une ligne doit être redéfinie. Les valeurs acceptables sont les décimaux compris entre 0 et 1. La valeur pas défaut est de 0.4, juste en dessous de la valeur médiane des longueurs de ligne dans le document. Si seules quelques lignes dans le document nécessitent d'être redéfinies, cette valeur devrait être réduite.

--replace-scene-breaks

Remplacer les ruptures de scène par le texte spécifié. Par défaut, le texte du document initial est utilisé.

RECHERCHER ET REMPLACER

Modifier le texte et la structure du document en utilisant des modèles définis par l'utilisateur.

--search-replace

Chemin vers un fichier contenant les expressions régulières de recherche et remplacement. Le fichier doit contenir en alternances des lignes d'expressions régulières suivies par des modèles de remplacement (qui peuvent être une ligne vide). L'expression régulière doit être écrite dans la syntaxe des regexp Python et le fichier doit être encodé en UTF-8.

--sr1-replace

Le remplacement pour remplacer le texte trouvé avec sr1-search.

--sr1-search

Motif de recherche (expression régulière) qui sera remplacé par sr1-replace.

--sr2-replace

Le remplacement pour remplacer le texte a trouvé avec sr2-search.

--sr2-search

Modèle de recherche (expression régulière) qui sera remplacé par sr2-replace.

--sr3-replace

Le remplacement pour remplacer le texte a trouvé avec sr3-search.

--sr3-search

Modèle de recherche (expression régulière) qui sera remplacée par sr3-replace.

DÉTECTION DE LA STRUCTURE

Contrôle de l'auto-détection de la structure du document.

--chapter

Une expression XPath pour détecter les titres de chapitres. Par défaut, les étiquettes <h1> ou <h2> qui contiennent les mots "chapter", "book", "section", "prologue", "epilogue", ou "part" ainsi que les étiquettes qui contiennent class="chapter". L'expression utilisée doit être évaluée comme une liste d'éléments. Pour désactiver la détection de chapitre, utiliser l'expression "/". Regarder le tutoriel XPath dans le Manuel Utilisateur de calibre pour plus d'aide sur l'utilisation de cette fonction.

--chapter-mark

Spécifier comment indiquer les chapitres détectés. Une valeur "pagebreak" insérera un saut de page avant chaque chapitre. Une valeur "rule" insérera un filet avant chaque chapitre. Une valeur "none" désactivera le marquage des chapitres et une valeur "both" utilisera à la fois un saut de page et un filet.

--disable-remove-fake-margins

Certains documents spécifient les marges de la page en précisant une marge gauche et droite sur chaque paragraphe individuel. calibre tentera de détecter et de supprimer ces marges. Cela peut éventuellement provoquer la suppression de certaines marges qui ne devaient pas être retirées. Dans ce cas, vous pouvez désactiver la suppression de ces marges.

--insert-metadata

Insérer les métadonnées au début du livre. Utile si votre lecteur de livre numérique ne prend pas en charge directement l'affichage/recherche des métadonnées.

--page-breaks-before

Une expression XPath. Des sauts de page sont insérés avant les éléments spécifiés. Pour désactiver utiliser l'expression : /

--prefer-metadata-cover

Utiliser la couverture détectée dans le fichier source de préférence à la couverture spécifiée.

--remove-first-image

Supprimer la première image du livre numérique en entrée. Utile si le document à transformer a une image de couverture qui n'est pas identifiée comme une couverture. Dans ce cas, si vous avez une image de couverture dans calibre, le document de sortie aura deux images de couverture si vous n'utilisez pas cette option.

--start-reading-at

Expression XPath pour détecter l'emplacement du début de la lecture dans le document. Certains programmes de lecture de livre numérique (dont le plus connu est le Kindle) utilisent cette position comme emplacement d'ouverture du livre. Voir le Tutoriel XPath dans le Manuel d'utilisation de calibre pour une aide supplémentaire concernant cette fonctionnalité.

TABLE DES MATIERES

Contrôle la génération automatique de la Table des Matières. Par défaut, si le fichier source a une Table des Matières, elle sera utilisée de préférence par rapport à celle qui est générée automatiquement.

--duplicate-links-in-toc

Lors de la création d'une Table des Matières à partir de liens dans le document d'entrée, autorise des entrées dupliquées, à savoir autorise plus d'une entrée avec le même texte, du moment qu'elles pointent vers des emplacements différents.

--level1-toc

Expression XPath indiquant toutes les étiquettes qui doivent être ajoutées au premier niveau de la Table des Matières. Si elle est spécifiée, elle sera prioritaire par rapport aux autres formes d'auto-détection. Voir le Tutoriel XPath dans le Manuel Utilisateur calibre pour des exemples.

--level2-toc

Expression XPath indiquant que toutes les étiquettes doivent être ajoutées au deuxième niveau de la table des matières. Chaque entrée est ajoutée en dessous de la précédente entrée de premier niveau. Voir le Tutoriel XPath dans le Manuel Utilisateur calibre pour des exemples.

--level3-toc

Expression XPath indiquant que toutes les étiquettes doivent être ajoutées au troisième niveau de la table des matières. Chaque entrée est ajoutée en dessous de la précédente entrée de second niveau. Voir le Tutoriel XPath dans le Manuel Utilisateur calibre pour des exemples.

--max-toc-links

Nombre maximum de liens à insérer dans la table des matières. Régler à 0 pour désactiver. Par défaut : 50. Les liens sont ajoutés à la Table des Matières seulement si le seuil du nombre de chapitres détectés n'a pas été atteint.

--no-chapters-in-toc

Ne pas ajouter à la Table des Matières les chapitres détectés automatiquement.

--toc-filter

Supprimer les entrées de la Table des Matières qui correspondent à l'expression régulière spécifiée. Les entrées correspondantes ainsi que leurs sous-entrées seront supprimées.

--toc-threshold

Lorsque le nombre de chapitres détectés est inférieur à ce chiffre, les liens sont ajoutés à la Table des Matières. Par défaut : 6

--use-auto-toc

Normalement, si le fichier source a déjà une Table des Matières, elle est utilisée de préférence à celle auto-générée. Avec cette option, l'auto-générée est toujours utilisée.

METADONNEES

Options pour indiquer les métadonnées en sortie

--author-sort

Chaîne à utiliser lors du tri par auteur.

--authors

Indiquer les auteurs. Les auteurs multiples doivent être séparés par des & (esperluètes).

--book-producer

Déterminer le producteur du livre.

--comments

Indiquer la description du livre numérique.

--cover

Déterminer la couverture à partir du fichier spécifié ou de l'URL

--isbn

Déterminer l'ISBN du livre.

--language

Déterminer la langue.

--pubdate

Défini la date de publication (supposée être dans le fuseau horaire local, sinon le fuseau horaire est explicitement spécifié)

--publisher

Indiquer l'éditeur du livre numérique.

--rating

Déterminer la note. Devrait être un nombre entre 1 et 5.

--read-metadata-from-opf, --from-opf, -m

Lire les métadonnées du fichier OPF spécifié. Les métadonnées lues à partir de ce fichier écraseront les métadonnées dans le fichier source.

--series

Indiquer la série à laquelle appartient ce livre numérique.

--series-index

Déterminer l'index de ce livre numérique dans cette série.

--tags

Déterminer les étiquettes du livre. Doit être une liste séparée par des virgules.

--timestamp

Définir l'horodatage (plus utilisé nulle part)

--title

Indiquer le titre.

--title-sort

La version du titre à utiliser pour le tri.

DÉBOGAGE

Options pour aider au débogage de la conversion

--debug-pipeline, -d

Enregistre le résultat des différents cycles du processus de conversion dans le dossier spécifié. Utile si vous ne savez pas à quel moment du cycle de conversion le bug est généré.

--verbose, -v

Niveau de verbosité. Spécifier des périodes multiples pour une plus grande verbosité. La spécification par deux fois aura comme conséquence la pleine verbosité, la verbosité par une fois la moyenne et les périodes zéro moins de verbosité.

13.1.8 ebook-edit

```
ebook-edit [opts] [path_to_ebook] [name_of_file_inside_book ...]
```

Lance l'outil Edition de livre calibre. Vous pouvez optionnellement spécifier également les noms des fichiers dans le livre qui devront être ouverts automatiquement pour l'édition.

Chaque fois que vous passez à **ebook-edit** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--detach

Détacher du terminal de contrôle, s'il y en a un (linux uniquement)

--help, -h

affiche ce message d'aide et quitte

--select-text

Le texte à sélectionner dans le livre lorsqu'il est ouvert pour être édité

--version

affiche le numéro de version du logiciel et quitte

13.1.9 ebook-meta

```
ebook-meta ebook_file [options]
```

Lire/Écrire des métadonnées de/vers des fichiers de livre numérique.

Formats pris en charge pour la lecture des métadonnées : azw, azw1, azw3, azw4, cb7, cbc, cbr, cbz, chm, docx, epub, fb2, fbz, html, htmlz, imp, lit, lrf, lrx, mobi, odt, oebzip, opf, pdb, pdf, pml, pmlz, pobi, prc, rar, rb, rtf, snb, tpz, txt, txtz, updb, zip

Formats pris en charge pour l'écriture des métadonnées : azw, azw1, azw3, azw4, docx, epub, fb2, fbz, htmlz, lrf, mobi, odt, pdb, pdf, prc, rtf, tpz, txtz

Différents types de fichiers prennent en charge différentes sortes de métadonnées. Si vous essayer de définir certaines métadonnées sur un type de fichier qui ne le prend pas en charge, les métadonnées seront ignorées sans avertissement.

Chaque fois que vous passez à **ebook-meta** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--author-sort
: Chaine utilisée pour le tri par auteur. Si non spécifiée, et que le(s) auteur(s) sont spécifié(s), il sera généré automatiquement à partir de(s) (l'')auteur(s).

--authors, -a
: Définir les auteurs. Les auteurs multiples doivent être séparés par le caractère &. Les noms d'auteur devrait être dans l'ordre Prénom Nom.

--book-producer, -k
: Déterminer le producteur du livre.

--category
: Définir la catégorie du livre.

--comments, -c
: Indiquer la description du livre numérique.

--cover
: Définir la couverture pour le fichier spécifié.

--date, -d
: Définir la date de publication.

--from-opf
: Lire les métadonnées à partir du fichier OPF spécifié et les utiliser pour indiquer les méta-données du livre numérique. Les métadonnées spécifiées en ligne de commande écraseront les métadonnées lues à partir du fichier OPF.

--get-cover
: Obtenir la couverture du livre numérique et l'enregistrer sous le fichier spécifié.

--help, -h
: affiche ce message d'aide et quitte

--identifier

Définir les identifiants du livre, peut être spécifié à plusieurs reprises. Par exemple : *--identifier* (page 349) uri :https ://acme.com *--identifier* (page 349) isbn :12345. Pour supprimer un identifiant, laissez la valeur en blanc, *--identifier* (page 349) isbn :. Notez que pour les fichiers EPUB, un identifiant marqué comme identifiant de paquet ne peut pas être supprimé.

--index, -i

Déterminer l'index de ce livre numérique dans cette série.

--isbn

Déterminer l'ISBN du livre.

--language, -l

Déterminer la langue.

--lrf-bookid

Définir l'identifiant du livre dans les fichiers LRF

--publisher, -p

Indiquer l'éditeur du livre numérique.

--rating, -r

Déterminer la note. Devrait être un nombre entre 1 et 5.

--series, -s

Indiquer la série à laquelle appartient ce livre numérique.

--tags

Déterminer les étiquettes du livre. Doit être une liste séparée par des virgules.

--title, -t

Indiquer le titre.

--title-sort

La version du titre à utilisé pour le tri. Si non spécifiée, et que le titre est spécifié, il sera généré automatiquement à partir du titre.

--to-opf

Spécifier le nom du fichier OPF. Les métadonnées seront écrites dans le fichier OPF.

--version

affiche le numéro de version du logiciel et quitte

13.1.10 ebook-polish

```
ebook-polish [options] input_file [output_file]
```

Le Polissage des livres constitue la dernière touche de perfection pour finaliser les livres numériques que vous avez soigneusement élaborés.

Le polissage tente de minimiser les modifications apportées au code interne de votre livre numérique. Contrairement à la conversion, il n'aplatit pas les CSS, ne renomme pas les fichiers, ne change pas les tailles de police, n'ajuste pas les marges, etc… Chaque action exécute seulement le minimum de modifications nécessaires pour obtenir l'effet désiré.

Vous pouvez utiliser cet outil en dernière étape de votre processus de création de livre numérique.

Notez que le polissage ne fonctionne que sur les fichiers aux formats AZW3 ou EPUB.

Chaque fois que vous passez à **ebook-polish** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--add-soft-hyphens, -H

Ajouter des traits d'union conditionnels à tous les mots du livre. Cela permet au livre d'être mieux affiché lorsque le texte est justifié, pour les lecteurs ne supportant pas la césure.

--compress-images, -i

Compresser les images sans perte dans le livre, pour les réduire, sans affecter la qualité de l'image.

--cover, -c

Chemin vers une image de couverture. Change la couverture spécifiée dans le livre numérique. Si aucune couverture n'est présente ou si la couverture n'est pas identifiée correctement, insère une nouvelle couverture.

--embed-fonts, -e

Incorporer toutes les polices qui sont référencées dans le document mais pas encore intégrées. Une recherche sera effectuée sur votre ordinateur et si les polices sont trouvées, elles seront intégrées dans le document. Veuillez vous assurer que vous possédez la licence appropriée avant d'incorporer les polices utilisées dans ce document.

--help, -h

affiche ce message d'aide et quitte

--jacket, -j

Insérer une page "jaquette de livre" au début du livre, sur laquelle on peut lire les éléments de métadonnées tels que le titre, les étiquettes, les auteurs, les séries, les commentaires, etc… Toute jaquette du livre préexistante sera remplacée.

--opf, -o

Chemin vers un fichier OPF. Les métadonnées du livre sont mises à jour à partir du ficher OPF.

--remove-jacket

Enlever la page jaquette préexistante.

--remove-soft-hyphens

Supprime les traits d'union conditionnels de la totalité du texte du livre.

--remove-unused-css, -u

Supprimer toutes les règles CSS inutilisées des feuilles de style et des mots-clés <style> . Certains livres créés à partir de modèles de production peuvent comporter un grand nombre de règles CSS supplémentaires qui ne correspondent à aucun contenu présent. Ces règles supplémentaires peuvent ralentir certains lecteurs qui ont besoin de toutes les analyser.

--smarten-punctuation, -p

Convertir les tirets de texte brut, ellipses, guillemets, multiples traits d'union, etc… dans leurs équivalents typographiques corrects. Notez que l'algorithme peut parfois générer des résultats erronés, en particulier en présence d'apostrophes utilisées pour les contractions de mots.

--subset-fonts, -f

Créer un sous-ensemble de polices signifie réduire une police embarquée pour contenir uniquement les caractères utilisés de cette police dans le livre. Cela réduit considérablement la taille des fichiers de police (une réduction de moitié de la taille du fichier de police est fréquent). Par exemple, si le livre utilise une police spécifique pour les en-têtes, alors créer un sous-ensemble réduira la police pour ne contenir que les caractères réellement présents dans les en-têtes du livre. Ou si le livre intègre les versions en gras et italique d'une police, mais que le texte en italique ou en gras est relativement rare, voire inexistant, alors les caractères gras et en italiques peuvent être soit réduits

à seulement quelques caractères, voire complètement retirés. Le seul inconvénient des sous-ensembles de polices, c'est que si, à une date ultérieure, vous décidez d'ajouter du texte à vos livres, le texte nouvellement ajouté pourrait ne pas être pris en compte par la police réduite.

--upgrade-book, -U

Met à jour la structure interne du livre, si possible. Par exemple, mise à jour de livres EPUB 2 en livres EPUB 3.

--verbose

Générer une sortie plus détaillée (utile pour le débogage)

--version

affiche le numéro de version du logiciel et quitte

13.1.11 ebook-viewer

```
ebook-viewer [options] fichier
```

Visualiser un livre numérique.

Chaque fois que vous passez à **ebook-viewer** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--continue

Continuer à lire le dernier livre ouvert

--detach

Détacher du terminal de contrôle, s'il y en a un (linux uniquement)

--force-reload

Forcer le rechargement de tous les livres ouverts

--full-screen, --fullscreen, -f

Si précisé, la fenêtre de la visionneuse de livre numérique essaiera, au démarrage, de s'ouvrir en plein écran.

--help, -h

affiche ce message d'aide et quitte

--new-instance

Ouvrir une nouvelle fenêtre de visionneuse même si l'option d'utilisation d'une seule fenêtre de visionneuse est activée

--open-at

La position à laquelle ouvrir le livre spécifié. La position est un emplacement ou une position que vous pouvez obtenir en utilisant l'action Aller à->Location dans les contrôles de la visionneuse. Vous pouvez également utiliser le formulaire toc :quelque chose pour ouvrir le livre à l'emplacement de la première entrée de la table des matières qui contient la chaîne "quelque chose". Le formulaire toc-href :quelque chose correspondra au href (destination du lien interne) des nœuds toc. La correspondance est exacte. Si vous souhaitez faire correspondre une sous-chaîne, utilisez la forme toc-href-contains :quelque chose. Le formulaire ref :quelque chose utilisera Reference en mode référence. Le formulaire search :quelque chose recherchera quelque chose après avoir ouvert le livre. La forme regex :quelque chose recherchera l'expression régulière quelque chose après avoir ouvert le livre.

--raise-window

Si spécifié, la fenêtre de la visionneuse de livre numérique essaiera, au lancement, d'apparaître au premier plan.

--version
affiche le numéro de version du logiciel et quitte

13.1.12 fetch-ebook-metadata

```
fetch-ebook-metadata [options]
```

Récupérer des métadonnées de livre depuis des sources en ligne. Vous devez spécifier au moins le titre, les auteurs ou l'ISBN.

Chaque fois que vous passez à **fetch-ebook-metadata** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--allowed-plugin, -p
Spécifier le nom d'une extension de téléchargement des métadonnées à utiliser. Par défaut, toutes les extensions de métadonnées seront utilisées. Peut être spécifié de multiples fois et pour de multiples extensions. Tous les noms d'extensions : Google, Google Images, Amazon.com, Edelweiss, Open Library, Big Book Search

--authors, -a
Auteur(s) du livre

--cover, -c
Spécifier un nom de fichier. La couverture, si disponible, y sera sauvée. Sans cette option, aucune couverture ne sera téléchargée.

--help, -h
affiche ce message d'aide et quitte

--identifier, -I
Les identificateurs tels que ASIN/Goodreads, etc. Peuvent être spécifiés plusieurs fois pour plusieurs identificateurs. Par exemple : *--identifier* (page 353) asin :B0082BAJA0

--isbn, -i
ISBN du livre

--opf, -o
Produit les métadonnées dans le format d'OPF au lieu d'un texte lisible pour l'homme.

--timeout, -d
Délai d'attente en secondes. Par défaut est à 30

--title, -t
Titre du livre

--verbose, -v
Imprimer le journal vers la console (stderr)

--version
affiche le numéro de version du logiciel et quitte

13.1.13 lrf2lrs

```
lrf2lrs book.lrf
```

Convertir un fichier LRF en un fichier LRS (encodage XML UTF-8)

Chaque fois que vous passez à **lrf2lrs** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--dont-output-resources
: Ne pas enregistrer les images incorporées et les fichiers de polices sur le disque

--help, -h
: affiche ce message d'aide et quitte

--output, -o
: Fichier de sortie LRS

--verbose
: Soyez plus détaillé

--version
: affiche le numéro de version du logiciel et quitte

13.1.14 lrfviewer

```
lrfviewer [options] book.lrf
```

Lit le livre numérique LRF livre.lrf

Chaque fois que vous passez à **lrfviewer** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--disable-hyphenation
: Désactive la césure. Devrait accélérer le rendu de manière significative.

--help, -h
: affiche ce message d'aide et quitte

--profile
: Profiler le rendu LRF

--verbose
: Affiche plus d'information à propos du processus de rendu

--version
: affiche le numéro de version du logiciel et quitte

--visual-debug

Active les aides visuelles pour corriger le moteur de rendu

--white-background

Par défaut le fond est blanc cassé que je trouve plus facile pour les yeux. Employez cette option pour faire un fond blanc pur.

13.1.15 lrs2lrf

```
lrs2lrf [options] file.lrs
```

Compiler un fichier LRS en un fichier LRF.

Chaque fois que vous passez à **lrs2lrf** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--help, -h

affiche ce message d'aide et quitte

--lrs

Convertir LRS en LRS, utile pour déboguer.

--output, -o

Chemin vers le fichier de sortie

--verbose

Traitement explicatif

--version

affiche le numéro de version du logiciel et quitte

13.1.16 web2disk

```
web2disk URL
```

Où URL est par exemple https://google.com

Chaque fois que vous passez à **web2disk** des arguments comportant des espaces, entourez l'argument à l'aide de guillemets. Par exemple : « /some path/with spaces »

[options]

--base-dir, -d

Dossier de base dans lequel l'URL est sauvegardée. Par défaut : .

--delay

Intervalle minimal entre deux téléchargements, en secondes. 0 par défaut.

--dont-download-stylesheets

Ne pas télécharger les feuilles de style CSS.

--encoding

Encodage des caractères pour les sites web que vous essayer de télécharger. Par défaut, essaie de trouver l'encodage.

--filter-regexp

N'importe quel lien correspondant à cette expression régulière sera ignoré. Cette option peut-être activée plusieurs fois, auquel cas n'importe quel lien sera ignoré aussi longtemps qu'il correspond à une expression régulière. Par défaut, aucun lien n'est ignoré. Si l'expression régulière filtre et correspondance sont spécifiées, alors l'expression régulière filtre est appliquée en premier.

--help, -h

affiche ce message d'aide et quitte

--match-regexp

Seuls les liens correspondant à cette expression régulière seront suivis. Cette option peut être spécifiée plusieurs fois auquel cas tout lien qui correspondra à cette expression sera suivi. Par défaut, tous les liens sont suivis.

--max-files, -n

Nombre maximum de fichiers à télécharger. Ceci ne sera valable que pour les fichiers à partir des balises <a href>.Par défaut : 9223372036854775807

--max-recursions, -r

Nombre maximum de niveaux de récursivité c-à-d le niveau de profondeur des liens à suivre. Par défaut : 1

--timeout, -t

Délai d'attente d'une réponse du serveur, en secondes. Par défaut : 10.0 s

--verbose

Affiche des informations détaillées. Utile pour le débogage

--version

affiche le numéro de version du logiciel et quitte

13.2 Commandes non documentées

- ebook-device
- markdown-calibre

Vous pouvez voir l'utilisation des commandes non documentées en les exécutants sans arguments dans un terminal

CHAPITRE 14

Mise en place d'un environnement de développement calibre

calibre est complètement open source, sous licence GNU GPL v3[107]. Cela signifie que vous êtes libre de télécharger et de modifier le programme à votre convenance. Dans cette section, vous apprendrez comment obtenir la mise en place d'un environnement de développement calibre sur le système d'exploitation de votre choix. calibre est principalement écrit en Python[108] avec un peu de code C/C++ pour la vitesse et l'interfaçage système. Notez que calibre requiert au moins Python 3.8.

Contenu

- *Philosophie de conception* (page 358)
  - *Disposition du code* (page 358)
- *Obtenir le code* (page 359)
  - *Soumettre vos changements à inclure* (page 360)
- *Environnement de développement Windows* (page 361)
- *Environnement de développement macOS* (page 361)
- *Environnement de développement Linux* (page 362)
- *Avoir des installations séparées de calibre « normale » et « développement » sur le même ordinateur* (page 363)
- *Astuces de dépannage* (page 363)
  - *En utilisant les déclarations d'impression* (page 363)
  - *Utilisation d'un interpréte Python interactif* (page 364)
  - *Utiliser le débogueur de Python comme débogueur à distance* (page 364)
  - *Utiliser le débogueur dans votre IDE Python préféré* (page 365)
  - *Exécuter des scripts quelconques dans l'environnement Python calibre* (page 365)

[107] https://www.gnu.org/copyleft/gpl.html
[108] https://www.python.org

- *Utilisation de calibre dans vos projets* (page 365)
  - *Installation binaire de calibre* (page 365)
  - *Installez les sources sur Linux* (page 366)
- *Documentation API pour différentes parties de calibre* (page 366)

14.1 Philosophie de conception

calibre a ses racines dans le monde UNIX, ce qui signifie que sa conception est très modulaire. Les modules interagissent les uns avec les autres par l'intermédiaire d'interfaces bien définies. C'est ce qui rend l'ajout de nouvelles fonctionnalités et la réparation des dysfonctionnements très faciles dans calibre, ayant pour résultat un rythme de développement frénétique. En raison de ses racines,calibre a une interface en ligne de commande complète pour toutes ses fonctions, documentée dans doc :*generated/fr/cli-index*.

La conception modulaire de calibre est exprimée par l'intermédiaire des `Extensions`. Il y a un *tutoriel* (page 267) sur l'écriture d'extensions calibre. Par exemple, ajouter le support d'un nouveau périphérique à calibre implique typiquement d'écrire moins de 100 lignes de code sous forme d'une extension pilote de périphérique. Vous pouvez parcourir les pilotes intégrés[109]. Similairement, ajouter le support pour de nouveaux formats de conversion implique l'écriture de nouvelles extensions de format entrée/sortie. Un autre exemple de la conception modulaire est le *système de recette* (page 33) pour récupérer des actualités. Pour plus d'exemples d'extensions conçues pour ajouter des fonctionnalités à calibre, voir l'Index des extensions[110].

14.1.1 Disposition du code

Tout le code Python de calibre est dans le paquet `calibre`. Ce paquet contient les principaux sous-paquets suivants :

- périphériques - Tous les pilotes de périphériques. Regardez juste certains des pilotes intégrés pour avoir une idée de la façon dont ils fonctionnent.
  - Pour plus de détails, voir : `devices.interface` qui définit l'interface supportée par les pilotes de périphériques et `devices.usbms` qui définit un pilote générique qui connecte un périphérique USBMS. Tous les pilotes basés USBMS dans calibre héritent de lui.
- livres numériques - Tout le code conversion/métadonnées de livre numérique. Un bon point de départ est `calibre.ebooks.conversion.cli` qui est le module alimentant la commande **`ebook-convert`**. Le processus de conversion est contrôlé par l'intermédiaire de `conversion.plumber`. Le code de format indépendant est entièrement dans `ebooks.oeb` et le code de format dépendant est dans `ebooks.format_name`.
  - La lecture, l'écriture et le téléchargement des métadonnées est entièrement dans `ebooks.metadata`
  - La conversion se produit dans un pipeline, pour la structure du pipeline, voir *Introduction* (page 63). Le pipeline consiste en une extension d'entrée, différentes transformations et une extension de sortie. Le code qui construit et pilote le pipeline est dans `plumber.py`. Le pipeline fonctionne sur une représentation d'un livre numérique qui est comme un epub décompressé, avec le manifeste, la structure organisationnelle, la TdM, le guide, le contenu html, etc. La classe qui gère cette représentation est OEBBook dans `ebooks.oeb.base`. Les diverses transformations qui sont appliquées au livre durant la conversion se situent dans `oeb/transforms/*.py`. Et les extensions d'origines et de sortie se situent dans `conversion/plugins/*.py`.

[109] https://github.com/kovidgoyal/calibre/tree/master/src/calibre/devices
[110] https://www.mobileread.com/forums/showthread.php?p=1362767#post1362767

  - L'édition de livre numérique se produit en utilisant un objet conteneur différent. Il est documenté dans *Documentation API pour l'outil d'édition de livre numérique* (page 376).

- db - Le back-end de la base de données. Voir *Documentation API pour l'interface base de données* (page 366) pour l'interface à la bibliothèque calibre.
- Serveur de contenu : `srv` est le Serveur de contenu calibre.
- gui2 - L'Interface Graphique Utilisateur. L'initialisation du GUI se produit dans `gui2.main` et `gui2.ui`. La visionneuse de livre numérique dans `gui2.viewer`. L'éditeur de livre numérique dans `gui2.tweak_book`.

Si vous voulez localiser les points d'entrée de tous les différents exécutables de calibre, regardez à la structure `entry_points` dans linux.py[111].

Si vous avez besoin d'aide pour comprendre le code, postez sur le forum de développement[112] et vous obtiendrez très probablement de l'aide de l'un des nombreux développeurs de calibre.

14.2 Obtenir le code

Vous pouvez obtenir le code source de calibre de deux manières soit en utilisant un système de contrôle de version soit en téléchargeant directement une tarball[113].

calibre utilise Git[114], un système de contrôle de version distribuée. Git est disponible sur toutes les plateformes supportées par calibre. Après avoir installé Git, vous pouvez obtenir le code source de calibre avec la commande :

```
git clone https://github.com/kovidgoyal/calibre.git
```

Sous Windows, vous aurez besoin du nom de chemin complet, ce sera quelque chose comme `C:\Program Files\Git\git.exe`.

calibre est un très grand projet, avec une très longue histoire de contrôle de source, aussi ce qui est décrit plus haut peut prendre un moment (10 minutes à une heure dépendant de la vitesse de votre connexion internet).

Si vous voulez obtenir le code source plus rapidement, le code source de la dernière version est toujours disponible comme archive[115].

Pour mettre à jour une branche du dernier code, utilisez la commande :

```
git pull --no-edit
```

Vous pouvez également parcourir le code sur GitHub[116].

[111] https://github.com/kovidgoyal/calibre/blob/master/src/calibre/linux.py
[112] https://www.mobileread.com/forums/forumdisplay.php?f=240
[113] https://calibre-ebook.com/dist/src
[114] https://www.git-scm.com/
[115] https://calibre-ebook.com/dist/src
[116] https://github.com/kovidgoyal/calibre

14.2.1 Soumettre vos changements à inclure

Si vous planifiez de faire quelques petits changements, vous pouvez faire vos changements et crée une « directive de fusion » que vous pouvez alors attacher à un ticket dans le bug tracker[117] de calibre. Pour faire cela, faites vos changements, et exécutez :

```
git commit -am "Comment describing your changes"
git format-patch origin/master --stdout > my-changes
```

Cela créera un fichier `my-changes` dans le dossier actuel, attachez le simplement à un ticket sur le bug tracker[118] de calibre. Notez que ceci inclura *toutes* les validations effectuées. Si vous voulez seulement envoyer une partie des validations, vous devez changer `origin/master` ci-dessus. Pour envoyer seulement la dernière validation, employez :

```
git format-patch HEAD~1 --stdout > my-changes
```

Pour envoyer les dernières *n* validations, remplacer *1* avec *n*, par exemple, pour les 3 derniers validations :

```
git format-patch HEAD~3 --stdout > my-changes
```

Faites attention à ne pas inclure des fusions en employant `HEAD~n`.

Si vous planifiez de faire beaucoup de développement sur calibre, alors la meilleure méthode est de créer un compte GitHub[119]. Ci-dessous se trouve un guide basique pour mettre en place votre propre embranchement de calibre d'une manière qui vous permet de soumettre des requêtes de déchargement pour l'inclusion dans le dépôt principal calibre :

- Installer git sur votre machine comme décrit dans cet article : Installer Git[120]
- Installez les clés SSH pour l'authentification sur GitHub, comme décrit ici : Generating SSH keys[121]
- Allez sur https://github.com/kovidgoyal/calibre et cliquez sur le bouton *Fork*.
- Dans un Terminal faites :

  ```
  git clone git@github.com:<username>/calibre.git
  git remote add upstream https://github.com/kovidgoyal/calibre.git
  ```

  Remplacez <username> ci-dessus par votre nom d'utilisateur Github. Cela obtiendra que votre embranchement sera vérifié localement.

- Vous pouvez y faire des changements et des validations chaque fois que vous le désirez. Quand vous êtes prêt à fusionner votre travail, faites un :

  ```
  git push
  ```

  et rendez vous sur `https://github.com/<username>/calibre` et cliquez le bouton *Pull Request* pour générer une requête de déchargement qui peut être fusionnée.

- Vous pouvez mettre à jour votre copie locale du code depuis le dépôt principal à n'importe quel moment en faisant :

  ```
  git pull upstream
  ```

Vous devriez aussi garder un oeil sur le forum de développement[122] calibre. Avant de faire des changements majeurs, vous devriez en discuter dans le forum ou contacter directement Kovid (son adresse mail est partout dans le code source).

[117] https://bugs.launchpad.net/calibre
[118] https://bugs.launchpad.net/calibre
[119] https://github.com
[120] https://help.github.com/articles/set-up-git
[121] https://help.github.com/articles/generating-ssh-keys
[122] https://www.mobileread.com/forums/forumdisplay.php?f=240

14.3 Environnement de développement Windows

Note : Vous devez aussi récupérer le code source calibre séparément comme décrit plus haut.

Installez calibre normalement, en utilisant l'installateur de Windows[123]. Ouvrez alors une Invite de Commande et changez le dossier calibre précédemment vérifié. Par exemple :

```
cd C:\Users\kovid\work\calibre
```

calibre est le dossier qui contient les sous-dossiers src et resources.

L'étape suivante est de régler la variable d'environnement `CALIBRE_DEVELOP_FROM` au chemin absolu du dossier src. Ainsi, en suivant l'exemple ci-dessus, il devrait être `C:\Users\kovid\work\calibre\src`. Voici un court guide[124] pour paramétrer les variables d'environnement sous Windows.

Une vois que vous avez réglé la variable d'environnement, ouvrez une nouvelle invite de commande et vérifiez qu'elle est correctement paramétrée en utilisant la commande :

```
echo %CALIBRE_DEVELOP_FROM%
```

Paramétrer cette variable d'environnement signifie que calibre chargera maintenant tout son code Python à partir de l'emplacement spécifié.

Voilà. vous êtes maintenant prêt à commencer à bidouiller le code calibre. Par exemple, ouvrez le fichier `src\calibre\__init__.py` dans votre éditeur préféré et ajoutez la ligne :

```
print("Hello, world!")
```

près du dessus du ficher. Exécutez maintenant la commande **`calibredb`**. La toute première ligne de production devrait être `Hello, world!`.

Vous pouvez également mettre en place l'environnement de développement dans le Microsoft Visual Studion libre, si vous préférez, suivre les instructions ici[125].

14.4 Environnement de développement macOS

Note : Vous devez aussi récupérer le code source calibre séparément comme décrit plus haut.

Installez calibre normalement en utilisant les .dmg mis à disposition[126]. Ouvrez alors un Terminal et changez pour le dossier de code précédemment vérifié, par exemple :

```
cd /Users/kovid/work/calibre
```

calibre est le dossier qui contient les sous-dossiers src et resources. Les outils en ligne de commande calibre se trouvent à l'intérieur du paquet d'applications calibre, dans `/Applications/calibre.app/Contents/MacOS` vous devez ajouter ce dossier à votre variable d'environnement PATH, si vous voulez exécuter facilement les outils en ligne de commande.

[123] https://calibre-ebook.com/download_windows
[124] https://docs.python.org/using/windows.html#excursus-setting-environment-variables
[125] https://www.mobileread.com/forums/showthread.php?t=251201
[126] https://calibre-ebook.com/download_osx

L'étape suivante est de créer un script bash qui paramétrera les variables d'environnement *CALIBRE_DEVELOP_FROM`* au chemin absolu du dossier src lors de l'exécution de calibre en mode dépannage.

Créez un fichier texte brut :

```
#!/bin/sh
export CALIBRE_DEVELOP_FROM="/Users/kovid/work/calibre/src"
calibre-debug -g
```

Sauvegardez le fichier comme `/usr/bin/calibre-develop`, paramétrer alors ses permissions afin qu'il puisse être exécuté

```
chmod +x /usr/local/bin/calibre-develop
```

Une fois que vous avez fait cela, exécuter :

```
calibre-develop
```

Vous devriez voir quelques informations diagnostiques dans la fenêtre Terminal au démarrage de calibre, et vous devriez voir une astérisque après le numéro de version dans la fenêtre GUI, indiquant que vous exécutez depuis la source.

14.5 Environnement de développement Linux

Note : Vous devez aussi récupérer le code source calibre séparément comme décrit plus haut.

calibre est principalement développé sous Linux. Vous avez deux choix pour paramétrer l'environnement de développement. Vous pouvez installer le binaire calibre comme à la normale et l'employer comme environnement d'exécution pour faire votre développement. Cette approche est similaire à celle utilisée dans Windows et macOS. Alternativement, vous pouvez installer calibre depuis la source. Les instructions pour installer un environnement de développement depuis la source sont dans le fichier INSTALL dans l'arborescence de source. Ici nous nous adresserons en utilisant le binaire d'exécution, qui est la méthode recommandée.

Installer calibre en utilisant l'installateur binaire[127]. Ouvrez alors un terminal et changez le dossier du code calibre précédemment vérifié, par exemple :

```
cd /home/kovid/work/calibre
```

calibre est le dossier qui contient les sous-dossiers src et resources.

L'étape suivante est de créer un script bash qui paramétrera les variables d'environnement `CALIBRE_DEVELOP_FROM` au chemin absolu du dossier src. Aussi, en suivant l'exemple ci-dessus, il devrait être `/home/kovid/work/calibre/src`. Comment paramétrer les variables d'environnement dépend de votre distribution Linux et de quel shell vous utilisez.

Note : Il est recommandé d'utiliser l'installeur binaire fourni en amont. Si vous insistez pour utiliser un paquet fourni par votre distribution, utilisez plutôt les variables `CALIBRE_PYTHON_PATH` et `CALIBRE_RESOURCES_PATH`.

Une fois que vous avez paramétré la variable d'environnement, ouvrez un nouveau terminal et vérifiez qu'elle a été correctement paramétrée en utilisant la commande :

[127] https://calibre-ebook.com/download_linux

```
echo $CALIBRE_DEVELOP_FROM
```

Paramétrer cette variable d'environnement signifie que calibre chargera maintenant tout son code Python à partir de l'emplacement spécifié.

Voilà ! Vous êtes maintenant prêt à commencer à bidouiller le code calibre. Par exemple, ouvrez le fichier `src\calibre\__init__.py` dans votre éditeur préféré et ajoutez la ligne :

```
print("Hello, world!")
```

près du dessus du ficher. Exécutez maintenant la commande **calibredb**. La toute première ligne de production devrait être `Hello, world!`.

14.6 Avoir des installations séparées de calibre « normale » et « développement » sur le même ordinateur

L'arborescence de source calibre est très stable et se rompt rarement, mais si vous sentez la nécessité d'exécuter depuis la source sur une bibliothèque distincte d'essai et de d'exécuter une version libre de calibre avec votre bibliothèque quotidienne, vous pouvez réaliser ceci facilement en utilisant des fichiers .bat ou des scripts shell pour lancer calibre. L'exemple ci-dessous expose comment faire ceci sur Windows utilisant des fichiers .bat (les instructions pour d'autres plates-formes sont identiques, employez juste un script shell au lieu d'un fichier .bat)

Pour lancer la nouvelle version de calibre avec votre bibliothèque quotidienne :

calibre-normal.bat :

```
calibre.exe "--with-library=C:\path\to\everyday\library folder"
```

calibre-dev.bat :

```
set CALIBRE_DEVELOP_FROM=C:\path\to\calibre\checkout\src
calibre.exe "--with-library=C:\path\to\test\library folder"
```

14.7 Astuces de dépannage

Python est un langage opérant sur des objets de types déterminés dynamiquement avec d'excellents équipements pour l'introspection. Kovid a écrit le noyau du code calibre sans utiliser une fois un débogueur. Il y a beaucoup de stratégies pour corriger le code calibre :

14.7.1 En utilisant les déclarations d'impression

C'est la manière favorite de Kovid pour dépanner. Insérez simplement des déclarations d'impression aux points d'intérêt et exécuter votre programme dans le terminal. Par exemple : vous pouvez démarrer le GUI à partir du terminal comme :

```
calibre-debug -g
```

Similairement, vous pouvez démarrer la visionneuse de livre numérique comme

```
calibre-debug -w /path/to/file/to/be/viewed
```

The e-book editor can be started as :

```
calibre-debug --edit-book /path/to/be/edited
```

14.7.2 Utilisation d'un interpréte Python interactif

Vous pouvez insérer les deux lignes de code suivantes pour démarrer la session interactive Python à cet endroit :

```
from calibre import ipython
ipython(locals())
```

Lors de l'exécution à partir de la ligne de commande, cela démarrera l'interpréteur python interactif avec accès à toutes les variables définies localement (variables dans la portée locale). L'invite interactive a même une complétion `Tab` pour les propriétés d'objet et vous pouvez utiliser les diverses facilités Python pour l'introspection, telles que `dir()`, `type()`, `repr()`, etc.

14.7.3 Utiliser le débogueur de Python comme débogueur à distance

Vous pouvez utiliser le débogueur intégré de Python (pdb) comme débogueur à distance en ligne de commande. D'abord, démarrez le débogueur à distance au point qui vous intéresse dans le code de calibre, comme ceci :

```
from calibre.rpdb import set_trace
set_trace()
```

Exécuter alors calibre, soit comme normal, ou en utilisant une des commandes calibre-debug décrite dans la section précédente. Une fois le point ci-dessus atteint dans le code, calibre gèlera, attendant le débogueur pour se connecter.

Ouvrez maintenant un terminal ou une invite de commande et utilisez la commande suivante pour démarrer la session de débogage :

```
calibre-debug -c "from calibre.rpdb import cli; cli()"
```

Vous pouvez lire à propos de comment utiliser le débogueur Python dans Python stdlib docs for the pdb module[128].

Note : Par défaut, le débogueur à distance essaiera de se connecter sur le port 4444. Vous pouvez le changer, en modifiant le paramètre de port des fonctions set_trace() et cli() ci-dessus, comme ceci : `set_trace(port=1234)` et `cli(port=1234)`.

Note : Le débogueur de Python ne peut pas manipuler des fils multiples, aussi vous devez appeler le set_trace une fois par fil, chaque fois avec un numéro de port différent.

---

[128] https://docs.python.org/library/pdb.html#debugger-commands

14.7.4 Utiliser le débogueur dans votre IDE Python préféré

Il est possible d'utiliser le débogueur intégré dans votre IDE Python préféré, s'il supporte le débogage à distance. La première étape est d'ajouter la src vérifiée calibre au `PYTHONPATH` dans votre IDE. En d'autres mots, le dossier que vous avec paramétré plus haut comme `CALIBRE_DEVELOP_FROM`, doit aussi être dans le `PYTHONPATH` de votre IDE.

Placez alors me module IDE de débogage à distance dans le sous-dossier `src` du code source vérifié calibre. Ajoutez ni'importe quel nécessaire pour lancer le débogueur à distance sur calibre au point d'intérêt, par exemple dans la fonction principale. Exécutez alors calibre comme normalement. Votre IDE devrait être capable de se connecter au débogueur à distance s'exécutant dans calibre.

14.7.5 Exécuter des scripts quelconques dans l'environnement Python calibre

La commande **`calibre-debug`** fournit quelques commutateurs pratiques pour exécuter votre propre code, avec l'accès aux modules calibre :

```
calibre-debug -c "some Python code"
```

est bien pour examiner un petit extrait de code sur la ligne de commande. Cela fonctionne de la même manière que le commutateur -c de l'interpréteur Python :

```
calibre-debug myscript.py
```

peut être employé pour exécuter votre propre script python. Cela fonctionne de la même manière que de passer le script à l'interpréteur python, sauf que l'environnement de calibre est entièrement initialisé, ainsi vous pouvez employer tout le code de calibre dans votre script. Pour employer les arguments en ligne de commande avec votre script, utilisez la forme :

```
calibre-debug myscript.py -- --option1 arg1
```

`--` entraîne tous les arguments suivants d'être passés à votre script.

14.8 Utilisation de calibre dans vos projets

Il est possible d'utiliser d'utiliser les fonctions/code calibre dans votre projet Python. Deux manières existent pour faire cela :

14.8.1 Installation binaire de calibre

Si vous avez une installation binaire de calibre, vous pouvez utiliser l'interpréteur python empaqueté avec calibre, comme ceci :

```
calibre-debug /path/to/your/python/script.py -- arguments to your script
```

14.8.2 Installez les sources sur Linux

En complément de l'utilisation de la technique ci-dessus, si vos faites une installation des sources sur Linux, vous pouvez aussi importer directement calibre, comme suit :

```
import init_calibre
import calibre

print(calibre.__version__)
```

Il est essentiel que vous importiez le module init_calibre avant tout autre modules/package calibre comme il a installé l'interpréteur pour exécuter le code calibre.

14.9 Documentation API pour différentes parties de calibre

14.9.1 Documentation API pour l'interface base de données

Cette API est un fil sûr (Elle utilise un lecteur multiple, un schéma rédacteur unique fermé). Vous pouvez accéder à cette API comme ceci :

```
from calibre.library import db
db = db('Path to calibre library folder').new_api
```

Si vous êtes dans une extension calibre qui est partie prenante du GUI principal de calibre, vous y obtenez l'accès comme ceci à la place :

```
db = self.gui.current_db.new_api
```

class calibre.db.cache.**Cache**(*backend*, *library_database_instance=None*)

Un cache mémoire du fichier metadata.db depuis une bibliothèque calibre. Cette classe sert également comme une API de sécurité pour accéder à la base de données. Le cache mémoire est maintenu dans une forme normale pour une performance maximale.

SQLITE est simplement utilisée comme une manière robuste de lire et écrire depuis metadata.db. Toute table lecture/tri/recherche/cache est ré-implémentée. Ceci était nécessaire pour un maximum de performance et de flexibilité.

class EventType(*value*, *names=None*, ***, *module=None*, *qualname=None*, *type=None*, *start=1*, *boundary=None*)

book_created = 4

Lorsqu'un nouvel enregistrement de livre est créé dans la base de données, avec comme seul argument l'identifiant du livre

book_edited = 8

Lorsqu'un format de livre est édité, avec des arguments : (book_id, fmt)

books_removed = 5

Lorsque des livres sont retirés de la base de données avec la liste des identifiants de livres comme seul argument

format_added = 2

Lorsqu'un format est ajouté à un livre, avec les arguments : (book_id, format)

formats_removed = 3

Lorsque des formats sont retirés d'un livre, avec des arguments : (correspondance entre l'identifiant du livre et l'ensemble des formats supprimés du livre)

indexing_progress_changed = 9

Lorsque la progression de l'indexation change

items_removed = 7

Lorsque des éléments tels que les étiquettes ou les auteurs sont supprimés de certains livres. Arguments : (field_name, ids des livres concernés, ids des éléments supprimés)

items_renamed = 6

Lorsque des éléments tels que les étiquettes ou les auteurs sont renommés dans certains ou tous les livres. Arguments : (field_name, ids de livres concernés, mappage de l'ancien id d'élément vers le nouvel id d'élément)

metadata_changed = 1

Lorsque certaines métadonnées sont modifiées pour certains livres, avec des arguments : (nom du champ modifié, ensemble d'identifiants de livres concernés)

add_books (*books*, *add_duplicates=True*, *apply_import_tags=True*, *preserve_uuid=False*, *run_hooks=True*, *dbapi=None*)

Ajouter les livres spécifiés à la bibliothèque. Les livres doivent être un itérable de 2-tuples, chaque 2-tuple sous la forme `(mi, format_map)` où mi est un objet métadonnée et format_map est un dictionnaire de la forme `{fmt: path_or_stream}`, par exemple : `{'EPUB': '/path/to/file.epub'}`.

Renvoie une paire de listes : `ids, duplicates`. `ids` contient les ids de livre pour tous les livres nouvellement créés dans la base de données. `duplicates` contient `(mi, format_map)` pour tous les livres qui existent déjà dans la base de données selon l'heuristique simple de détection de doublons utilisée par *`has_book()`* (page 372).

add_custom_book_data (*name*, *val_map*, *delete_first=False*)

Ajouter les données pour le nom où val_map est une carte des book_ids vers les valeurs. Si delete_first est Vrai, toutes les données stockées précédemment seront supprimées.

add_extra_files (*book_id*, *map_of_relpath_to_stream_or_path*, *replace=True*, *auto_rename=False*)

Ajouter des fichiers de données supplémentaires

add_format (*book_id*, *fmt*, *stream_or_path*, *replace=True*, *run_hooks=True*, *dbapi=None*)

Ajouter un format au livre spécifié. Renvoie True si le format a été ajouté avec succès.

Paramètres

- **replace** – Si Vrai, remplace le format existant, autrement si le format existe déjà, renvoie Faux.
- **run_hooks** – Si Vrai, les extensions de type de fichier sont exécutées. dans le format avant et après avoir été ajoutées
- **dbapi** – Usage interne uniquement.

add_listener (*event_callback_function*, *check_already_added=False*)

Enregistrer une fonction de rappel qui sera appelée après que certaines actions aient été effectuées sur cette base de données. La fonction doit prendre trois arguments : (*`EventType`* (page 366), library_id, event_type_specific_data)

add_notes_resource (*path_or_stream_or_data*, *name : str*, *mtime : float = None*) → int

Add the specified resource so it can be referenced by notes and return its content hash

all_book_ids(*type=<class 'frozenset'>*)

Ensemble gelé de tous les ids de livre connus.

all_field_for(*field*, *book_ids*, *default_value=None*)

Identique à field_for, excepté qu'il agit sur de multiples livres en une fois

all_field_ids(*name*)

Ensemble gelé d'ids pour toutes les valeurs dans le champ``name``.

all_field_names(*field*)

Ensemble gelé de tous les champs de noms (devrait uniquement être utilisé pour les champs many-one et many-many)

author_data(*author_ids=None*)

Renvoie la donnée auteur comme un dictionnaire avec des clés : nom, genre, lien

Si aucuns auteurs avec les ids spécifiés ne sont trouvés un dictionnaire vide est renvoyé. Si author_ids est None, les données pour tous les auteurs sont renvoyées.

author_sort_from_authors(*authors*, *key_func=<function make_change_case_func.<locals>.change_case>*)

En fonction d'une liste d'auteurs, renvoie la chaîne author_sort pour les auteurs, en préférant l'auteur assorti associé à l'auteur plutôt que la chaîne calculée.

books_for_field(*name*, *item_id*)

Renvoie tous les livres associés avec l'élément identifié par `item_id`, où l'élément appartient au champ `name`.

La valeur retournée est un ensemble d'ids de livre, ou l'ensemble vide si l'élément n'existe pas.

books_in_virtual_library(*vl*, *search_restriction=None*, *virtual_fields=None*)

Affiche l'ensemble des livres dans la bibliothèque virtuelle spécifiée

compress_covers(*book_ids*, *jpeg_quality=100*, *progress_callback=None*)

Compresser les images de couverture des livres spécifiés. Une qualité de compression de 100 effectuera une compression sans perte, sinon la compression sera avec perte.

Le rappel d'avancement sera appelé avec le book_id et les anciens et nouveaux formats pour chaque livre qui a été traité. Si une erreur se produit, la nouvelle taille sera une chaîne de caractères contenant les détails de l'erreur.

copy_cover_to(*book_id*, *dest*, *use_hardlink=False*, *report_file_size=None*)

Copier la couverture dans le fichier comme un objet `dest`. Renvoie False si aucune couverture n'existe ou dest est le même fichier que la couverture actuelle. dest peut aussi être un chemin dans tel cas la couverture y est copiée si le chemin est différent du chemin actuel (prendre en compte la sensibilité à la casse).

copy_format_to(*book_id*, *fmt*, *dest*, *use_hardlink=False*, *report_file_size=None*)

Copier le format `fmt` dans le fichier comme un objet `dest`. Si le format spécifié n'existe pas, indique l'erreur `NoSuchFormat`. dest peut aussi être un chemin (vers un fichier) dans tel cas la couverture y est copiée si et seulement si le chemin est différent du chemin actuel (prendre en compte la sensibilité à la casse).

cover(*book_id*, *as_file=False*, *as_image=False*, *as_path=False*, *as_pixmap=False*)

Renvoie l'image de couverture ou Aucun. Par défaut, renvoie la couverture comme une chaîne de caractères.

ATTENTION : l'utilisation de as_path copiera la couverture vers un fichier temporaire et renverra le chemin vers ce fichier temporaire. Vous devez supprimer le fichier temporaire quand vous en avez terminé avec celui-ci.

Paramètres

- **as_file** – Si Vrai renvoie l'image en tant qu'objet fichier ouvert (une file d'attente de fichiers temporaires)
- **as_image** – Si vrai renvoie l'image comme un objet QImage
- **as_pixmap** – Si True renvoie l'image comme un objet QPixmage
- **as_path** – Si vrai renvoie l'image comme un chemin pointant vers un fichier temporaire

data_for_find_identical_books()

Renvoie les données qui peuvent être utilisées pour implémenter *find_identical_books()* (page 369) dans un processus de travail sans accès à la bd. Voir db.utils pour une implémentation.

data_for_has_book()

Renvoie les données appropriées pour être utilisées dans *has_book()* (page 372). Ceci peut être utilisé pour une implémentation de *has_book()* (page 372) dans un processus de travail sans accès à la bd.

delete_custom_book_data(*name*, *book_ids=()*)

Effacer les données pour le nom. Par défaut efface toutes les données, si vous voulez seulement effacer les données pour certaines ids de livres, passez dans une liste d'ids de livres.

delete_trash_entry(*book_id*, *category*)

Supprimer une entrée de la corbeille. Ici, la catégorie est "b" pour les livres et "f" pour les formats.

embed_metadata(*book_ids*, *only_fmts=None*, *report_error=None*, *report_progress=None*)

Mettre à jour les métadonnées pour tous les fomats du book_ids sélectionné vers les métadonnées actuelle dans la base de données.

expire_old_trash()

Expirer les entrées trop anciennes de la corbeille

export_note(*field*, *item_id*) → str

Export the note as a single HTML document with embedded images as data : URLs

fast_field_for(*field_obj*, *book_id*, *default_value=None*)

Semblable à field_for, sauf qu'il évite la consultation supplémentaire pour obtenir l'objet du champ

field_for(*name*, *book_id*, *default_value=None*)

Renvoie la valeur du champ `name` pour le livre identifié par `book_id`. Si un tel livre n'existe pas ou s'il n'y a pas de valeur spécifiée pour le champ `name` ou qu'un tel champ n'existe pas, alors `default_value` est renvoyée.

`default_value` n'est pas utilisée pour title, title_sort, authors, author_sort and series_index. C'est parce que ceux-ci ont toujours des valeurs dans la base de données. `default_value` est utilisée pour toutes les colonnes personnalisées.

La valeur renvoyée pour les champs is_multiple est toujours un tuples, même si aucune valeur n'est trouvée (en d'autres termes, default_value est ignoré). L'exception concerne les identificateurs, pour lesquels la valeur retournée est toujours un dictionnaire. Les tuples renvoyés sont toujours dans l'ordre des liens, c'est-à-dire l'ordre dans lequel ils ont été créés.

field_ids_for(*name*, *book_id*)

Renvoie les ids (en tant que tuples) pour les valeurs que le champ `name` à dans le livre identifié par `book_id`. S'il n'y a pas de valeurs, ou pas de tel livre, ou pas de tel champ, un tuple vide est renvoyé.

field_supports_notes(*field=None*) → bool

Return True iff the specified field supports notes. If field is None return frozenset of all fields that support notes.

find_identical_books(*mi*, *search_restriction=''*, *book_ids=None*)

Trouve les livres qui ont un sur-ensemble d'auteurs dans mi et ont le même titre (titre est à correspondance floue). Voir aussi *data_for_find_identical_books()* (page 369).

format(*book_id*, *fmt*, *as_file=False*, *as_path=False*, *preserve_filename=False*)

Renvoie le format de livre numérique comme une chaîne de caractères ou *None* si le format n'existe pas, ou que nous n'avons pas la permission d'écrire dans le fichier de livre numérique.

Paramètres

- **as_file** – Si True le format de livre numérique est renvoyé comme un objet fichier. Notez que l'objet fichier est un fichier temporaire de file d'attente, aussi si ce que vous voulez faire est de copier le format sur un autre fichier, utilisez *copy_format_to()* (page 368) à la place pour la performance.
- **as_path** – Copie le format de fichier vers un fichier temporaire et renvoie le chemin vers le fichier temporaire
- **preserve_filename** – Si Vrai et renvoie un chemin le nom de fichier est le même que celui utilisé dans la bibliothèque. Notez qu'utiliser ceci signifie donner des appels répétés du même fichier temporaire (qui est recréé chaque fois)

format_abspath(*book_id*, *fmt*)

Renvoie le chemin absolu du fichier livre numérique du format *format*. Vous ne devriez presque jamais utiliser ceci, car compromet la promesse threadsafe de cette API. A la place utilisez, *copy_format_to()* (page 368).

Actuellement utilisé uniquement dans la liste calibredb, la visionneuse, éditer le livre, compare_format au format original, Open With et les catalogues (via get_data_as_dict()).

A part depuis la visionneuse, open with et éditer le livre, je ne crois pas qu'aucun des autres ne fassent de fichier écriture I/O avec les résultats de cet appel.

format_hash(*book_id*, *fmt*)

Renvoie le hash du format spécifié pour le livre spécifié. Le type de hash est dépendant du système principal, mais il est habituellement SHA-256.

format_metadata(*book_id*, *fmt*, *allow_cache=True*, *update_db=False*)

Renvoie le chemin, la taille et mtime pour le format spécifié. Vous ne devriez presque jamais utiliser path à moins que vous n'en ayez absolument besoin, dès qu'il est accédé il compromet directement la promesse threadsafe de cette API. A la place utilisez la méthode *copy_format_to()* (page 368).

Paramètres

- **allow_cache** – Si `Vrai` les valeurs en cache sont utilisées, autrement un accès lent au système de fichiers est fait. Le cache des valeurs peut être périmé si l'accès est effectué au système de fichier en dehors de cette API.
- **update_db** – Si `Vrai` Le champ max_size de la base de données est mis à jour pour ce livre.

formats(*book_id*, *verify_formats=True*)

Renvoie le tuple pour tous les formats du livre spécifié. Si verify_format est Vrai, vérifie que les fichiers existent sur le disque.

get_all_items_that_have_notes(*field_name=None*) → set[int] | dict[str, set[int]]

Return all item_ids for items that have notes in the specified field or all fields if field_name is None

get_all_link_maps_for_book(*book_id*)

Renvoie tous les liens pour tous les champs référencés par book_id. Si book_id n'existe pas, la méthode renvoie {}.

Exemple : Supposons que l'auteur A ait un lien X, que l'auteur B ait un lien Y, que le tag S ait un lien F, et la balise T a un lien G. Si le livre 1 a un auteur A et une balise T, cette méthode renvoie {"authors" :{"A" :"X"}, "tags" :{"T", "G"}}. Si l'auteur du livre 2 n'est ni A ni B et qu'il n'a pas d'étiquette, cette méthode renvoie {}.

Paramètres
book_id – l'identifiant du livre en question.

Renvoie
{field : {field_value, link_value}, … pour tous les champs qui ont une valeur de lien non vide pour ce livre

get_categories(*sort='name'*, *book_ids=None*, *already_fixed=None*, *first_letter_sort=False*)

Utilisé en interne pour implémenter le Navigateur d'Etiquettes

get_custom_book_data(*name*, *book_ids=()*, *default=None*)

Donne les données pour le nom. Par défaut renvoie les données pour tous les book_ids, passez dans une liste d'ids de livre si vous voulez seulement quelques données. Renvoie un mappage du book_id à la valeur. Si une valeur particulière ne peut être décodée, utilise l'option par défaut pour celle-ci.

get_id_map(*field*)

Renvoie un mappage des nombres id aux valeurs pour le champ spécifié. Le champ doit être un champ many-one ou many-many, autrement une ValueError est évoquée.

get_ids_for_custom_book_data(*name*)

Renvoie un ensemble d'ids de livre pour lesquels le nom a des données.

get_item_id(*field*, *item_name*)

Renvoie l'id élément pour item_name(insensible à la casse) ou None si non trouvé

get_item_ids(*field*, *item_names*)

Renvoie l'id élément pour item_name(sensible à la casse)

get_item_name(*field*, *item_id*)

Renvoie le nom d'élément pour l'élément spécifié par item_id dans le champ spécifié. Voir aussi *get_id_map()* (page 371).

get_item_name_map(*field*, *normalize_func=None*)

Return mapping of item values to ids

get_link_map(*for_field*)

Renvoie un dictionnaire de liens pour le champ fourni.

Paramètres
for_field – le nom du champ pour lequel la carte de liens est souhaitée

Renvoie
{field_value :link_value, …} pour les liens non vides

get_metadata(*book_id*, *get_cover=False*, *get_user_categories=True*, *cover_as_data=False*)

Renvoie les métadonnées pour le livre identifié par book_id comme un un objet *calibre.ebooks.metadata.book.base.Metadata* (page 221) . Notez que la liste des formats n'est pas vérifiée. Si get_cover est True, la couverture est retournée, soit un chemin au fichier temporaire comme mi.cover ou si cover_as_date est True comme mi.cover_data.

get_next_series_num_for(*series*, *field='series'*, *current_indices=False*)

Renvoie les index de séries suivants pour les séries spécifiées, prendre en compte les préférences variées qui contrôlent la génération du numéro suivant de séries.

Paramètres

- **field** – Le champ series-like (par défaut à la colonne séries intégrées)
- **current_indices** – Si Vrai, renvoie à la place un mappage de book_id à l'actuelle valeur series_index.

get_notes_resource(*resource_hash*) → dict | None

Return a dict containing the resource data and name or None if no resource with the specified hash is found

get_proxy_metadata(*book_id*)

Semblable à *get_metadata()* (page 371) excepté qu'il renvoie un objet ProxyMedadata qui lit uniquement les valeurs depuis la base de données à la demande. Ceci est beaucoup plus rapide que get_metadata quand seulement un petit nombre de champs ont besoin d'être accédés depuis l'objet métadonnées renvoyé.

get_usage_count_by_id(*field*)

Renvoie un mappage de l'id au compte d'utilisation pour toutes les valeurs du champ spécifié, qui doit être un champ many-one ou many-many.

has_book(*mi*)

Renvoie True si et seulement si la base de donnée contient une entrée avec le même titre que l'objet Metadata transmis. La comparaison est sensible à la casse. Voir aussi *data_for_has_book()* (page 369).

has_format(*book_id*, *fmt*)

Renvoie Vrai si et seulement si le format existe sur le disque

has_id(*book_id*)

Renvoie Vrai si et seulement si le book_id spécifié existe dans la base de données.

import_note(*field*, *item_id*, *path_to_html_file*, *path_is_data=False*)

Import a previously exported note or an arbitrary HTML file as the note for the specified item

init()

Initialiser ce cache avec des données depuis le système principal.

items_with_notes_in_book(*book_id : int*) → dict[str, dict[int, str]]

Return a dict of field to items that have associated notes for that field for the specified book

link_for(*field*, *item_id*)

Return the link, if any, for the specified item or None if no link is found

list_extra_files(*book_id*, *use_cache=False*, *pattern=''*) → Tuple[ExtraFile, ...]

Obtenir des informations sur les fichiers supplémentaires dans le répertoire du livre

Paramètres

- **book_id** – l'identifiant de livre dans la base de données pour ce livre
- **pattern** – le modèle de noms de fichiers à rechercher. Le modèle vide correspond à tous les fichiers supplémentaires. Les modèles doivent utiliser / comme séparateur. Utilisez la constante DATA_FILE_PATTERN pour faire correspondre les fichiers à l'intérieur du répertoire de données.

Renvoie

Un n-uplet de tous les fichiers supplémentaires correspondant au modèle spécifié. Chaque élément du tuple est ExtraFile(relpath, file_path, stat_result). Où relpath est le chemin relatif du fichier vers le répertoire du livre en utilisant / comme séparateur. stat_result est le résultat de l'appel à os.stat() sur le fichier.

merge_extra_files (*dest_id*, *src_ids*, *replace=False*)

Fusionnez les fichiers supplémentaires de src_ids dans dest_id. Les fichiers en conflit sont renommés automatiquement sauf si replace=True auquel cas ils sont remplacés.

move_book_from_trash (*book_id*)

Annuler la suppression d'un livre dans le répertoire de la corbeille

move_format_from_trash (*book_id*, *fmt*)

Annuler la suppression d'un format dans le répertoire de la corbeille

multisort (*fields*, *ids_to_sort=None*, *virtual_fields=None*)

Renvoie une liste des ids de livre triés. Si ids_to_sort est Aucun, tout les ids de livre sont retournés.

les champs doivent être une liste de 2-tuples de la forme (field_name, ascending = True ou False). Le champ le plus important est le 2-tuple.

notes_data_for (*field*, *item_id*) → str

Return all notes data as a dict or None if note does not exist

notes_for (*field*, *item_id*) → str

Return the notes document or an empty string if not found

notes_resources_used_by (*field*, *item_id*)

Return the set of resource hashes of all resources used by the note for the specified item

pref (*name*, *default=None*, *namespace=None*)

Renvoie la valeur pour la préférence spécifiée ou la valeur spécifiée comme étant `par défaut` si la préférence n'est pas paramétrée.

read_backup (*book_id*)

Renvoie la sauvegarde des métadonnées OPF comme une chaîne de caractères ou Aucun si une telle sauvegarde n'existe pas.

remove_books (*book_ids*, *permanent=False*)

Supprimer de la base de données les livres spécifiés par les book_ids et effacer leurs formats de fichier . Si `permanent` est à False, alors les formats de fichiers sont placés dans la corbeille de la bibliothèque.

remove_formats (*formats_map*, *db_only=False*)

Supprimer les formats spécifiés des livres spécifiés.

Paramètres

- **formats_map** – Un mappage de book_id à une liste de formats à supprimer du livre.
- **db_only** – Si Vrai, retirer uniquement l'enregistrement pour le format de la base de données, n'efface pas le fichier de format actuel du système de fichiers.

Renvoie

Une correspondance entre l'identifiant du livre et l'ensemble des formats effectivement supprimés du système de fichiers pour ce livre.

remove_items (*field*, *item_ids*, *restrict_to_book_ids=None*)

Supprimer tous les éléments dans le champ spécifié avec les ids spécifiés. Renvoie l'ensemble des ids de livres affectés. `restrict_to_book_ids` est un ensemble optionnel d'ids de livres. Si spécifié les éléments seront uniquement supprimés de ces livres.

rename_extra_files (*book_id*, *map_of_relpath_to_new_relpath*, *replace=False*)

Renommer des fichiers de données supplémentaires

rename_items (*field*, *item_id_to_new_name_map*, *change_index=True*, *restrict_to_book_ids=None*)

Renommer les éléments d'un champ many-one ou many-many comme étiquettes ou séries.

Paramètres

- **change_index** – Lors du renommage d'un champ series-like change aussi les valeurs series_index.
- **restrict_to_book_ids** – Un ensemble optionnel d'ids de livres pour lesquels le renommage est effectué, par défaut à tous les livres.

restore_book (*book_id*, *mi*, *last_modified*, *path*, *formats*, *annotations=()*)

Restaure l'entre du livre dans la base de données pour un livre qui existe déjà dans le système de fichiers.

restore_original_format (*book_id*, *original_fmt*)

Restaure le format spécifié à partir du ORIGINAL_FORMAT précédemment sauvé, s'il y en a. Renvoie Vrai en cas de réussite. Le ORIGINAL_FORMAT est supprimé après une restauration réussie.

property safe_read_lock

Un verrou en lecture sûr qui ne fait rien si le thread à déjà un verrou en écriture, autrement il acquiert un verrou en lecture.Ceci est nécessaire pour prévenir les DowngradeLockErrors, qui peuvent survenir quand on met à jour le cache de recherche en la présence de colonnes composées. Mettre à jour le cache de recherche préserve un verrou exclusif, mais la recherche d'une colonne composée implique de lire des valeurs de champ par l'intermédiaire de ProxyMetadata qui essaye d'obtenir un verrou partagé. Il peut y avoir d'autres scénarios qui déclenchent ceci également.

Cette propriété renvoie un nouvel objet verrou à chaque accès. Cet objet verrou n'est pas récursif (pour la performance) et doit être uniquement utilisé dans une déclaration with comme `with cache.safe_read_lock:` autrement de mauvaises choses peuvent survenir.

save_original_format (*book_id*, *fmt*)

Sauver une copie du format spécifié comme ORIGINAL_FORMAT, écrase tout ORIGINAL_FORMAT existant.

search (*query*, *restriction=''*, *virtual_fields=None*, *book_ids=None*)

Rechercher dans la base de données pour la requête spécifiée, en renvoyant un ensemble d'ids de livre correspondants.

Paramètres

- **restriction** – Une restriction qui est ANDed à la requête spécifique. Notez que ces restrictions sont en cache, dès lors une recherche pour a AND b plus lente que celle avec une restriction b.
- **virtual_fields** – Utilisé en interne (champs virtuels tels que on_device pour y faire une recherche)
- **book_ids** – Si n'est pas Aucun, un ensemble d'ids de livre pour les livres sera recherché à la place de rechercher tous les livres.

search_notes(*fts_engine_query=''*, *use_stemming=True*, *highlight_start=None*, *highlight_end=None*, *snippet_size=None*, *restrict_to_fields=()*, *return_text=True*, *result_type=<class 'tuple'>*, *process_each_result=None*, *limit=None*)

Search the text of notes using an FTS index. If the query is empty return all notes.

set_conversion_options(*options*, *fmt='PIPE'*)

les options doivent être une carte de la forme {book_id :conversion_options}

set_cover(*book_id_data_map*)

Paramétrer la couverture pour ce livre. Les données peuvent être soit une QImage, une QPixmap, un objet fichier ou une chaîne de caractères. Elles peuvent être aussi Aucun, dans quel cas toute couverture existante sera supprimée.

set_field(*name*, *book_id_to_val_map*, *allow_case_change=True*, *do_path_update=True*)

Paramétrer les valeurs du champ spécifié par `name`. Renvoie l'ensemble de tous les ids de livre affectés par le changement.

Paramètres

- **book_id_to_val_map** – Mappage des book_ids aux valeurs qui devraient être appliquées.
- **allow_case_change** – Si Vrai, la casse des champs many-one ou many-many sera changée. Par exemple, si un livre à l'étiquette `tag1` et que vous paramétrez l'étiquette pour un autre livre à `Tag1` alors les deux livres ont l'étiquette `Tag1` si allow_case_change est Vrai, autrement ils ont tous les deux l'étiquette `tag1`.
- **do_path_update** – Utilisé en interne, vous ne devriez jamais le changer.

set_link_map(*field*, *value_to_link_map*, *only_set_if_no_existing_link=False*)

Définit les liens pour les valeurs des éléments dans le champ. Remarque : cette méthode ne modifie pas les valeurs qui ne figurent pas dans le value_to_link_map

Paramètres

- **field** – le nom recherché
- **value_to_link_map** – dict(field_value :link, …). Notez qu'il s'agit de valeurs et non d'identifiants de champ.

Renvoie
livres modifiés en définissant le lien

set_metadata(*book_id*, *mi*, *ignore_errors=False*, *force_changes=False*, *set_title=True*, *set_authors=True*, *allow_case_change=False*)

Paramétrer les métadonnées pour le livre *id* depuis l'objet *Metadata mi*.

En réglant force_changes=True forcera set_metadata à mettre à jour les champs même si mi contient des valeurs vides. Dans ce cas, "Aucun" est distingué de "vide". Si mi.XXX est Aucun, le XXX n'est pas remplacé, autrement il l'est. Les étiquettes, identifiants, et attributs de couverture sont des cas spéciaux. Étiquettes et identifiants ne peuvent être paramétrés à Aucun aussi seront-ils toujours remplacés si force_change est Vrai. Vous devez vous assurer que mi contient les valeurs que vous voulez que le livre ait.. Les couvertures sont toujours changées si une nouvelle couverture est procurée, mais ne sont jamais supprimées. Aussi notez que force_changes n'a pas d'effet sur les réglages titre ou auteurs.

set_notes_for(*field*, *item_id*, *doc : str*, *searchable_text : str = ''*, *resource_hashes=()*, *remove_unused_resources=False*) → int

Set the notes document. If the searchable text is different from the document, specify it as searchable_text. If the document references resources their hashes must be present in resource_hashes. Set

remove_unused_resources to True to cleanup unused resources, note that updating a note automatically cleans up resources pertaining to that note anyway.

set_pref (*name*, *val*, *namespace=None*)

Paramétrer la préférence spécifiée à la valeur spécifiée. Voir aussi *pref()* (page 373).

split_if_is_multiple_composite (*f*, *val*)

Si f est une clé de recherche de colonne composite et que la colonne est is_multiple, diviser v en valeurs uniques et non vides. La comparaison est sensible à la casse. L'ordre n'est pas préservé. Retourne une list() pour des raisons de compatibilité avec les getters des champs de proxy metadata , par exemple les tags.

tags_older_than (*tag*, *delta=None*, *must_have_tag=None*, *must_have_authors=None*)

Renvoie les ids de tous les livres ayant l'étiquette `tag` qui sont plus anciennes que le moment spécifié, la comparaison d'étiquette est insensible à la casse.

Paramètres

- **delta** – Un objet timedelta ou Aucun. Si Aucun, alors tous les ids avec l'étiquette sont retournées.
- **must_have_tag** – Si pas Aucun la liste des correspondances sera restreinte aux livres qui ont cette étiquette
- **must_have_authors** – Une liste d'auteurs. Si pas Aucun la liste des correspondances sera restreinte aux livres qui ont ces auteurs (insensible à la casse).

unretire_note_for (*field*, *item_id*) → int

Unretire a previously retired note for the specified item. Notes are retired when an item is removed from the database

user_categories_for_books (*book_ids*, *proxy_metadata_map=None*)

Retourne les catégories utilisateur pour les livres spécifiés. proxy_metadata_map est optionnel et est utile pour une amélioration de la performance, dans les contextes où un objet ProxyMetadata pour les livres existe déjà. Il devrait être une cartographie des book_ids à leurs objets correspondants ProxyMetadata.

14.9.2 Documentation API pour l'outil d'édition de livre numérique

Les outils d'édition de livre numérique consistent en un objet *calibre.ebooks.oeb.polish.container.Container* (page 377) qui représente un livre comme une collection d'HTML + des fichiers ressources, et différents outils qui peuvent être utilisés pour effectuer des opérations sur le conteneur. Tous les outils sont sous forme de fonctions de niveau de module dans les divers modules de `calibre.ebooks.oeb.polish.*` Vous obtenez un objet de conteneur pour un livre à un chemin comme ceci :

```
from calibre.ebooks.oeb.polish.container import get_container
container = get_container('Path to book file', tweak_mode=True)
```

Si vous écrivez une extension pour l'" Éditeur de livre numérique, vous obtenez le conteneur actuel pour le livre étant édité comme ceci :

```
from calibre.gui2.tweak_book import current_container
container = current_container()
if container is None:
    report_error # No book has been opened yet
```

L'objet Conteneur

class calibre.ebooks.oeb.polish.container.**Container**(*rootpath*, *opfpath*, *log*, *clone_data=None*)

Un conteneur représente un livre numérique ouvert comme un dossier remplit de fichiers et un fichier OPF. Il y a deux concepts importants :

- Le répertoire racine. Ceci est la base du livre numérique. Tous les fichiers des livres numériques sont dans ce dossier ou dans ses sous-dossiers.
- Noms : Ce sont des chemins vers les fichiers de livres relatifs au dossier racine. Ils contiennent toujours des séparateurs POSIX et sont non cotés. Ils peuvent être considérés en tant qu'identificateurs conformes pour les fichiers dans le livre. La plupart des méthodes sur l'objet conteneur fonctionnent avec des noms. Les noms sont toujours sous la forme normale unicode de NFC.
- Clones : l'objet conteneur soutient le clonage efficace sur disque, qui est employé pour mettre en application des points de contrôle dans l'éditeur de livre numérique. Afin de faire ce travail, vous devriez ne jamais accéder à des dossiers directement sur le système de fichiers. Au lieu de cela, utilisez : meth :*raw_data* ou `open()` (page 379) à la lecture/écriture aux fichiers composants le livre.

En convertissant entre les hrefs et les noms, employez les méthodes fournies par cette classe, elles supposent que tous les hrefs sont cités.

abspath_to_name(*fullpath*, *root=None*)

Convertir un fichier absolu vers un nom conforme relatif à `root`

Paramètres

root – Le dossier de base. Par défaut la racine de cet objet conteneur est utilisée.

add_file(*name*, *data*, *media_type=None*, *spine_index=None*, *modify_name_if_needed=False*, *process_manifest_item=None*)

Ajouter un fichier à ce conteneur. Les entrées pour le fichier sont automatiquement créées dans la structure organisationnelle et la structure OPF (si le fichier est un document texte)

add_name_to_manifest(*name*, *process_manifest_item=None*)

Ajouter une entrée au manifeste pour le fichier avec le nom spécifié. Renvoie l'id du manifeste.

add_properties(*name*, **properties*)

Ajouter les propriétés spécifiées à l'élément du manifeste identifié par le nom.

apply_unique_properties(*name*, **properties*)

Assure que les propriétés spécifiées soient paramétrées uniquement sur l'élément du manifeste spécifié par un nom. Vous pouvez utiliser None comme nom pour supprimer la propriété de tous les éléments.

book_type = 'oeb'

Le type de livre (epub pour les fichiers EPUB et azw3 pour les fichiers AZW3)

commit(*outpath=None*, *keep_parsed=False*)

Engager tous les fichiers analysés encrassés vers le système de fichiers et écrire le fichier de livre numérique à outpath.

Paramètres

- **output** – Le chemin où écrire le fichier du livre numérique sauvegardé. Si aucun, le chemin du livre original est utilisé.
- **keep_parsed** – Si True les représentations analysées des éléments engagés sont gardées dans le cache.

commit_item (*name*, *keep_parsed=False*)

Soumet un objet analysé au disque (il est numéroté et écrit par rapport fichier sous-jacent). Si `keep_parsed` est True la représentation analysée est maintenue dans le cache. Voir également : :meth :`parsed `

dirty (*name*)

Marque l'objet correspondant au nom comme impropre. Voir aussi : *parsed()* (page 379).

exists (*name*)

Vrai si et seulement si un fichier/dossier correspondant au nom de référence existe. Notez que cette fonction souffre des limitations du système de fichiers de l'OS sous-jacent, en particulier l"/la (in)sensibilité à la casse. Aussi sur un système de fichiers non sensible à la casse renvoie Vrai même dans le cas où le nom est différent de la casse du système de fichiers sous-jacent. Voir aussi : *has_name()* (page 378)

filesize (*name*)

Renvoie la taille en bytes du fichier représenté par le nom de référence spécifié. Traite automatiquement les objets impropres analysés. Voir aussi : *parsed()* (page 379)

generate_item (*name*, *id_prefix=None*, *media_type=None*, *unique_href=True*)

Ajoute un article au manifeste avec un href dérivé du nom attribué. Assure automatiquement le caractère unique de l'href et de l'id. Revoie l'article généré.

get_file_path_for_processing (*name*, *allow_modification=True*)

Similaire à open() excepté qu'il renvoie un chemin de fichier, au lieu d'un objet fichier ouvert.

property guide_type_map

Mappage du type de guide au fichier de référence

has_name (*name*)

Renvoie Vrai si un fichier avec le même nom de référence que spécifié existe. Contrairement à *exists()* (page 378) cette méthode est toujours sensible à la casse.

href_to_name (*href*, *base=None*)

Convertir une href (relative à la base) en un nom. La n-base peut être un nom ou Aucun, dans ce cas self.root est utilisé.

insert_into_xml (*parent*, *item*, *index=None*)

Insérer l'article dans le parent (ou apposez si l'index est Aucun), fixant l'indentation. Travaille seulement avec les articles à fermeture automatique.

is_dir = False

Si ce conteneur représente un livre décompressé (un répertoire)

iterlinks (*name*, *get_line_numbers=True*)

Répéter sur tous les liens dans le nom. Si get_line_numbers est Vrai elle produit un résultat dans la forme (link, line_number, offset). Où line_number est le numéro de ligne où le lien intervient et offset est le nombre de caractères depuis le début de la ligne. Notez que offset pourrait actuellement comprendre plusieurs lignes sinon zéro.

make_name_unique (*name*)

S'assurer que le *nom* n'est pas déjà présent dans ce livre. Si c'est le cas, renvoyer une version modifiée qui n'existe pas

manifest_has_name (*name*)

Renvoie True si le manifeste a une entrée correspondante au nom

property manifest_id_map

Mappage de l'id du manifeste aux noms de référence

manifest_items_of_type (*predicate*)

Les noms de tous les éléments du manifeste dont le media-type correspond à predicate. *predicate* peut être un paramètre, une liste, une chaîne ou une fonction comprenant un simple argument, qui sera appelé avec le media-type.

manifest_items_with_property (*property_name*)

Tous les les éléments du manifeste qui ont la propriété spécifiée

property manifest_type_map

Mappage du media-type du manifeste pour lister les noms de référence de ce media-type

property mi

Les métadonnées de ce livre comme un objet Metadata. Notez que cet objet est construit au vol chaque fois que cette propriété est requise, aussi utilisez-la de façon modérée.

name_to_abspath (*name*)

Convertir un nom de référence en un chemin absolu dépendant de l'OS

name_to_href (*name*, *base=None*)

Convertir un nom en une href relative à la base, qui doit être un nom ou Aucun dans quel cas self.root est utilisé comme la base

property names_that_must_not_be_changed

Ensemble de noms qui ne doivent jamais être renommés. Dépend du format de fichier du livre numérique.

property names_that_must_not_be_removed

Ensemble de noms qui ne doivent jamais être supprimés du conteneur. Dépend du format de fichier du livre numérique.

property names_that_need_not_be_manifested

Ensemble de noms qui ont la possibilité d'être absents du manifeste. Dépend du format de fichier du livre numérique

open (*name*, *mode='rb'*)

Ouvrir le fichier indiqué par le nom pour lecture/écriture directe. Notez que ceci compromettra le fichier s'il est sali et le retire du cache d'analyse. Vous devez terminer avec ce fichier avant d'encore accéder la version analysée, ou de mauvaises choses pourraient se produire.

property opf

Le fichier OPF analysé

opf_get_or_create (*name*)

Méthode commode pour soit récupérer le premier élément XML avec le nom spécifique ou le créer sous l'élément opf :package puis le renvoyer, s'il n'existe pas déjà.

property opf_version

La version du <package> de l'élément OPF

property opf_version_parsed

La version paramétrée dans l'élément <package> d'OPF à un tuple de nombres entiers

opf_xpath (*expr*)

Méthode commode pour évaluer une expression Xpath du fichier OPF, dont les préfixes d'espace de nom opf : et dc : ont été prédéfinis.

parsed (*name*)

Renvoie une représentation analysée du fichier spécifié par le nom. Pour l'HTML et le XML un arbre **|xml|** est renvoyé. Pour les CSS une feuille de style css_parser est renvoyée. Notez que ces objets analysez sont mis en cache pour la performance. Si vous faites n'importe quel changement à l'objet analysé, vous devez appeler *dirty()* (page 378) ainsi le conteneur sait qu'il doit mettre à jour le cache. Voir aussi *replace()* (page 380).

raw_data (*name*, *decode=True*, *normalize_to_nfc=True*)

Renvoyer les données brutes correspondant au fichier spécifié par le nom

Paramètres

- **decode** – Si Vrai et le fichier à un type MIME basé sur du texte, le décode et renvoie un objet unicode à la place de bytes brutes.
- **normalize_to_nfc** – Si vrai l'unicode renvoyé est normalisé dans la forme normale NFC comme il est requit pour les formats EPUB et AZW3.

relpath (*path*, *base=None*)

Convertir un chemin absolu (avec les séparateurs OS) en un chemin relatif à la pase (par défaut à self.root). Le chemin relatif n'est « pas » un nom. Utiliser *abspath_to_name()* (page 377) pour cela.

remove_from_spine (*spine_items*, *remove_if_no_longer_in_spine=True*)

Supprimer les articles spécifiés (par nom de référence) de la structure organisationnelle. Si `remove_if_no_longer_in_spine` est à Vrai, les articles sont aussi supprimés du livre. Pas uniquement de la structure organisationnelle.

remove_from_xml (*item*)

Supprime l'article du parent, fixant l'indentation (fonctionne seulement avec les articles à fermeture automatique)

remove_item (*name*, *remove_from_guide=True*)

Supprimer l'article spécifié par le nom de ce conteneur. Ceci supprime toutes les références de l'article dans le manifeste, le guide, la structure organisationelle de l'OPF tout comme tout cache interne.

rename (*current_name*, *new_name*)

Renomme un fichier de current_name à new_name. Il redirige automatiquement tous les liens dans le fichier si le dossier où ce trouve le fichier change. Notez cependant, ces liens ne sont pas mis à jour dans les autres fichiers qui peuvent référencer ce fichier. Ceci est pour la performance, une telle mise à jour devrait être faite une fois, en lot.

replace (*name*, *obj*)

Remplacer l'objet analysé correspondant au nom avec obj. qui doit être un objet similaire, par ex. un arbre **|xml|** pour du HTML/XML ou une feuille de style css_parser pour un fichier CSS.

replace_links (*name*, *replace_func*)

Remplacer tous les liens en nom en utilisant replace_func, qui doit être un appelable qui accepte une URL et retourne l'URL remplacée. Il doit aussi avoir un attribut "remplacé'qui doit être paramétré à Vrai si tout remplacement actuel est fini. Des manières commodes de créer de tels appelables sont d'utiliser les classes `LinkReplacer` et `LinkRebaser`.

serialize_item (*name*)

Convertir un objet analysé (identifié par nom de référence) en une chaîne de bytes.Voir aussi : *parsed()* (page 379).

set_spine (*spine_items*)

Régler la structure organisationnelle pour être une spine_items où spine_items est une itération de la forme (nom, linéaire). Soulèvera une erreur si un des noms n'est pas présent dans le manifeste.

property spine_items

Un itérateur rapportant le chemin pour chaque élément dans la structure organisationnelle des livres. Voir aussi : *spine_iter* (page 381) and *spine_items* (page 380).

property spine_iter

Un itérateur qui rapporte l'article, nom is_linear pour chaque article dans la structure organisationnelle des livres, l'article est l'élément |xml|, le nom est le nom de fichier de référence et is_linear est Vrai si l'article est linéaire. Voir aussi : *spine_names* (page 381) et *spine_items* (page 380).

property spine_names

Un itérateur rapportant le nom et is_linear pour chaque article dans la structure organisationnelle des livres. Voir aussi : *spine_iter* (page 381) et *spine_items* (page 380).

Gestion des fichiers composant dans un conteneur

calibre.ebooks.oeb.polish.replace.**replace_links**(*container*, *link_map*, *frag_map=<function <lambda>>*, *replace_in_opf=False*)

Remplacer les liens aux fichiers dans le conteneur. Itératera sur tous les fichiers dans le conteneur et changera les liens spécifiés par ceux-ci.

Paramètres

- **link_map** – Un mappage d'un ancien nom de référence vers un nouveau nom de référence. Par exemple : `{'images/old.png': 'images/new.png'}`
- **frag_map** – Un appelable qui prend deux arguments `(name, anchor)` et renvoie une nouvelle ancre. Ceci est utile si vous avez besoin de changer les ancres dans des fichiers HTML. Par défaut, il ne fait rien.
- **replace_in_opf** – Si Faux, les liens ne sont pas remplacés dans le ficher OPF..

calibre.ebooks.oeb.polish.replace.**rename_files**(*container*, *file_map*)

Renommer les fichier dans le conteneur, met à jour automatiquement tous les liens vers ceux-ci.

Paramètres

file_map – Un mappage d'un ancien nom de référence vers un nouveau nom de référence. Par exemple : `{'text/chapter1.html': 'chapter1.html'}`.

calibre.ebooks.oeb.polish.replace.**get_recommended_folders**(*container*, *names*)

Renvoie les répertoires qui sont recommandés pour les noms de fichier donnés. La recommandation est basée sur où la majorité des fichiers du même type sont localisés dans le conteneur. Si aucun fichiers d'un type particulier ne sont présents, le répertoire recommandé est supposé être le répertoire contenant le fichier OPF.

Impression enjolivée et fixation automatique de l'analyse des erreurs

calibre.ebooks.oeb.polish.pretty.**fix_html**(*container*, *raw*)

Répare toutes les erreurs analysées dans l'HTML représentées par une chaîne en brut. La réparation se fait en utilisant l'algorithme d'analyse HTML5.

calibre.ebooks.oeb.polish.pretty.**fix_all_html**(*container*)

Répare toutes les erreurs analysées dans tous les fichiers HTML dans le conteneur. La réparation se fait en utilisant l'algorithme d'analyse HTML5.

calibre.ebooks.oeb.polish.pretty.**pretty_html**(*container*, *name*, *raw*)

Sortie enjolivée de l'HTML présentée comme une chaîne en brut.

calibre.ebooks.oeb.polish.pretty.**pretty_css**(*container*, *name*, *raw*)

Sortie enjolivée du CSS présenté comme une chaîne en brut.

calibre.ebooks.oeb.polish.pretty.**pretty_xml**(*container*, *name*, *raw*)

Sortie enjolivée de l'XML présenté comme une chaîne en brut. Si le « nom » est le nom de l'OPF, un OPF-spécifique supplémentaire est réalisé.

calibre.ebooks.oeb.polish.pretty.**pretty_all**(*container*)

Sortie enjolivée de tous les fichiers HTML/CSS/XML du conteneur.

Gérer les jaquettes de livre

calibre.ebooks.oeb.polish.jacket.**remove_jacket**(*container*)

Supprimer une jaquette existante, s'il en existe une. Renvoie False si aucune jaquette existante n'est trouvée.

calibre.ebooks.oeb.polish.jacket.**add_or_replace_jacket**(*container*)

Soit crée une nouvelle jaquette à partir des métadonnées du livre ou remplace une jaquette existante. Renvoie Vrai si une jaquette existante était remplacée.

Scission et fusion de fichiers

calibre.ebooks.oeb.polish.split.**split**(*container*, *name*, *loc_or_xpath*, *before=True*, *totals=None*)

Diviser le fichier spécifié par nom à la position spécifiée par loc_or_xpath. La scission migre automatiquement tous les liens et références aux fichiers affectés.

Paramètres

- **loc_or_xpath** – Doit être une expression XPath telle que //h :div[@id= »split_here »]. Peut aussi être un « loc » qui est utilisé en interne pour implémenter la scission dans le panneau de prévisualisation.
- **before** – Si Vrai la scission s'applique avant l'élément identifié autrement après celui-ci.
- **totals** – Utilisé en interne

calibre.ebooks.oeb.polish.split.**multisplit**(*container*, *name*, *xpath*, *before=True*)

Scinder le fichier spécifié à de multiples endroits (toutes les balises qui correspondent à l'expression XPath spécifiée). Voir aussi : *split()* (page 382). La scission migre automatiquement tous les liens et références aux fichiers affectés.

Paramètres

before – Si Vrai la scission s'applique avant l'élément identifié autrement après celui-ci.

calibre.ebooks.oeb.polish.split.**merge**(*container*, *category*, *names*, *master*)

Fusionner les fichiers spécifiés en un seul fichier, migre automatiquement tous les liens et références aux fichiers affectés. Les fichiers doivent tous être des fichiers soit HTML soit CSS.

Paramètres

- **category** – Doivent être soit *"texte"*" pour l'HTML soit `'styles'` pour les fichiers CSS
- **names** – La liste des fichiers à fusionner.
- **master** – Lequel des fichiers fusionnés est le fichier *maître*, c'est à dire, le fichier qui restera après la fusion.

Gérer les couvertures

calibre.ebooks.oeb.polish.cover.**set_cover**(*container*, *cover_path*, *report=None*, *options=None*)

Faire correspondre la couverture du livre à l'image indiquée par cover_path.

Paramètres

- **cover_path** – Soit le chemin absolu vers un fichier image ou le nom de référence d'une image dans le livre. lors de l'utilisation d'une image dans le livre, vous devez aussi paramétrer les options. Voir ci-dessous.
- **report** – Un appelable optionnel qui prend un seul argument. Il sera appelé avec les informations à propos des tâches en cours d'exécution.
- **options** – Aucun ou un dictionnaire qui contrôle comment la couverture est paramétrée. Le dictionnaire peut avoir des entrées **keep_aspect** : Vrai ou Faux (Préserve le ratio d'aspect des couvertures dans l'EPUB) **no_svg** : Vrai ou Faux (Utiliser un habillage de couverture SVG dans la page de titre de l'EPUB) **existing** : Vrai ou Faux (`cover_path` réfère à une image existante dans le livre)

calibre.ebooks.oeb.polish.cover.**mark_as_cover**(*container*, *name*)

Marquer l'image spécifiée comme l'image de couverture.

calibre.ebooks.oeb.polish.cover.**mark_as_titlepage**(*container*, *name*, *move_to_start=True*)

Marquer le fichier HTML spécifié comme la page de titre de l'EPUB.

Paramètres

move_to_start – Si Vrai le fichier HTML est dépalcé au début de la structure organisationnelle.

Travailler avec le CSS

calibre.ebooks.oeb.polish.fonts.**change_font**(*container*, *old_name*, *new_name=None*)

Changer une famille de police de ancien_nom à nouveau_nom. Change toutes les occurrences de la famille de police dans les feuilles de style, les balises de styles et les attributs de style. Si l'ancien_nom correspond à une police intégrée, elle est retirée. Vous pouvez paramétrer new_name à Aucun pour supprimer une famille de police au lieu de la changer.

calibre.ebooks.oeb.polish.css.**remove_unused_css**(*container*, *report=None*, *remove_unused_classes=False*, *merge_rules=False*, *merge_rules_with_identical_properties=False*, *remove_unreferenced_sheets=False*)

Supprimer toutes les règles CSS inutilisées du livre. Une règle CSS inutilisée est une qui ne correspond pas à tout contenu actuel.

Paramètres

- **report** – Un appelable optionnel qui prend un seul argument. Il sera appelé avec les informations à propos des opérations en cours d'exécution.
- **remove_unused_classes** – Si Vrai, les attibuts de classe HTML qui ne correspondent à aucunes règles CSS sont aussi supprimées.
- **merge_rules** – Si Vrai, les règles avec des sélecteurs identiques sont fusionnées
- **merge_rules_with_identical_properties** – Si Vrai, les règles avec des propriétés identiques sont fusionnées

- **remove_unreferenced_sheets** – Si True, les feuilles de style qui ne sont pas référencées par un contenu sont supprimées

calibre.ebooks.oeb.polish.css.**filter_css**(*container*, *properties*, *names=()*)

Supprimer les propriétés du CSS spécifié de toutes les règles CSS dans le livre.

Paramètres

- **properties** – Ensemble de propriétés à supprimer. Par exemple : `{'font-family', 'color'}`.
- **names** – Les fichiers desquels supprimer les propriétés. Options par défaut pour tous les fichiers HTML et CSS dans le livre.

Travailler avec la Table des Matières

calibre.ebooks.oeb.polish.toc.**from_xpaths**(*container*, *xpaths*, *prefer_title=False*)

Générer une Table des Matières à partir d'une liste d'expressions XPath. Chaque expresssion dans la liste correspond à un niveau de la TdM générée. Par exemple : `['//h:h1', '//h:h2', '//h:h3']` générera une Table des Matières à trois niveaux à partir des balises `<h1>`, `<h2>` et `<h3>`.

calibre.ebooks.oeb.polish.toc.**from_links**(*container*)

Générer une Table des Matières à partir de liens dans le livre.

calibre.ebooks.oeb.polish.toc.**from_files**(*container*)

Générer la table des matières à partir de fichiers dans le livre.

calibre.ebooks.oeb.polish.toc.**create_inline_toc**(*container*, *title=None*)

Créer une Table des Matières intégrée (en HTML) à partir d'une Table des Matières NCX existante.

Paramètres

title – Le titre pour cette table des matières

Outil d'édition de livre

class calibre.gui2.tweak_book.plugin.**Tool**

Bases : `object`

La classe de base pour les outils individuels dans l'extension Editer le Livre. Les membres utiles incluent :

- `self.plugin` : Une référence à l'objet *calibre.customize.Plugin* (page 268) auquel cet outil appartient.
- self. *boss* (page 385)
- self. *gui* (page 385)

Méthodes qui doivent êtres annulées dans les sous classes :

- *create_action()* (page 385)
- *register_shortcut()* (page 385)

name = None

Paramétrer ceci à un nom unique il sera utiliser comme une clé

allowed_in_toolbar = True

Si Vrai l'utilisateur peut choisir de placer cet outil dans la barre d'outil des extensions

allowed_in_menu = True

Si Vrai l'utilisateur peut choisir de placer cet outil dans le menu des extensions

toolbar_button_popup_mode = 'delayed'

Le mode du menu contextuel (s'il y en a) du bouton de la barre d'outil. Les valeurs possibles sont "delayed", "instant", "button"

property boss

L'objet *calibre.gui2.tweak_book.boss.Boss* (page 386) . Utiliser pour contrôler l'interface utilisateur.

property gui

La fenêtre principale de l'interface utilisateur

property current_container

Renvoie l'objet actuel *calibre.ebooks.oeb.polish.container.Container* (page 377) qui représente le livre en cours d'édition.

register_shortcut(*qaction*, *unique_name*, *default_keys=()*, *short_text=None*, *description=None*, ***extra_data*)

Enregistrer un raccourci clavier qui déclenchera la `qaction` spécifiée. Ce raccourci clavier deviendra automatiquement personnalisable pat l'utilisateur via la section Raccourcis clavier des préférences de l'éditeur.

Paramètres

- **qaction** – Un objet QAction, il sera déclenché quand la combinaison de touches configurée est pressée par l'utilisateur.
- **unique_name** – Un nom unique pour ce raccourci/action. Il sera utilisé en interne, il ne doit pas être partagé par toutes autres actions dans cette extension.
- **default_keys** – Une liste des raccourcis clavier par défaut. Si pas spécifié aucuns raccourcis par défaut ne sera paramétré. Si les raccourcis spécifiés ici entre en conflit avec soit les raccourcis intégrés soit les raccourcis d'une configuration utilisateur/d'autres extensions, ils seront ignorés. Dans ce cas, les utilisateurs auront à configurer les raccourcis manuellement via Préférences. Par exemple : `default_keys=('Ctrl+J', 'F9')`.
- **short_text** – Une courte description optionnelle de cette action. Si pas spécifiée le texte de QAction sera utilisé.
- **description** – Une description optionnelle plus longue de cette action, il sera utilisé dans les entrées préférences de ce raccourci.

create_action(*for_toolbar=True*)

Créer une QAction qui sera ajoutée à soit la barre d'outils extensions soit le menu d'extensions dépendant de `for_toolbar`. Par exemple :

```
def create_action(self, for_toolbar=True):
    ac = QAction(get_icons('myicon.png'), 'Do something')
    if for_toolbar:
        # We want the toolbar button to have a popup menu
        menu = QMenu()
        ac.setMenu(menu)
        menu.addAction('Do something else')
        subaction = menu.addAction('And another')

        # Register a keyboard shortcut for this toolbar action be
        # careful to do this for only one of the toolbar action or
        # the menu action, not both.
```

(suite sur la page suivante)

(suite de la page précédente)

```
        self.register_shortcut(ac, 'some-unique-name', default_keys=('Ctrl+K',
↪))
    return ac
```

Voir aussi :

Méthode *register_shortcut()* (page 385).

Contrôler l'interface utilisateur de l'éditeur

L'interface utilisateur de l'éditeur de livre numérique est contrôlée par un unique objet global *Boss*. Il y a beaucoup de méthodes utiles qui peuvent être utilisée dans le code de l'extension pour effectuer des tâches courantes.

class calibre.gui2.tweak_book.boss.**Boss**(*parent*, *notify=None*)

add_savepoint(*msg*)

Créer un point de contrôle de restauration avec le nom spécifié en tant que `msg`

apply_container_update_to_gui(*mark_as_modified=True*)

Mettre à jour tous les composants de l'interface utilisateur pour refléter les dernières données dans le conteneur actuel du livre.

Paramètres

mark_as_modified – Si Vrai, le livre sera marqué comme modifié, ainsi l'utilisateur sera averti de le sauvé en quittant.

close_editor(*name*)

Fermer l'éditeur qui édite le fichier spécifié par le `name`

commit_all_editors_to_container()

Appliquer toutes les modifications que l'utilisateur a apportées aux fichiers ouverts dans les éditeurs dans le conteneur. Vous devriez appeler cette méthode avant d'effectuer n'importe quelles actions sur le conteneur actuel

property currently_editing

Renvoie le nom du fichier qui actuellement édité ou Aucun s'il n'y a pas de fichier en cours d'édition

edit_file(*name*, *syntax=None*, *use_template=None*)

Ouvrir le fichier spécifié par le nom dans un éditeur

Paramètres

- **syntax** – Le type de média du fichier, par exemple, ``"text/html. Si pas spécifié il est estimé à partir de l'extension de fichier.
- **use_template** – Un modèle pour initialiser l'ouverture de l'éditeur

open_book(*path=None*, *edit_file=None*, *clear_notify_data=True*, *open_folder=False*, *search_text=None*)

Ouvrez le livre numérique au `path` pour l'édition. Montrera une erreur si le livre numérique n'est pas dans un format soutenu ou que le livre actuel a des changements non sauvegardés.

Paramètres

edit_file – Le nom d'un fichier à l'intérieur du livre nouvellement ouvert pour commencer l'édition. Peut aussi être une liste de noms.

rewind_savepoint()

Annule la création précédente d'un point de contrôle de restauration, utile si vous créer un point de contrôle, termine alors l'opération sans changements

save_book()

Sauvegarder le livre. La sauvegarde s'effectue en tâche de fond

set_modified()

Marquer le livre comme ayant été modifié

show_current_diff(*allow_revert=True*, *to_container=None*)

Montrer les changements dans le livre depuis sont dernier état au point de contrôle

Paramètres

- **allow_revert** – Si Vrai la boîte de dialogue diff aura un bouton qui permet à l'utilisateur d'annuler tous les changements
- **to_container** – Un objet conteneur pour y comparer le conteneur actuel. Si Aucun, le conteneur du point de contrôle précédent est utilisé

show_editor(*name*)

Montrer l'éditeur qui édite le fichier spécifié par le `name`

sync_preview_to_editor()

Synchronise la position du panneau de configuration avec la position actuelle du curseur dans l'éditeur actuel

CHAPITRE 15

Gestion des Droits Numériques (GDN - DRM)

La Gestion des Droits Numériques (GDN - DRM) est un terme générique désignant les technologies de contrôle d'accès qui peuvent être utilisées par les fabricants de matériel, les éditeurs, les détenteurs de droits d'auteur et les particuliers pour tenter d'imposer des limites à l'utilisation des contenus et des appareils numériques. Il est aussi, parfois, décrit de façon désobligeante comme la gestion des restrictions numériques. Le terme est utilisé pour décrire toute technologie qui empêche les utilisations (légitimes ou non) de contenu numérique qui n'étaient pas souhaitées ou prévues par le fournisseur de contenu. Le terme ne fait généralement pas référence à d'autres formes de protection contre la copie qui peuvent être contournées sans modifier le fichier ou le dispositif, telles que les numéros de série ou les fichiers clés. Il peut également faire référence à des restrictions associées à des cas spécifiques d'œuvres ou de dispositifs numériques. Les technologies GDN tentent de contrôler l'utilisation des supports numériques en empêchant l'accès, la copie ou la conversion vers d'autres formats par les utilisateurs finaux. Voir Wikipedia[129].

15.1 Qu'est-ce que la GDN implique pour moi personnellement ?

Lorsque vous achetez un livre numérique avec GDN, vous ne le possédez pas vraiment mais vous avez acheté l'autorisation de l'utiliser de la manière qui vous a été dictée par le vendeur. La GDN limite ce que vous pouvez faire avec les livres numériques que vous avez « achetés ». Souvent, les personnes qui achètent des livres avec GDN ne sont pas conscientes de l'étendue de ces restrictions. Ces restrictions vous empêchent de reformater le livre numérique à votre convenance, y compris d'apporter des modifications stylistiques comme l'ajustement de la taille des caractères, bien qu'il existe des logiciels qui vous permettent de faire de telles choses pour les livres non GDN. Les gens sont souvent surpris qu'un livre électronique qu'ils ont acheté dans un format particulier ne puisse pas être converti dans un autre format si le livre numérique est doté de GDN. Ainsi, si vous avez un Amazon Kindle et que vous achetez un livre vendu par Barnes and Nobles, vous devez savoir que si ce livre électronique est doté de GDN, vous ne pourrez pas le lire sur votre Kindle. Notez que je parle d'un livre que vous achetez, pas d'un livre volé ou piraté mais d'un livre ACHETÉ.

[129] https://en.wikipedia.org/wiki/Digital_rights_management

15.2 Que fait la GDN pour les auteurs ?

Les éditeurs de livres numériques avec GDN soutiennent que la GDN est dans l'intérêt des auteurs, pour protéger leur intégrité artistique et prévenir le piratage. Mais la GDN n'empêche PAS le piratage. Les personnes qui veulent pirater des contenus ou utiliser des contenus piratés le font quand même et réussissent. Les trois principaux systèmes de GDN pour les livres numériques sont aujourd'hui gérés par Amazon, Adobe et Barnes and Noble, et ils ont tous les trois été piratés. Les GDN ne font qu'incommoder les utilisateurs légitimes. On peut affirmer qu'elle nuit en fait aux auteurs, car les personnes qui auraient acheté le livre choisissent de trouver une version piratée car elles ne sont pas prêtes à supporter les GDN. Ceux qui pirateraient en l'absence de GDN le font aussi en sa présence. Pour réitérer, le point essentiel est que les GDN *n'empêchent pas le piratage*. Les GDN sont donc non seulement inutiles et nuisibles pour les acheteurs de livres numériques, mais aussi un gaspillage d'argent.

15.3 GDN et liberté

Bien que le contenu numérique puisse être utilisé pour rendre l'information ainsi que les œuvres créatives facilement accessibles à tous et donner du pouvoir à l'humanité, ce n'est pas dans l'intérêt de certains éditeurs qui veulent éloigner les gens de cette possibilité de liberté simplement pour maintenir leur pertinence dans un monde qui se développe si vite qu'ils ne peuvent pas suivre.

15.4 Pourquoi calibre ne soutient-il pas les GDN ?

calibre est un logiciel à libre accès, alors que la GDN est fermée par nature. Si calibre devait permettre d'ouvrir ou de visualiser des fichiers GDN, il pourrait être trivialement modifié pour servir d'outil de suppression des GDN, ce qui est illégal en vertu des lois actuelles. Les logiciels libres et les GDN sont un conflit de principes. Alors que la GDN vise à contrôler l'utilisateur, le logiciel libre vise à donner du pouvoir à l'utilisateur. Les deux ne peuvent tout simplement pas coexister.

15.5 Quel est le point de vue de calibre sur les fournisseurs de contenu ?

Nous croyons fermement que les auteurs et autres fournisseurs de contenu doivent être rémunérés pour leurs efforts, mais la GDN n'est pas la solution. Nous développons cette base de données de livres numériques sans DRM provenant de diverses sources pour vous aider à trouver des alternatives sans DRM et pour aider les auteurs et les éditeurs indépendants de livres numériques sans GDN à faire connaître leur contenu. Nous espérons que cela vous sera utile et nous vous demandons de ne pas pirater le contenu mis à votre disposition ici.

15.6 Comment puis-je contribuer à la lutte contre les GDN ?

En tant que personne qui lit et achète des livres numériques, vous pouvez contribuer à la lutte contre les GDN. N'achetez pas de livres numériques avec des GDN. Il existe des éditeurs qui publient des livres électroniques sans GDN. Faites un effort pour savoir s'ils proposent le livre numérique que vous recherchez. Si vous aimez les livres de certains auteurs indépendants qui vendent des livres numériques sans GDN et que vous pouvez vous le permettre, faites-leur des dons. C'est de l'argent bien dépensé car leurs livres numériques ont tendance à être moins chers (il peut y avoir des exceptions) que ceux que vous achèteriez auprès d'éditeurs de livres avec GDN et ils fonctionneraient probablement sur tous les appareils que vous posséderiez à l'avenir, ce qui vous éviterait d'acheter à nouveau le livre numérique. Ne découragez pas les éditeurs et les auteurs de livres numériques sans GDN en piratant leur contenu. Les fournisseurs de contenu méritent

une compensation pour leurs efforts. Ne les punissez pas pour avoir essayé d'améliorer votre expérience de lecture en mettant à disposition des livres numériques sans GDN. À long terme, cela vous sera préjudiciable. Si vous avez acheté des livres à des vendeurs qui proposent à la fois des livres avec et sans GDN, ne sachant pas s'ils proposent des GDN ou ne se souciant pas de laisser un commentaire ou une critique sur le site web pour informer les futurs acheteurs de son statut de GDN. De nombreux vendeurs ne pensent pas qu'il soit important d'indiquer clairement à leurs acheteurs si un livre numérique comporte ou non des GDN. Vous trouverez ici <https://www.defectivebydesign.org/guide/ebooks>` un guide pour vivre sans GDN.

CHAPITRE 16

Glossaire

RSS

** RSS ** * (Really Simple Syndication - Syndication Vraiment Simple) * est un format d'alimentation Web qui est employé pour publier le contenu fréquemment mis à jour, comme des articles d'actualités, des courriers de blog, etc. C'est un format qui est particulièrement adapté à la lecture par des ordinateurs, et est donc le moyen privilégié d'entrer du contenu du web dans un livre numérique. Il y a beaucoup d'autres formats d'alimentation en service sur l'Internet, et calibre comprend la plupart d'entre eux. En particulier, il a une bonne prise en charge du format *ATOM*, qui est utilisé généralement pour des blogs.

Recette

Une recette est un ensemble d'instructions qui apprennent à calibre comment convertir une source d'actualités en ligne, comme un magazine ou un blog, en un livre numérique. Une recette est essentiellement du code Python[130]. En soi, il est capable de convertir n'importe quelle source complexe d'actualités en livres numériques. Au niveau le plus simple, c'est juste un ensemble de variables, tels que des URLs, qui donnent à calibre assez d'informations pour aller sur Internet et télécharger les actualités.

HTML

HTML *(Hyper Text Mark-Up Language - Langage de Balisage HyperTexte)*, un sous-ensemble de Standard Generalized Mark-Up Language (SGML) pour l'édition électronique, c'est le standard spécifique utilisé pour le World Wide Web.

CSS

CSS *(Cascading Style Sheets - Feuilles de Style en Cascade)* est un langage utilisé pour décrire comment un document *HTML* devrait être restitué (esthétique visuelle).

API

API *(Application Programming Interface - Interface de Programmation d'Application)* est une interface de code source qu'une bibliothèque fournit aux demandes de soutien pour que les services y soient réalisés par des programmes informatiques.

LRF

LRF Le format de livre numérique qui est lu par les lecteurs de livre numérique SONY.

URL

URL *(Uniform Resource Locator- Localisateur de Ressources Uniformes)* par exemple : `http://example.com`

[130] https://www.python.org

regexp

Les expressions régulières fournissent des moyens concis et flexibles pour identifier des chaînes de texte d'intérêt comme des caractères particuliers, des mots ou des modèles de caractères. Voir *le tutoriel* (page 225) pour la syntaxe des expressions régulières utilisées dans Python.

Index des modules Python

C

calibre.customize, 268
calibre.customize.conversion, 278
calibre.db.cache, 366
calibre.devices.interface, 281
calibre.ebooks.metadata.book.base, 221
calibre.ebooks.metadata.sources.base, 274
calibre.ebooks.oeb.polish.container, 376
calibre.ebooks.oeb.polish.cover, 383
calibre.ebooks.oeb.polish.css, 383
calibre.ebooks.oeb.polish.jacket, 382
calibre.ebooks.oeb.polish.pretty, 381
calibre.ebooks.oeb.polish.replace, 381
calibre.ebooks.oeb.polish.split, 382
calibre.ebooks.oeb.polish.toc, 384
calibre.gui2.tweak_book.boss, 386
calibre.utils.formatter_functions, 200
calibre.web.feeds.news, 42

Index

Symboles

-1
    option de ligne de commande calibredb-add, 328
-H
    option de ligne de commande ebook-polish, 351
-I
    option de ligne de commande calibredb-add, 328
    option de ligne de commande fetch-ebook-metadata, 353
-S
    option de ligne de commande calibredb-add, 328
-T
    option de ligne de commande calibredb-add, 328
-U
    option de ligne de commande ebook-polish, 352
--access-log
    option de ligne de commande calibre-server, 319
--add
    option de ligne de commande calibredb-add, 328
--add-plugin
    option de ligne de commande calibre-customize, 316
--add-simple-plugin
    option de ligne de commande calibre-debug, 317
--add-soft-hyphens
    option de ligne de commande ebook-polish, 351
--ajax-timeout
    option de ligne de commande calibre-server, 319
--all
    option de ligne de commande calibredb-backup_metadata, 336
    option de ligne de commande calibredb-export, 330
--allowed-plugin
    option de ligne de commande fetch-ebook-metadata, 353
--append
    option de ligne de commande calibredb-set_custom, 335
--as-extra-data-file
    option de ligne de commande calibredb-add_format, 329
--as-opf
    option de ligne de commande calibredb-show_metadata, 329
--ascending
    option de ligne de commande calibredb-list, 326
--asciiize
    option de ligne de commande ebook-convert, 341
--attachment
    option de ligne de commande calibre-smtp, 324
--auth-mode
    option de ligne de commande calibre-server, 319
--author-sort
    option de ligne de commande ebook-convert, 347
    option de ligne de commande ebook-meta, 349
--authors
    option de ligne de commande calibredb-add, 327
    option de ligne de commande ebook-convert, 347
    option de ligne de commande ebook-

meta, 349
option de ligne de commande fetch-ebook-metadata, 353
--auto-reload
option de ligne de commande calibre-server, 319
--automerge
option de ligne de commande calibredb-add, 327
--ban-after
option de ligne de commande calibre-server, 319
--ban-for
option de ligne de commande calibre-server, 320
--base-dir
option de ligne de commande web2disk, 355
--base-font-size
option de ligne de commande ebook-convert, 341
--book-list-mode
option de ligne de commande calibre-server, 320
--book-producer
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 349
--build-plugin
option de ligne de commande calibre-customize, 316
--cafile
option de ligne de commande calibre-smtp, 324
--catalog-title
option de ligne de commande calibredb-catalog, 332
--categories
option de ligne de commande calibredb-list_categories, 336
--category
option de ligne de commande ebook-meta, 349
--change-justification
option de ligne de commande ebook-convert, 341
--chapter
option de ligne de commande ebook-convert, 345
--chapter-mark
option de ligne de commande ebook-convert, 345
--command
option de ligne de commande calibre-debug, 317
--comments
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 349
--compress-images
option de ligne de commande ebook-polish, 351
--compress-min-size
option de ligne de commande calibre-server, 320
--continue
option de ligne de commande ebook-viewer, 352
--cover
option de ligne de commande calibredb-add, 327
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 349
option de ligne de commande ebook-polish, 351
option de ligne de commande fetch-ebook-metadata, 353
--cross-reference-authors
option de ligne de commande calibredb-catalog, 332
--csv
option de ligne de commande calibredb-check_library, 335
option de ligne de commande calibredb-list_categories, 336
--custom-list-template
option de ligne de commande calibre-server, 320
--customize-plugin
option de ligne de commande calibre-customize, 316
--daemonize
option de ligne de commande calibre-server, 320
--date
option de ligne de commande ebook-meta, 349
--debug-device-driver
option de ligne de commande calibre-debug, 317
--debug-pipeline
option de ligne de commande calibredb-catalog, 332
option de ligne de commande ebook-

convert, 348
--default-programs
option de ligne de commande calibre-debug, 317
--delay
option de ligne de commande web2disk, 355
--detach
option de ligne de commande calibre, 316
option de ligne de commande ebook-edit, 348
option de ligne de commande ebook-viewer, 352
--details
option de ligne de commande calibredb-custom_columns, 334
--dialect
option de ligne de commande calibredb-list_categories, 336
--diff
option de ligne de commande calibre-debug, 317
--disable-allow-socket-preallocation
option de ligne de commande calibre-server, 320
--disable-auth
option de ligne de commande calibre-server, 320
--disable-dehyphenate
option de ligne de commande ebook-convert, 344
--disable-delete-blank-paragraphs
option de ligne de commande ebook-convert, 344
--disable-fallback-to-detected-interface
option de ligne de commande calibre-server, 320
--disable-fix-indents
option de ligne de commande ebook-convert, 344
--disable-font-rescaling
option de ligne de commande ebook-convert, 342
--disable-format-scene-breaks
option de ligne de commande ebook-convert, 344
--disable-hyphenation
option de ligne de commande lrfviewer, 354
--disable-italicize-common-cases
option de ligne de commande ebook-convert, 344
--disable-local-write
option de ligne de commande calibre-server, 320
--disable-log-not-found
option de ligne de commande calibre-server, 320
--disable-markup-chapter-headings
option de ligne de commande ebook-convert, 344
--disable-plugin
option de ligne de commande calibre-customize, 316
--disable-remove-fake-margins
option de ligne de commande ebook-convert, 345
--disable-renumber-headings
option de ligne de commande ebook-convert, 344
--disable-unwrap-lines
option de ligne de commande ebook-convert, 344
--disable-use-bonjour
option de ligne de commande calibre-server, 321
--disable-use-sendfile
option de ligne de commande calibre-server, 321
--display
option de ligne de commande calibredb-add_custom_column, 334
--displayed-fields
option de ligne de commande calibre-server, 320
--do-not-match-on-related-words
option de ligne de commande calibredb-fts_search, 338
--dont-asciiize
option de ligne de commande calibredb-export, 330
--dont-download-stylesheets
option de ligne de commande web2disk, 355
--dont-output-resources
option de ligne de commande lrf2lrs, 354
--dont-replace
option de ligne de commande calibredb-add_format, 329
--dont-save-cover
option de ligne de commande calibredb-export, 330
--dont-save-extra-files
option de ligne de commande calibredb-export, 330

--dont-split-on-page-breaks
option de ligne de commande ebook-convert, 340
--dont-update-metadata
option de ligne de commande calibredb-export, 330
--dont-verify-server-certificate
option de ligne de commande calibre-smtp, 324
--dont-write-opf
option de ligne de commande calibredb-export, 331
--duplicate-links-in-toc
option de ligne de commande ebook-convert, 346
--duplicates
option de ligne de commande calibredb-add, 328
--edit-book
option de ligne de commande calibre-debug, 318
--embed-all-fonts
option de ligne de commande ebook-convert, 342
--embed-font-family
option de ligne de commande ebook-convert, 342
--embed-fonts
option de ligne de commande ebook-polish, 351
--empty
option de ligne de commande calibredb-add, 328
--enable-allow-socket-preallocation
option de ligne de commande calibre-server, 320
--enable-auth
option de ligne de commande calibre-server, 320
--enable-fallback-to-detected-interface
option de ligne de commande calibre-server, 320
--enable-heuristics
option de ligne de commande ebook-convert, 344
--enable-local-write
option de ligne de commande calibre-server, 320
--enable-log-not-found
option de ligne de commande calibre-server, 320
--enable-plugin
option de ligne de commande calibre-customize, 316
--enable-use-bonjour
option de ligne de commande calibre-server, 321
--enable-use-sendfile
option de ligne de commande calibre-server, 321
--encoding
option de ligne de commande web2disk, 355
--encryption-method
option de ligne de commande calibre-smtp, 324
--epub-flatten
option de ligne de commande ebook-convert, 340
--epub-inline-toc
option de ligne de commande ebook-convert, 340
--epub-max-image-size
option de ligne de commande ebook-convert, 340
--epub-toc-at-end
option de ligne de commande ebook-convert, 340
--epub-version
option de ligne de commande ebook-convert, 340
--exclude-genre
option de ligne de commande calibredb-catalog, 332
--exclusion-rules
option de ligne de commande calibredb-catalog, 332
--exec-file
option de ligne de commande calibre-debug, 318
--expand-css
option de ligne de commande ebook-convert, 342
--explode-book
option de ligne de commande calibre-debug, 318
--export-all-calibre-data
option de ligne de commande calibre-debug, 318
--extra-css
option de ligne de commande ebook-convert, 342
--extract-to
option de ligne de commande ebook-convert, 341
--field
option de ligne de commande

calibredb-set_metadata, 330
--fields
option de ligne de commande calibredb-list, 326
--filter-css
option de ligne de commande ebook-convert, 342
--filter-regexp
option de ligne de commande web2disk, 356
--fix-multiprocessing
option de ligne de commande calibre-debug, 318
--flow-size
option de ligne de commande ebook-convert, 341
--font-size-mapping
option de ligne de commande ebook-convert, 342
--for-machine
option de ligne de commande calibredb-list, 326
--force
option de ligne de commande calibredb-remove_custom_column, 334
--force-reload
option de ligne de commande ebook-viewer, 352
--fork
option de ligne de commande calibre-smtp, 323
--formats
option de ligne de commande calibredb-export, 331
--from-opf
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 349
--full-screen
option de ligne de commande ebook-viewer, 352
--fullscreen
option de ligne de commande ebook-viewer, 352
--generate-authors
option de ligne de commande calibredb-catalog, 332
--generate-descriptions
option de ligne de commande calibredb-catalog, 332
--generate-genres
option de ligne de commande calibredb-catalog, 332
--generate-recently-added
option de ligne de commande calibredb-catalog, 332
--generate-series
option de ligne de commande calibredb-catalog, 332
--generate-titles
option de ligne de commande calibredb-catalog, 332
--genre-source-field
option de ligne de commande calibredb-catalog, 332
--get-cover
option de ligne de commande ebook-meta, 349
--gui
option de ligne de commande calibre-debug, 318
--gui-debug
option de ligne de commande calibre-debug, 318
--header-note-source-field
option de ligne de commande calibredb-catalog, 333
--help
option de ligne de commande, 326
option de ligne de commande calibre, 316
option de ligne de commande calibre-customize, 317
option de ligne de commande calibre-debug, 318
option de ligne de commande calibre-server, 321
option de ligne de commande calibre-smtp, 323
option de ligne de commande ebook-convert, 339
option de ligne de commande ebook-edit, 348
option de ligne de commande ebook-meta, 349
option de ligne de commande ebook-polish, 351
option de ligne de commande ebook-viewer, 352
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande lrf2lrs, 354
option de ligne de commande lrfviewer, 354
option de ligne de commande lrs2lrf,

355
option de ligne de commande web2disk, 356
--html-unwrap-factor
option de ligne de commande ebook-convert, 344
--identifier
option de ligne de commande calibredb-add, 328
option de ligne de commande ebook-meta, 349
option de ligne de commande fetch-ebook-metadata, 353
--ids
option de ligne de commande calibredb-catalog, 331
--ignore
option de ligne de commande calibredb-add, 328
--ignore_extensions
option de ligne de commande calibredb-check_library, 335
--ignore_names
option de ligne de commande calibredb-check_library, 335
--ignore-plugins
option de ligne de commande calibre, 316
--ignored-fields
option de ligne de commande calibre-server, 321
--implode-book
option de ligne de commande calibre-debug, 318
--import-calibre-data
option de ligne de commande calibre-debug, 318
--include-snippets
option de ligne de commande calibredb-fts_search, 338
--index
option de ligne de commande ebook-meta, 350
--indexing-speed
option de ligne de commande calibredb-fts_index, 338
--indexing-threshold
option de ligne de commande calibredb-fts_search, 338
--input-encoding
option de ligne de commande ebook-convert, 340
--input-profile
option de ligne de commande ebook-convert, 339
--insert-blank-line
option de ligne de commande ebook-convert, 342
--insert-blank-line-size
option de ligne de commande ebook-convert, 342
--insert-metadata
option de ligne de commande ebook-convert, 345
--inspect-mobi
option de ligne de commande calibre-debug, 318
--is-multiple
option de ligne de commande calibredb-add_custom_column, 334
--isbn
option de ligne de commande calibredb-add, 328
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 350
option de ligne de commande fetch-ebook-metadata, 353
--item_count
option de ligne de commande calibredb-list_categories, 336
--jacket
option de ligne de commande ebook-polish, 351
--keep-ligatures
option de ligne de commande ebook-convert, 342
--language
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 350
--languages
option de ligne de commande calibredb-add, 328
--level1-toc
option de ligne de commande ebook-convert, 346
--level2-toc
option de ligne de commande ebook-convert, 346
--level3-toc
option de ligne de commande ebook-convert, 346
--library-path
option de ligne de commande, 326
--limit

option de ligne de commande calibredb-list, 327
option de ligne de commande calibredb-search, 337
--line-height
option de ligne de commande ebook-convert, 342
--line-width
option de ligne de commande calibredb-list, 327
--linearize-tables
option de ligne de commande ebook-convert, 342
--list-fields
option de ligne de commande calibredb-set_metadata, 330
--list-plugins
option de ligne de commande calibre-customize, 317
--list-recipes
option de ligne de commande ebook-convert, 339
--listen-on
option de ligne de commande calibre-server, 321
--localhost
option de ligne de commande calibre-smtp, 323
--log
option de ligne de commande calibre-server, 321
--lrf-bookid
option de ligne de commande ebook-meta, 350
--lrs
option de ligne de commande lrs2lrf, 355
--manage-users
option de ligne de commande calibre-server, 321
--margin-bottom
option de ligne de commande ebook-convert, 343
--margin-left
option de ligne de commande ebook-convert, 343
--margin-right
option de ligne de commande ebook-convert, 343
--margin-top
option de ligne de commande ebook-convert, 343
--match-end-marker
option de ligne de commande calibredb-fts_search, 338
--match-regexp
option de ligne de commande web2disk, 356
--match-start-marker
option de ligne de commande calibredb-fts_search, 338
--max-files
option de ligne de commande web2disk, 356
--max-header-line-size
option de ligne de commande calibre-server, 321
--max-job-time
option de ligne de commande calibre-server, 321
--max-jobs
option de ligne de commande calibre-server, 321
--max-log-size
option de ligne de commande calibre-server, 321
--max-opds-items
option de ligne de commande calibre-server, 321
--max-opds-ungrouped-items
option de ligne de commande calibre-server, 322
--max-recursions
option de ligne de commande web2disk, 356
--max-request-body-size
option de ligne de commande calibre-server, 322
--max-toc-links
option de ligne de commande ebook-convert, 346
--merge-comments-rule
option de ligne de commande calibredb-catalog, 333
--minimum-line-height
option de ligne de commande ebook-convert, 343
--new-instance
option de ligne de commande ebook-viewer, 352
--no-chapters-in-toc
option de ligne de commande ebook-convert, 346
--no-default-epub-cover
option de ligne de commande ebook-convert, 341
--no-svg-cover

option de ligne de commande ebook-convert, 341
--no-update-check
option de ligne de commande calibre, 316
--num-per-page
option de ligne de commande calibre-server, 322
--one-book-per-directory
option de ligne de commande calibredb-add, 328
--only-formats
option de ligne de commande calibredb-embed_metadata, 337
--open-at
option de ligne de commande ebook-viewer, 352
--opf
option de ligne de commande ebook-polish, 351
option de ligne de commande fetch-ebook-metadata, 353
--outbox
option de ligne de commande calibre-smtp, 323
--output
option de ligne de commande lrf2lrs, 354
option de ligne de commande lrs2lrf, 355
--output-format
option de ligne de commande calibredb-fts_search, 338
--output-profile
option de ligne de commande calibredb-catalog, 333
option de ligne de commande ebook-convert, 340
--page-breaks-before
option de ligne de commande ebook-convert, 346
--password
option de ligne de commande, 326
option de ligne de commande calibre-smtp, 324
--paths
option de ligne de commande calibre-debug, 318
--permanent
option de ligne de commande calibredb-remove, 329
--pidfile
option de ligne de commande calibre-server, 322
--port
option de ligne de commande calibre-server, 322
option de ligne de commande calibre-smtp, 324
--prefer-metadata-cover
option de ligne de commande ebook-convert, 346
--prefix
option de ligne de commande calibredb-list, 327
--prefix-rules
option de ligne de commande calibredb-catalog, 333
--preserve-cover-aspect-ratio
option de ligne de commande ebook-convert, 341
--preset
option de ligne de commande calibredb-catalog, 333
--pretty-print
option de ligne de commande ebook-convert, 341
--profile
option de ligne de commande lrfviewer, 354
--progress
option de ligne de commande calibredb-export, 331
--pubdate
option de ligne de commande ebook-convert, 347
--publisher
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 350
--raise-window
option de ligne de commande ebook-viewer, 352
--rating
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 350
--read-metadata-from-opf
option de ligne de commande ebook-convert, 347
--really-do-it
option de ligne de commande calibredb-restore_database, 335
--recurse
option de ligne de commande calibredb-add, 328

--relay
option de ligne de commande calibre-smtp, 324
--remove-first-image
option de ligne de commande ebook-convert, 346
--remove-jacket
option de ligne de commande ebook-polish, 351
--remove-paragraph-spacing
option de ligne de commande ebook-convert, 343
--remove-paragraph-spacing-indent-size
option de ligne de commande ebook-convert, 343
--remove-plugin
option de ligne de commande calibre-customize, 317
--remove-soft-hyphens
option de ligne de commande ebook-polish, 351
--remove-unused-css
option de ligne de commande ebook-polish, 351
--replace-scene-breaks
option de ligne de commande ebook-convert, 344
--replace-whitespace
option de ligne de commande calibredb-export, 331
--report
option de ligne de commande calibredb-check_library, 335
--restrict-to
option de ligne de commande calibredb-fts_search, 338
--run-plugin
option de ligne de commande calibre-debug, 318
--run-test
option de ligne de commande calibre-debug, 318
--run-without-debug
option de ligne de commande calibre-debug, 319
--search
option de ligne de commande calibredb-catalog, 331
option de ligne de commande calibredb-list, 327
--search-replace
option de ligne de commande ebook-convert, 345
--search-the-net-urls
option de ligne de commande calibre-server, 322
--select-text
option de ligne de commande ebook-edit, 348
--separator
option de ligne de commande calibredb-list, 327
--series
option de ligne de commande calibredb-add, 328
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 350
--series-index
option de ligne de commande calibredb-add, 328
option de ligne de commande ebook-convert, 347
--shutdown-running-calibre
option de ligne de commande calibre, 316
option de ligne de commande calibre-debug, 319
--shutdown-timeout
option de ligne de commande calibre-server, 322
--single-dir
option de ligne de commande calibredb-export, 331
--smarten-punctuation
option de ligne de commande ebook-convert, 343
option de ligne de commande ebook-polish, 351
--sort-by
option de ligne de commande calibredb-list, 327
--sr1-replace
option de ligne de commande ebook-convert, 345
--sr1-search
option de ligne de commande ebook-convert, 345
--sr2-replace
option de ligne de commande ebook-convert, 345
--sr2-search
option de ligne de commande ebook-convert, 345
--sr3-replace
option de ligne de commande ebook-convert, 345

--sr3-search
option de ligne de commande ebook-convert, 345
--ssl-certfile
option de ligne de commande calibre-server, 322
--ssl-keyfile
option de ligne de commande calibre-server, 322
--start-in-tray
option de ligne de commande calibre, 316
--start-reading-at
option de ligne de commande ebook-convert, 346
--subject
option de ligne de commande calibre-smtp, 324
--subset-embedded-fonts
option de ligne de commande ebook-convert, 343
--subset-font
option de ligne de commande calibre-debug, 319
--subset-fonts
option de ligne de commande ebook-polish, 351
--tags
option de ligne de commande calibredb-add, 328
option de ligne de commande ebook-convert, 347
option de ligne de commande ebook-meta, 350
--template
option de ligne de commande calibredb-export, 331
option de ligne de commande calibredb-list, 327
--template_file
option de ligne de commande calibredb-list, 327
--template_heading
option de ligne de commande calibredb-list, 327
--test-build
option de ligne de commande calibre-debug, 319
--thumb-width
option de ligne de commande calibredb-catalog, 333
--timefmt
option de ligne de commande calibredb-export, 331
--timeout
option de ligne de commande, 326
option de ligne de commande calibre-server, 322
option de ligne de commande calibre-smtp, 323
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande web2disk, 356
--timestamp
option de ligne de commande ebook-convert, 348
--title
option de ligne de commande calibredb-add, 328
option de ligne de commande ebook-convert, 348
option de ligne de commande ebook-meta, 350
option de ligne de commande fetch-ebook-metadata, 353
--title-sort
option de ligne de commande ebook-convert, 348
option de ligne de commande ebook-meta, 350
--to-dir
option de ligne de commande calibredb-export, 331
--to-lowercase
option de ligne de commande calibredb-export, 331
--to-opf
option de ligne de commande ebook-meta, 350
--toc-filter
option de ligne de commande ebook-convert, 346
--toc-threshold
option de ligne de commande ebook-convert, 347
--toc-title
option de ligne de commande ebook-convert, 341
--transform-css-rules
option de ligne de commande ebook-convert, 343
--transform-html-rules
option de ligne de commande ebook-convert, 344
--trusted-ips
option de ligne de commande calibre-server, 322

--unsmarten-punctuation
option de ligne de commande ebook-convert, 344
--upgrade-book
option de ligne de commande ebook-polish, 352
--url-prefix
option de ligne de commande calibre-server, 322
--use-auto-toc
option de ligne de commande ebook-convert, 347
--use-existing-cover
option de ligne de commande calibredb-catalog, 333
--userdb
option de ligne de commande calibre-server, 322
--username
option de ligne de commande, 326
option de ligne de commande calibre-smtp, 324
--vacuum-fts-db
option de ligne de commande calibredb-check_library, 335
--verbose
option de ligne de commande calibre, 316
option de ligne de commande calibre-smtp, 323
option de ligne de commande calibredb-catalog, 331
option de ligne de commande ebook-convert, 348
option de ligne de commande ebook-polish, 352
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande lrf2lrs, 354
option de ligne de commande lrfviewer, 354
option de ligne de commande lrs2lrf, 355
option de ligne de commande web2disk, 356
--version
option de ligne de commande, 326
option de ligne de commande calibre, 316
option de ligne de commande calibre-customize, 317
option de ligne de commande calibre-debug, 319
option de ligne de commande calibre-server, 322
option de ligne de commande calibre-smtp, 323
option de ligne de commande ebook-convert, 340
option de ligne de commande ebook-edit, 348
option de ligne de commande ebook-meta, 350
option de ligne de commande ebook-polish, 352
option de ligne de commande ebook-viewer, 352
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande lrf2lrs, 354
option de ligne de commande lrfviewer, 354
option de ligne de commande lrs2lrf, 355
option de ligne de commande web2disk, 356
--viewer
option de ligne de commande calibre-debug, 319
--visual-debug
option de ligne de commande lrfviewer, 354
--wait-for-completion
option de ligne de commande calibredb-fts_index, 338
--white-background
option de ligne de commande lrfviewer, 355
--width
option de ligne de commande calibredb-list_categories, 336
--with-library
option de ligne de commande, 326
option de ligne de commande calibre, 316
--worker-count
option de ligne de commande calibre-server, 323
-a
option de ligne de commande calibre-customize, 316
option de ligne de commande calibre-smtp, 324
option de ligne de commande calibredb-add, 327
option de ligne de commande

calibredb-set_custom, 335
option de ligne de commande ebook-meta, 349
option de ligne de commande fetch-ebook-metadata, 353
-b
option de ligne de commande calibre-customize, 316
-c
option de ligne de commande calibre-debug, 317
option de ligne de commande calibredb-add, 327
option de ligne de commande calibredb-check_library, 335
option de ligne de commande calibredb-list_categories, 336
option de ligne de commande ebook-meta, 349
option de ligne de commande ebook-polish, 351
option de ligne de commande fetch-ebook-metadata, 353
-d
option de ligne de commande calibre-debug, 317
option de ligne de commande calibredb-add, 328
option de ligne de commande calibredb-custom_columns, 334
option de ligne de commande ebook-convert, 348
option de ligne de commande ebook-meta, 349
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande web2disk, 355
-e
option de ligne de commande calibre-debug, 318
option de ligne de commande calibre-smtp, 324
option de ligne de commande calibredb-add, 328
option de ligne de commande calibredb-check_library, 335
option de ligne de commande ebook-polish, 351
-f
option de ligne de commande calibre-debug, 319
option de ligne de commande calibre-smtp, 323
option de ligne de commande calibredb-embed_metadata, 337
option de ligne de commande calibredb-list, 326
option de ligne de commande calibredb-remove_custom_column, 334
option de ligne de commande calibredb-set_metadata, 330
option de ligne de commande ebook-polish, 351
option de ligne de commande ebook-viewer, 352
-g
option de ligne de commande calibre-debug, 318
-h
option de ligne de commande, 326
option de ligne de commande calibre, 316
option de ligne de commande calibre-customize, 317
option de ligne de commande calibre-debug, 318
option de ligne de commande calibre-server, 321
option de ligne de commande calibre-smtp, 323
option de ligne de commande ebook-convert, 339
option de ligne de commande ebook-edit, 348
option de ligne de commande ebook-meta, 349
option de ligne de commande ebook-polish, 351
option de ligne de commande ebook-viewer, 352
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande lrf2lrs, 354
option de ligne de commande lrfviewer, 354
option de ligne de commande lrs2lrf, 355
option de ligne de commande web2disk, 356
-i
option de ligne de commande calibre-debug, 318
option de ligne de commande calibredb-add, 328
option de ligne de commande

calibredb-catalog, 331
option de ligne de commande calibredb-list_categories, 336
option de ligne de commande ebook-meta, 350
option de ligne de commande ebook-polish, 351
option de ligne de commande fetch-ebook-metadata, 353
-j
option de ligne de commande ebook-polish, 351
-k
option de ligne de commande ebook-meta, 349
-l
option de ligne de commande calibre-customize, 317
option de ligne de commande calibre-smtp, 323
option de ligne de commande calibredb-add, 328
option de ligne de commande calibredb-search, 337
option de ligne de commande calibredb-set_metadata, 330
option de ligne de commande ebook-meta, 350
-m
option de ligne de commande calibre-debug, 318
option de ligne de commande calibredb-add, 327
option de ligne de commande ebook-convert, 347
-n
option de ligne de commande calibredb-check_library, 335
option de ligne de commande web2disk, 356
-o
option de ligne de commande calibre-smtp, 323
option de ligne de commande ebook-polish, 351
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande lrf2lrs, 354
option de ligne de commande lrs2lrf, 355
-p
option de ligne de commande calibre-smtp, 324
option de ligne de commande ebook-meta, 350
option de ligne de commande ebook-polish, 351
option de ligne de commande fetch-ebook-metadata, 353
-r
option de ligne de commande calibre-customize, 317
option de ligne de commande calibre-debug, 318
option de ligne de commande calibre-smtp, 324
option de ligne de commande calibredb-add, 328
option de ligne de commande calibredb-check_library, 335
option de ligne de commande calibredb-list_categories, 336
option de ligne de commande calibredb-restore_database, 335
option de ligne de commande ebook-meta, 350
option de ligne de commande web2disk, 356
-s
option de ligne de commande calibre, 316
option de ligne de commande calibre-debug, 319
option de ligne de commande calibre-smtp, 324
option de ligne de commande calibredb-add, 328
option de ligne de commande calibredb-catalog, 331
option de ligne de commande calibredb-list, 327
option de ligne de commande ebook-meta, 350
-t
option de ligne de commande calibre-debug, 318
option de ligne de commande calibre-smtp, 323
option de ligne de commande calibredb-add, 328
option de ligne de commande calibredb-list, 327
option de ligne de commande ebook-meta, 350
option de ligne de commande fetch-ebook-metadata, 353
option de ligne de commande web2disk,

356
-u
option de ligne de commande calibre-smtp, 324
option de ligne de commande ebook-polish, 351
-v
option de ligne de commande calibre, 316
option de ligne de commande calibre-smtp, 323
option de ligne de commande calibredb-catalog, 331
option de ligne de commande ebook-convert, 348
option de ligne de commande fetch-ebook-metadata, 353
-w
option de ligne de commande calibre-debug, 319
option de ligne de commande calibredb-list, 327
option de ligne de commande calibredb-list_categories, 336
-x
option de ligne de commande calibre-debug, 318

A

abort_article() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 42
abort_recipe_processing() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 42
abspath_to_name() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
accept_drag_move_event() (*méthode calibre.gui2.actions.InterfaceAction*), 296
accept_enter_event() (*méthode calibre.gui2.actions.InterfaceAction*), 296
accepts_drops (*attribut calibre.gui2.actions.InterfaceAction*), 296
action_add_menu (*attribut calibre.gui2.actions.InterfaceAction*), 296
action_menu_clone_qaction (*attribut calibre.gui2.actions.InterfaceAction*), 296
action_shortcut_name (*attribut calibre.gui2.actions.InterfaceAction*), 296
action_spec (*attribut calibre.gui2.actions.InterfaceAction*), 296
action_type (*attribut calibre.gui2.actions.InterfaceAction*), 296
add_annotation_to_library() (*méthode calibre.devices.usbms.device.Device*), 293
add_book() (*méthode calibre.devices.interface.BookList*), 289
add_books() (*méthode calibre.db.cache.Cache*), 367
add_books_to_metadata() (*méthode calibre.devices.usbms.driver.USBMS*), 294
add_books_to_metadata() (*méthode de la classe calibre.devices.interface.DevicePlugin*), 286
add_custom_book_data() (*méthode calibre.db.cache.Cache*), 367
add_extra_files() (*méthode calibre.db.cache.Cache*), 367
add_file() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
add_format() (*méthode calibre.db.cache.Cache*), 367
add_listener() (*méthode calibre.db.cache.Cache*), 367
add_name_to_manifest() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
add_notes_resource() (*méthode calibre.db.cache.Cache*), 367
add_or_replace_jacket() (*dans le module calibre.ebooks.oeb.polish.jacket*), 382
add_properties() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
add_savepoint() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
add_toc_thumbnail() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 42
adeify_images() (*méthode de la classe calibre.web.feeds.news.BasicNewsRecipe*), 43
all_book_ids() (*méthode calibre.db.cache.Cache*), 367
all_field_for() (*méthode calibre.db.cache.Cache*), 368
all_field_ids() (*méthode calibre.db.cache.Cache*), 368
all_field_keys() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
all_field_names() (*méthode calibre.db.cache.Cache*), 368
all_non_none_fields() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
allowed_in_menu (*attribut calibre.gui2.tweak_book.plugin.Tool*), 384
allowed_in_toolbar (*attribut*

calibre.gui2.tweak_book.plugin.Tool), 384
API, **393**
apply_container_update_to_gui() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
apply_unique_properties() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
articles_are_obfuscated (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
ASK_TO_ALLOW_CONNECT (*attribut calibre.devices.interface.DevicePlugin*), 282
author (*attribut calibre.customize.InterfaceActionBase*), 298
author (*attribut calibre.customize.MetadataReaderPlugin*), 272
author (*attribut calibre.customize.MetadataWriterPlugin*), 273
author (*attribut calibre.customize.Plugin*), 269
author (*attribut calibre.customize.PreferencesPlugin*), 299
author (*attribut calibre.devices.usbms.driver.USBMS*), 293
author (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
author_data() (*méthode calibre.db.cache.Cache*), 368
author_sort_from_authors() (*méthode calibre.db.cache.Cache*), 368
auto_cleanup (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
auto_cleanup_keep (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
auto_repeat (*attribut calibre.gui2.actions.InterfaceAction*), 296
auto_trim_covers (*attribut calibre.ebooks.metadata.sources.base.Source*), 275

B

BACKLOADING_ERROR_MESSAGE (*attribut calibre.devices.usbms.device.Device*), 291
BasicNewsRecipe (*classe dans calibre.web.feeds.news*), 42
BCD (*attribut calibre.devices.interface.DevicePlugin*), 281
BCD (*attribut calibre.devices.usbms.device.Device*), 290
book_class (*attribut calibre.devices.usbms.driver.USBMS*), 293
book_created (*attribut calibre.db.cache.Cache.EventType*), 366
book_edited (*attribut calibre.db.cache.Cache.EventType*), 366
book_type (*attribut calibre.ebooks.oeb.polish.container.Container*), 377
BookList (*classe dans calibre.devices.interface*), 289
booklist_class (*attribut calibre.devices.usbms.driver.USBMS*), 293
books() (*méthode calibre.devices.interface.DevicePlugin*), 285
books() (*méthode calibre.devices.usbms.driver.USBMS*), 294
books_for_field() (*méthode calibre.db.cache.Cache*), 368
books_in_virtual_library() (*méthode calibre.db.cache.Cache*), 368
books_removed (*attribut calibre.db.cache.Cache.EventType*), 366
Boss (*classe dans calibre.gui2.tweak_book.boss*), 386
boss (*propriété calibre.gui2.tweak_book.plugin.Tool*), 385
BuiltinAdd (*classe dans calibre.utils.formatter_functions*), 200
BuiltinAnd (*classe dans calibre.utils.formatter_functions*), 201
BuiltinAnnotationCount (*classe dans calibre.utils.formatter_functions*), 204
BuiltinApproximateFormats (*classe dans calibre.utils.formatter_functions*), 204
BuiltinArguments (*classe dans calibre.utils.formatter_functions*), 214
BuiltinAssign (*classe dans calibre.utils.formatter_functions*), 214
BuiltinAuthorLinks (*classe dans calibre.utils.formatter_functions*), 204
BuiltinAuthorSorts (*classe dans calibre.utils.formatter_functions*), 204
BuiltinBookCount (*classe dans calibre.utils.formatter_functions*), 219
BuiltinBooksize (*classe dans calibre.utils.formatter_functions*), 205
BuiltinBookValues (*classe dans calibre.utils.formatter_functions*), 219
BuiltinCapitalize (*classe dans calibre.utils.formatter_functions*), 216
BuiltinCeiling (*classe dans calibre.utils.formatter_functions*), 200
BuiltinCharacter (*classe dans calibre.utils.formatter_functions*), 217
BuiltinCheckYesNo (*classe dans calibre.utils.formatter_functions*), 208
BuiltinCmp (*classe dans calibre.utils.formatter_functions*), 215
BuiltinConnectedDeviceName (*classe dans calibre.utils.formatter_functions*), 205
BuiltinConnectedDeviceUUID (*classe dans calibre.utils.formatter_functions*), 205
BuiltinContains (*classe dans calibre.utils.formatter_functions*), 208
BuiltinCount (*classe dans*

calibre.utils.formatter_functions), 211
BuiltinCurrentLibraryName (*classe dans calibre.utils.formatter_functions*), 205
BuiltinCurrentLibraryPath (*classe dans calibre.utils.formatter_functions*), 205
BuiltinCurrentVirtualLibraryName (*classe dans calibre.utils.formatter_functions*), 205
BuiltinDateArithmetic (*classe dans calibre.utils.formatter_functions*), 201
BuiltinDaysBetween (*classe dans calibre.utils.formatter_functions*), 202
BuiltinDivide (*classe dans calibre.utils.formatter_functions*), 200
BuiltinEval (*classe dans calibre.utils.formatter_functions*), 215
BuiltinExtraFileModtime (*classe dans calibre.utils.formatter_functions*), 219
BuiltinExtraFileNames (*classe dans calibre.utils.formatter_functions*), 219
BuiltinExtraFileSize (*classe dans calibre.utils.formatter_functions*), 220
BuiltinField (*classe dans calibre.utils.formatter_functions*), 206
BuiltinFieldExists (*classe dans calibre.utils.formatter_functions*), 208
BuiltinFinishFormatting (*classe dans calibre.utils.formatter_functions*), 202
BuiltinFirstMatchingCmp (*classe dans calibre.utils.formatter_functions*), 216
BuiltinFirstNonEmpty (*classe dans calibre.utils.formatter_functions*), 209
BuiltinFloor (*classe dans calibre.utils.formatter_functions*), 200
BuiltinFormatDate (*classe dans calibre.utils.formatter_functions*), 202
BuiltinFormatDateField (*classe dans calibre.utils.formatter_functions*), 203
BuiltinFormatNumber (*classe dans calibre.utils.formatter_functions*), 203
BuiltinFormatsModtimes (*classe dans calibre.utils.formatter_functions*), 206
BuiltinFormatsPaths (*classe dans calibre.utils.formatter_functions*), 206
BuiltinFormatsSizes (*classe dans calibre.utils.formatter_functions*), 206
BuiltinFractionalPart (*classe dans calibre.utils.formatter_functions*), 200
BuiltinGetLink (*classe dans calibre.utils.formatter_functions*), 220
BuiltinGetNote (*classe dans calibre.utils.formatter_functions*), 220
BuiltinGlobals (*classe dans calibre.utils.formatter_functions*), 215
BuiltinHasCover (*classe dans calibre.utils.formatter_functions*), 206
BuiltinHasExtraFiles (*classe dans calibre.utils.formatter_functions*), 220
BuiltinHasNote (*classe dans calibre.utils.formatter_functions*), 220
BuiltinHumanReadable (*classe dans calibre.utils.formatter_functions*), 203
BuiltinIdentifierInList (*classe dans calibre.utils.formatter_functions*), 210
BuiltinIfempty (*classe dans calibre.utils.formatter_functions*), 209
BuiltinInList (*classe dans calibre.utils.formatter_functions*), 210
BuiltinIsMarked (*classe dans calibre.utils.formatter_functions*), 206
BuiltinLanguageCodes (*classe dans calibre.utils.formatter_functions*), 207
BuiltinLanguageStrings (*classe dans calibre.utils.formatter_functions*), 207
BuiltinListCountMatching (*classe dans calibre.utils.formatter_functions*), 211
BuiltinListDifference (*classe dans calibre.utils.formatter_functions*), 211
BuiltinListEquals (*classe dans calibre.utils.formatter_functions*), 211
BuiltinListIntersection (*classe dans calibre.utils.formatter_functions*), 211
BuiltinListitem (*classe dans calibre.utils.formatter_functions*), 210
BuiltinListJoin (*classe dans calibre.utils.formatter_functions*), 212
BuiltinListRe (*classe dans calibre.utils.formatter_functions*), 212
BuiltinListReGroup (*classe dans calibre.utils.formatter_functions*), 212
BuiltinListRemoveDuplicates (*classe dans calibre.utils.formatter_functions*), 213
BuiltinListSort (*classe dans calibre.utils.formatter_functions*), 213
BuiltinListSplit (*classe dans calibre.utils.formatter_functions*), 213
BuiltinListUnion (*classe dans calibre.utils.formatter_functions*), 213
BuiltinLookup (*classe dans calibre.utils.formatter_functions*), 209
BuiltinLowercase (*classe dans calibre.utils.formatter_functions*), 216
BuiltinMod (*classe dans calibre.utils.formatter_functions*), 200
BuiltinMultiply (*classe dans calibre.utils.formatter_functions*), 200
BuiltinNot (*classe dans calibre.utils.formatter_functions*), 201
BuiltinOndevice (*classe dans*

calibre.utils.formatter_functions), 207
BuiltinOr (*classe dans calibre.utils.formatter_functions*), 201
BuiltinPrint (*classe dans calibre.utils.formatter_functions*), 215
BuiltinRange (*classe dans calibre.utils.formatter_functions*), 213
BuiltinRatingToStars (*classe dans calibre.utils.formatter_functions*), 203
BuiltinRawField (*classe dans calibre.utils.formatter_functions*), 207
BuiltinRawList (*classe dans calibre.utils.formatter_functions*), 207
BuiltinRe (*classe dans calibre.utils.formatter_functions*), 217
BuiltinReGroup (*classe dans calibre.utils.formatter_functions*), 217
BuiltinRound (*classe dans calibre.utils.formatter_functions*), 201
BuiltinSelect (*classe dans calibre.utils.formatter_functions*), 210
BuiltinSeriesSort (*classe dans calibre.utils.formatter_functions*), 207
BuiltinSetGlobals (*classe dans calibre.utils.formatter_functions*), 220
BuiltinShorten (*classe dans calibre.utils.formatter_functions*), 217
BuiltinStrcat (*classe dans calibre.utils.formatter_functions*), 218
BuiltinStrcatMax (*classe dans calibre.utils.formatter_functions*), 218
BuiltinStrcmp (*classe dans calibre.utils.formatter_functions*), 216
BuiltinStrcmpcase (*classe dans calibre.utils.formatter_functions*), 216
BuiltinStrInList (*classe dans calibre.utils.formatter_functions*), 210
BuiltinStrlen (*classe dans calibre.utils.formatter_functions*), 218
BuiltinSubitems (*classe dans calibre.utils.formatter_functions*), 214
BuiltinSublist (*classe dans calibre.utils.formatter_functions*), 214
BuiltinSubstr (*classe dans calibre.utils.formatter_functions*), 218
BuiltinSubtract (*classe dans calibre.utils.formatter_functions*), 201
BuiltinSwapAroundArticles (*classe dans calibre.utils.formatter_functions*), 218
BuiltinSwapAroundComma (*classe dans calibre.utils.formatter_functions*), 218
BuiltinSwitch (*classe dans calibre.utils.formatter_functions*), 209
BuiltinSwitchIf (*classe dans calibre.utils.formatter_functions*), 209
BuiltinTemplate (*classe dans calibre.utils.formatter_functions*), 215
BuiltinTest (*classe dans calibre.utils.formatter_functions*), 209
BuiltinTitlecase (*classe dans calibre.utils.formatter_functions*), 216
BuiltinToday (*classe dans calibre.utils.formatter_functions*), 202
BuiltinToHex (*classe dans calibre.utils.formatter_functions*), 219
BuiltinTransliterate (*classe dans calibre.utils.formatter_functions*), 219
BuiltinUppercase (*classe dans calibre.utils.formatter_functions*), 217
BuiltinUrlsFromIdentifiers (*classe dans calibre.utils.formatter_functions*), 203
BuiltinUserCategories (*classe dans calibre.utils.formatter_functions*), 208
BuiltinVirtualLibraries (*classe dans calibre.utils.formatter_functions*), 208

C

Cache (*classe dans calibre.db.cache*), 366
Cache.EventType (*classe dans calibre.db.cache*), 366
cached_cover_url_is_reliable (*attribut calibre.ebooks.metadata.sources.base.Source*), 275
calibre.customize
    module, 268
calibre.customize.conversion
    module, 278
calibre.db.cache
    module, 366
calibre.devices.interface
    module, 281
calibre.ebooks.metadata.book.base
    module, 221
calibre.ebooks.metadata.sources.base
    module, 274
calibre.ebooks.oeb.polish.container
    module, 376
calibre.ebooks.oeb.polish.cover
    module, 383
calibre.ebooks.oeb.polish.css
    module, 383
calibre.ebooks.oeb.polish.jacket
    module, 382
calibre.ebooks.oeb.polish.pretty
    module, 381
calibre.ebooks.oeb.polish.replace
    module, 381
calibre.ebooks.oeb.polish.split
    module, 382

calibre.ebooks.oeb.polish.toc module, 384
calibre.gui2.tweak_book.boss module, 386
calibre.utils.formatter_functions module, 199
calibre.web.feeds.news module, 42
can_be_disabled (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
can_be_disabled (*attribut calibre.customize.conversion.OutputFormatPlugin*), 280
can_be_disabled (*attribut calibre.customize.InterfaceActionBase*), 298
can_be_disabled (*attribut calibre.customize.Plugin*), 269
can_be_disabled (*attribut calibre.customize.PreferencesPlugin*), 299
CAN_DO_DEVICE_DB_PLUGBOARD (*attribut calibre.devices.interface.DevicePlugin*), 281
can_get_multiple_covers (*attribut calibre.ebooks.metadata.sources.base.Source*), 275
can_handle() (*méthode calibre.devices.interface.DevicePlugin*), 283
can_handle_windows() (*méthode calibre.devices.interface.DevicePlugin*), 283
can_handle_windows() (*méthode calibre.devices.usbms.device.Device*), 292
CAN_SET_METADATA (*attribut calibre.devices.interface.DevicePlugin*), 281
CAN_SET_METADATA (*attribut calibre.devices.usbms.driver.USBMS*), 293
canonicalize_internal_url() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
capabilities (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
card_prefix() (*méthode calibre.devices.interface.DevicePlugin*), 284
card_prefix() (*méthode calibre.devices.usbms.device.Device*), 291
CatalogPlugin (*classe dans calibre.customize*), 273
category (*attribut calibre.customize.PreferencesPlugin*), 299
category_order (*attribut calibre.customize.PreferencesPlugin*), 299
center_navbar (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
change_font() (*dans le module calibre.ebooks.oeb.polish.fonts*), 383
changed_signal (*attribut calibre.gui2.preferences.ConfigWidgetBase*), 300
changed_signal (*attribut calibre.gui2.preferences.ConfigWidgetInterface*), 299
clean_downloaded_metadata() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
cleanup() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
CLI (*classe dans calibre.devices.usbms.cli*), 293
cli_main() (*méthode calibre.customize.Plugin*), 270
cli_options (*attribut calibre.customize.CatalogPlugin*), 273
clone_browser() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
close_editor() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
commit() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
commit() (*méthode calibre.gui2.preferences.ConfigWidgetBase*), 301
commit() (*méthode calibre.gui2.preferences.ConfigWidgetInterface*), 300
commit_all_editors_to_container() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
commit_item() (*méthode calibre.ebooks.oeb.polish.container.Container*), 377
common_options (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
common_options (*attribut calibre.customize.conversion.OutputFormatPlugin*), 280
compress_covers() (*méthode calibre.db.cache.Cache*), 368
compress_news_images (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
compress_news_images_auto_size (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
compress_news_images_max_size (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 47
config_help_message (*attribut calibre.ebooks.metadata.sources.base.Source*), 275
config_widget (*attribut calibre.customize.PreferencesPlugin*), 299
config_widget() (*méthode calibre.customize.Plugin*), 269

config_widget() (*méthode calibre.ebooks.metadata.sources.base.Source*), 275
config_widget() (*méthode de la classe calibre.devices.interface.DevicePlugin*), 286
ConfigWidgetBase (*classe dans calibre.gui2.preferences*), 300
ConfigWidgetInterface (*classe dans calibre.gui2.preferences*), 299
Container (*classe dans calibre.ebooks.oeb.polish.container*), 377
contains(), **169**
conversion_options (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
convert() (*méthode calibre.customize.conversion.InputFormatPlugin*), 279
convert() (*méthode calibre.customize.conversion.OutputFormatPlugin*), 280
copy_cover_to() (*méthode calibre.db.cache.Cache*), 368
copy_format_to() (*méthode calibre.db.cache.Cache*), 368
core_usage (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
cover() (*méthode calibre.db.cache.Cache*), 368
cover_margins (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
create_action() (*méthode calibre.gui2.tweak_book.plugin.Tool*), 385
create_inline_toc() (*dans le module calibre.ebooks.oeb.polish.toc*), 384
create_menu_action() (*méthode calibre.gui2.actions.InterfaceAction*), 297
create_widget() (*méthode calibre.customize.PreferencesPlugin*), 299
CSS, **393**
current_container (*propriété calibre.gui2.tweak_book.plugin.Tool*), 385
currently_editing (*propriété calibre.gui2.tweak_book.boss.Boss*), 386
custom_field_keys() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
customization_help() (*méthode calibre.customize.Plugin*), 270
customization_help() (*méthode calibre.ebooks.metadata.sources.base.Source*), 275

D

data_for_find_identical_books() (*méthode calibre.db.cache.Cache*), 369
data_for_has_book() (*méthode calibre.db.cache.Cache*), 369
debug_managed_device_detection() (*méthode calibre.devices.interface.DevicePlugin*), 283
deepcopy() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
default_cover() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
delay (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
delete_books() (*méthode calibre.devices.interface.DevicePlugin*), 286
delete_books() (*méthode calibre.devices.usbms.driver.USBMS*), 295
delete_custom_book_data() (*méthode calibre.db.cache.Cache*), 369
delete_trash_entry() (*méthode calibre.db.cache.Cache*), 369
description (*attribut calibre.customize.Plugin*), 268
description (*attribut calibre.customize.PreferencesPlugin*), 299
description (*attribut calibre.devices.usbms.driver.USBMS*), 293
description (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
description (*propriété calibre.customize.conversion.OutputFormatPlugin*), 280
detect_managed_devices() (*méthode calibre.devices.interface.DevicePlugin*), 283
Device (*classe dans calibre.devices.usbms.device*), 290
DevicePlugin (*classe dans calibre.devices.interface*), 281
dirty() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
do_user_config() (*méthode calibre.customize.Plugin*), 269
dont_add_to (*attribut calibre.gui2.actions.InterfaceAction*), 296
dont_remove_from (*attribut calibre.gui2.actions.InterfaceAction*), 296
download() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
download_cover() (*méthode calibre.ebooks.metadata.sources.base.Source*), 277
drop_event() (*méthode calibre.gui2.actions.InterfaceAction*), 296

E

edit_file() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
eject() (*méthode calibre.devices.interface.DevicePlugin*), 284
eject() (*méthode calibre.devices.usbms.device.Device*), 292
embed_metadata() (*méthode calibre.db.cache.Cache*), 369
encoding (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
exists() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
expire_old_trash() (*méthode calibre.db.cache.Cache*), 369
export_note() (*méthode calibre.db.cache.Cache*), 369
extra_css (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
extract_readable_article() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43

F

fast_field_for() (*méthode calibre.db.cache.Cache*), 369
feeds (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
field_for() (*méthode calibre.db.cache.Cache*), 369
field_ids_for() (*méthode calibre.db.cache.Cache*), 369
field_supports_notes() (*méthode calibre.db.cache.Cache*), 369
file_type (*attribut calibre.customize.conversion.OutputFormatPlugin*), 280
file_types (*attribut calibre.customize.CatalogPlugin*), 273
file_types (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
file_types (*attribut calibre.customize.FileTypePlugin*), 270
file_types (*attribut calibre.customize.MetadataReaderPlugin*), 272
file_types (*attribut calibre.customize.MetadataWriterPlugin*), 273
filename_callback() (*méthode calibre.devices.usbms.device.Device*), 293
filesize() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
FileTypePlugin (*classe dans calibre.customize*), 270
filter_css() (*dans le module calibre.ebooks.oeb.polish.css*), 384
filter_regexps (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 48
find_identical_books() (*méthode calibre.db.cache.Cache*), 369
fix_all_html() (*dans le module calibre.ebooks.oeb.polish.pretty*), 381
fix_html() (*dans le module calibre.ebooks.oeb.polish.pretty*), 381
for_viewer (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
format() (*méthode calibre.db.cache.Cache*), 370
format_abspath() (*méthode calibre.db.cache.Cache*), 370
format_added (*attribut calibre.db.cache.Cache.EventType*), 366
format_field() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
format_hash() (*méthode calibre.db.cache.Cache*), 370
format_metadata() (*méthode calibre.db.cache.Cache*), 370
FORMATS (*attribut calibre.devices.interface.DevicePlugin*), 281
FORMATS (*attribut calibre.devices.usbms.driver.USBMS*), 293
formats() (*méthode calibre.db.cache.Cache*), 370
formats_removed (*attribut calibre.db.cache.Cache.EventType*), 366
free_space() (*méthode calibre.devices.interface.DevicePlugin*), 285
free_space() (*méthode calibre.devices.usbms.device.Device*), 292
from_files() (*dans le module calibre.ebooks.oeb.polish.toc*), 384
from_links() (*dans le module calibre.ebooks.oeb.polish.toc*), 384
from_xpaths() (*dans le module calibre.ebooks.oeb.polish.toc*), 384

G

generate_item() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
genesis() (*méthode calibre.gui2.actions.InterfaceAction*), 297
genesis() (*méthode calibre.gui2.preferences.ConfigWidgetInterface*), 300
get_all_items_that_have_notes() (*méthode calibre.db.cache.Cache*), 370
get_all_link_maps_for_book() (*méthode calibre.db.cache.Cache*), 370
get_all_standard_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222

get_all_user_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
get_annotations() (*méthode calibre.devices.usbms.device.Device*), 293
get_article_url() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
get_author_tokens() (*méthode calibre.ebooks.metadata.sources.base.Source*), 275
get_book_url() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
get_book_url_name() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
get_book_urls() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
get_browser() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 43
get_cached_cover_url() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
get_categories() (*méthode calibre.db.cache.Cache*), 371
get_collections() (*méthode calibre.devices.interface.BookList*), 289
get_cover_url() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_custom_book_data() (*méthode calibre.db.cache.Cache*), 371
get_device_information() (*méthode calibre.devices.interface.DevicePlugin*), 284
get_device_information() (*méthode calibre.devices.usbms.driver.USBMS*), 293
get_device_uid() (*méthode calibre.devices.interface.DevicePlugin*), 287
get_driveinfo() (*méthode calibre.devices.interface.DevicePlugin*), 284
get_extra_css() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_feeds() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_file() (*méthode calibre.devices.interface.DevicePlugin*), 286
get_file_path_for_processing() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
get_id_map() (*méthode calibre.db.cache.Cache*), 371
get_identifiers() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
get_ids_for_custom_book_data() (*méthode calibre.db.cache.Cache*), 371
get_images() (*méthode calibre.customize.conversion.InputFormatPlugin*), 279
get_item_id() (*méthode calibre.db.cache.Cache*), 371
get_item_ids() (*méthode calibre.db.cache.Cache*), 371
get_item_name() (*méthode calibre.db.cache.Cache*), 371
get_item_name_map() (*méthode calibre.db.cache.Cache*), 371
get_link_map() (*méthode calibre.db.cache.Cache*), 371
get_masthead_title() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_masthead_url() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_metadata() (*méthode calibre.customize.MetadataReaderPlugin*), 273
get_metadata() (*méthode calibre.db.cache.Cache*), 371
get_next_series_num_for() (*méthode calibre.db.cache.Cache*), 371
get_notes_resource() (*méthode calibre.db.cache.Cache*), 372
get_obfuscated_article() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_open_popup_message() (*méthode de la classe calibre.devices.interface.DevicePlugin*), 282
get_option() (*méthode calibre.devices.interface.DevicePlugin*), 288
get_proxy_metadata() (*méthode calibre.db.cache.Cache*), 372
get_recommended_folders() (*dans le module calibre.ebooks.oeb.polish.replace*), 381
get_standard_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
get_title_tokens() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
get_url_specific_delay() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 44
get_usage_count_by_id() (*méthode*

calibre.db.cache.Cache), 372
get_user_blacklisted_devices() (*méthode calibre.devices.interface.DevicePlugin*), 287
get_user_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
gui (*propriété calibre.gui2.tweak_book.plugin.Tool*), 385
gui_category (*attribut calibre.customize.PreferencesPlugin*), 299
gui_configuration_widget() (*méthode calibre.customize.conversion.InputFormatPlugin*), 279
gui_configuration_widget() (*méthode calibre.customize.conversion.OutputFormatPlugin*), 281
gui_layout_complete() (*méthode calibre.gui2.actions.InterfaceAction*), 298
gui_name (*attribut calibre.customize.PreferencesPlugin*), 299
guide_type_map (*propriété calibre.ebooks.oeb.polish.container.Container*), 378

H

handle_gzip (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
has_book() (*méthode calibre.db.cache.Cache*), 372
has_format() (*méthode calibre.db.cache.Cache*), 372
has_html_comments (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
has_id() (*méthode calibre.db.cache.Cache*), 372
has_name() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
href_to_name() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
HTML, **393**

I

icon (*attribut calibre.customize.PreferencesPlugin*), 299
icon (*attribut calibre.devices.interface.DevicePlugin*), 282
id_from_url() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
identify() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
identify_results_keygen() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
ignore_connected_device() (*méthode calibre.devices.interface.DevicePlugin*), 287
ignore_duplicate_articles (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
ignore_ssl_errors (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
image_url_processor() (*méthode de la classe calibre.web.feeds.news.BasicNewsRecipe*), 45
import_note() (*méthode calibre.db.cache.Cache*), 372
index_to_soup() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 45
indexing_progress_changed (*attribut calibre.db.cache.Cache.EventType*), 367
init() (*méthode calibre.db.cache.Cache*), 372
initial_tab_changed() (*méthode calibre.gui2.preferences.ConfigWidgetInterface*), 300
initialization_complete() (*méthode calibre.gui2.actions.InterfaceAction*), 298
initialize() (*méthode calibre.customize.CatalogPlugin*), 274
initialize() (*méthode calibre.customize.Plugin*), 269
initialize() (*méthode calibre.gui2.preferences.ConfigWidgetBase*), 301
initialize() (*méthode calibre.gui2.preferences.ConfigWidgetInterface*), 300
InputFormatPlugin (*classe dans calibre.customize.conversion*), 278
insert_into_xml() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
installation_type (*attribut calibre.customize.Plugin*), 269
InterfaceAction (*classe dans calibre.gui2.actions*), 295
InterfaceActionBase (*classe dans calibre.customize*), 298
InternalMetadataCompareKeyGen (*classe dans calibre.ebooks.metadata.sources.base*), 277
is_configured() (*méthode calibre.ebooks.metadata.sources.base.Source*), 275
is_dir (*attribut calibre.ebooks.oeb.polish.container.Container*), 378
is_dynamically_controllable() (*méthode calibre.devices.interface.DevicePlugin*), 288
is_image_collection (*attribut calibre.customize.conversion.InputFormatPlugin*), 278

is_link_wanted() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 45
is_null() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
is_running() (*méthode calibre.devices.interface.DevicePlugin*), 288
is_usb_connected() (*méthode calibre.devices.interface.DevicePlugin*), 282
items_removed (*attribut calibre.db.cache.Cache.EventType*), 367
items_renamed (*attribut calibre.db.cache.Cache.EventType*), 367
items_with_notes_in_book() (*méthode calibre.db.cache.Cache*), 372
iterlinks() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378

K

keep_only_tags (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49

L

language (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
library_about_to_change() (*méthode calibre.gui2.actions.InterfaceAction*), 298
library_changed() (*méthode calibre.gui2.actions.InterfaceAction*), 298
link_for() (*méthode calibre.db.cache.Cache*), 372
list_extra_files() (*méthode calibre.db.cache.Cache*), 372
load_actual_plugin() (*méthode calibre.customize.InterfaceActionBase*), 298
load_resources() (*méthode calibre.customize.Plugin*), 269
load_resources() (*méthode calibre.gui2.actions.InterfaceAction*), 297
location_selected() (*méthode calibre.gui2.actions.InterfaceAction*), 298
LRF, **393**

M

make_name_unique() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
MANAGES_DEVICE_PRESENCE (*attribut calibre.devices.interface.DevicePlugin*), 282
manifest_has_name() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
manifest_id_map (*propriété calibre.ebooks.oeb.polish.container.Container*), 378
manifest_items_of_type() (*méthode calibre.ebooks.oeb.polish.container.Container*), 378
manifest_items_with_property() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
manifest_type_map (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
mark_as_cover() (*dans le module calibre.ebooks.oeb.polish.cover*), 383
mark_as_titlepage() (*dans le module calibre.ebooks.oeb.polish.cover*), 383
masthead_url (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
match_regexps (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
max_articles_per_feed (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
MAX_PATH_LEN (*attribut calibre.devices.usbms.device.Device*), 291
merge() (*dans le module calibre.ebooks.oeb.polish.split*), 382
merge_extra_files() (*méthode calibre.db.cache.Cache*), 373
Metadata (*classe dans calibre.ebooks.metadata.book.base*), 221
metadata_changed (*attribut calibre.db.cache.Cache.EventType*), 367
metadata_for_field() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
MetadataReaderPlugin (*classe dans calibre.customize*), 272
MetadataWriterPlugin (*classe dans calibre.customize*), 273
mi (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
minimum_calibre_version (*attribut calibre.customize.Plugin*), 269
module
    calibre.customize, 268
    calibre.customize.conversion, 278
    calibre.db.cache, 366
    calibre.devices.interface, 281
    calibre.ebooks.metadata.book.base, 221
    calibre.ebooks.metadata.sources.base,

274
calibre.ebooks.oeb.polish.container, 376
calibre.ebooks.oeb.polish.cover, 383
calibre.ebooks.oeb.polish.css, 383
calibre.ebooks.oeb.polish.jacket, 382
calibre.ebooks.oeb.polish.pretty, 381
calibre.ebooks.oeb.polish.replace, 381
calibre.ebooks.oeb.polish.split, 382
calibre.ebooks.oeb.polish.toc, 384
calibre.gui2.tweak_book.boss, 386
calibre.utils.formatter_functions, 199
calibre.web.feeds.news, 42
move_book_from_trash() (*méthode calibre.db.cache.Cache*), 373
move_format_from_trash() (*méthode calibre.db.cache.Cache*), 373
multisort() (*méthode calibre.db.cache.Cache*), 373
multisplit() (*dans le module calibre.ebooks.oeb.polish.split*), 382

N

name (*attribut calibre.customize.Plugin*), 268
name (*attribut calibre.gui2.actions.InterfaceAction*), 296
name (*attribut calibre.gui2.tweak_book.plugin.Tool*), 384
name(), **169**
name_order (*attribut calibre.customize.PreferencesPlugin*), 299
name_to_abspath() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
name_to_href() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
names_that_must_not_be_changed (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
names_that_must_not_be_removed (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
names_that_need_not_be_manifested (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
needs_subscription (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 49
NEWS_IN_FOLDER (*attribut calibre.devices.usbms.device.Device*), 291
no_stylesheets (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
normalize_path() (*méthode de la classe calibre.devices.usbms.driver.USBMS*), 295
notes_data_for() (*méthode calibre.db.cache.Cache*), 373
notes_for() (*méthode calibre.db.cache.Cache*), 373
notes_resources_used_by() (*méthode calibre.db.cache.Cache*), 373
NUKE_COMMENTS (*attribut calibre.devices.interface.DevicePlugin*), 282

O

oldest_article (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
on_import (*attribut calibre.customize.FileTypePlugin*), 270
on_postconvert (*attribut calibre.customize.FileTypePlugin*), 271
on_postdelete (*attribut calibre.customize.FileTypePlugin*), 271
on_postimport (*attribut calibre.customize.FileTypePlugin*), 271
on_postprocess (*attribut calibre.customize.FileTypePlugin*), 271
on_preprocess (*attribut calibre.customize.FileTypePlugin*), 271
open() (*méthode calibre.devices.interface.DevicePlugin*), 284
open() (*méthode calibre.devices.usbms.device.Device*), 292
open() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
open_book() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
OPEN_FEEDBACK_MESSAGE (*attribut calibre.devices.interface.DevicePlugin*), 282
opf (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
opf_get_or_create() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
opf_version (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
opf_version_parsed (*propriété calibre.ebooks.oeb.polish.container.Container*), 379
opf_xpath() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
option de ligne de commande
--help, 326
--library-path, 326

--password, 326
--timeout, 326
--username, 326
--version, 326
--with-library, 326
-h, 326
option de ligne de commande calibre
--detach, 316
--help, 316
--ignore-plugins, 316
--no-update-check, 316
--shutdown-running-calibre, 316
--start-in-tray, 316
--verbose, 316
--version, 316
--with-library, 316
-h, 316
-s, 316
-v, 316
option de ligne de commande calibre-customize
--add-plugin, 316
--build-plugin, 316
--customize-plugin, 316
--disable-plugin, 316
--enable-plugin, 316
--help, 317
--list-plugins, 317
--remove-plugin, 317
--version, 317
-a, 316
-b, 316
-h, 317
-l, 317
-r, 317
option de ligne de commande calibre-debug
--add-simple-plugin, 317
--command, 317
--debug-device-driver, 317
--default-programs, 317
--diff, 317
--edit-book, 318
--exec-file, 318
--explode-book, 318
--export-all-calibre-data, 318
--fix-multiprocessing, 318
--gui, 318
--gui-debug, 318
--help, 318
--implode-book, 318
--import-calibre-data, 318
--inspect-mobi, 318
--paths, 318
--run-plugin, 318
--run-test, 318
--run-without-debug, 319
--shutdown-running-calibre, 319
--subset-font, 319
--test-build, 319
--version, 319
--viewer, 319
-c, 317
-d, 317
-e, 318
-f, 319
-g, 318
-h, 318
-i, 318
-m, 318
-r, 318
-s, 319
-t, 318
-w, 319
-x, 318
option de ligne de commande calibre-server
--access-log, 319
--ajax-timeout, 319
--auth-mode, 319
--auto-reload, 319
--ban-after, 319
--ban-for, 320
--book-list-mode, 320
--compress-min-size, 320
--custom-list-template, 320
--daemonize, 320
--disable-allow-socket-preallocation, 320
--disable-auth, 320
--disable-fallback-to-detected-interface, 320
--disable-local-write, 320
--disable-log-not-found, 320
--disable-use-bonjour, 321
--disable-use-sendfile, 321
--displayed-fields, 320
--enable-allow-socket-preallocation, 320
--enable-auth, 320
--enable-fallback-to-detected-interface, 320
--enable-local-write, 320
--enable-log-not-found, 320
--enable-use-bonjour, 321
--enable-use-sendfile, 321
--help, 321
--ignored-fields, 321

--listen-on, 321
--log, 321
--manage-users, 321
--max-header-line-size, 321
--max-job-time, 321
--max-jobs, 321
--max-log-size, 321
--max-opds-items, 321
--max-opds-ungrouped-items, 322
--max-request-body-size, 322
--num-per-page, 322
--pidfile, 322
--port, 322
--search-the-net-urls, 322
--shutdown-timeout, 322
--ssl-certfile, 322
--ssl-keyfile, 322
--timeout, 322
--trusted-ips, 322
--url-prefix, 322
--userdb, 322
--version, 322
--worker-count, 323
-h, 321
option de ligne de commande calibre-smtp
--attachment, 324
--cafile, 324
--dont-verify-server-certificate, 324
--encryption-method, 324
--fork, 323
--help, 323
--localhost, 323
--outbox, 323
--password, 324
--port, 324
--relay, 324
--subject, 324
--timeout, 323
--username, 324
--verbose, 323
--version, 323
-a, 324
-e, 324
-f, 323
-h, 323
-l, 323
-o, 323
-p, 324
-r, 324
-s, 324
-t, 323
-u, 324
-v, 323
option de ligne de commande calibredb-add
-1, 328
-I, 328
-S, 328
-T, 328
--add, 328
--authors, 327
--automerge, 327
--cover, 327
--duplicates, 328
--empty, 328
--identifier, 328
--ignore, 328
--isbn, 328
--languages, 328
--one-book-per-directory, 328
--recurse, 328
--series, 328
--series-index, 328
--tags, 328
--title, 328
-a, 327
-c, 327
-d, 328
-e, 328
-i, 328
-l, 328
-m, 327
-r, 328
-s, 328
-t, 328
option de ligne de commande calibredb-add_custom_column
--display, 334
--is-multiple, 334
option de ligne de commande calibredb-add_format
--as-extra-data-file, 329
--dont-replace, 329
option de ligne de commande calibredb-backup_metadata
--all, 336
option de ligne de commande calibredb-catalog
--catalog-title, 332
--cross-reference-authors, 332
--debug-pipeline, 332
--exclude-genre, 332
--exclusion-rules, 332
--generate-authors, 332
--generate-descriptions, 332
--generate-genres, 332

--generate-recently-added, 332
--generate-series, 332
--generate-titles, 332
--genre-source-field, 332
--header-note-source-field, 333
--ids, 331
--merge-comments-rule, 333
--output-profile, 333
--prefix-rules, 333
--preset, 333
--search, 331
--thumb-width, 333
--use-existing-cover, 333
--verbose, 331
-i, 331
-s, 331
-v, 331
option de ligne de commande calibredb-check_library
--csv, 335
--ignore_extensions, 335
--ignore_names, 335
--report, 335
--vacuum-fts-db, 335
-c, 335
-e, 335
-n, 335
-r, 335
option de ligne de commande calibredb-custom_columns
--details, 334
-d, 334
option de ligne de commande calibredb-embed_metadata
--only-formats, 337
-f, 337
option de ligne de commande calibredb-export
--all, 330
--dont-asciiize, 330
--dont-save-cover, 330
--dont-save-extra-files, 330
--dont-update-metadata, 330
--dont-write-opf, 331
--formats, 331
--progress, 331
--replace-whitespace, 331
--single-dir, 331
--template, 331
--timefmt, 331
--to-dir, 331
--to-lowercase, 331
option de ligne de commande calibredb-fts_index
--indexing-speed, 338
--wait-for-completion, 338
option de ligne de commande calibredb-fts_search
--do-not-match-on-related-words, 338
--include-snippets, 338
--indexing-threshold, 338
--match-end-marker, 338
--match-start-marker, 338
--output-format, 338
--restrict-to, 338
option de ligne de commande calibredb-list
--ascending, 326
--fields, 326
--for-machine, 326
--limit, 327
--line-width, 327
--prefix, 327
--search, 327
--separator, 327
--sort-by, 327
--template, 327
--template_file, 327
--template_heading, 327
-f, 326
-s, 327
-t, 327
-w, 327
option de ligne de commande calibredb-list_categories
--categories, 336
--csv, 336
--dialect, 336
--item_count, 336
--width, 336
-c, 336
-i, 336
-r, 336
-w, 336
option de ligne de commande calibredb-remove
--permanent, 329
option de ligne de commande calibredb-remove_custom_column
--force, 334
-f, 334
option de ligne de commande calibredb-restore_database
--really-do-it, 335
-r, 335
option de ligne de commande calibredb-search
--limit, 337

-l, 337
option de ligne de commande calibredb-set_custom
--append, 335
-a, 335
option de ligne de commande calibredb-set_metadata
--field, 330
--list-fields, 330
-f, 330
-l, 330
option de ligne de commande calibredb-show_metadata
--as-opf, 329
option de ligne de commande ebook-convert
--asciiize, 341
--author-sort, 347
--authors, 347
--base-font-size, 341
--book-producer, 347
--change-justification, 341
--chapter, 345
--chapter-mark, 345
--comments, 347
--cover, 347
--debug-pipeline, 348
--disable-dehyphenate, 344
--disable-delete-blank-paragraphs, 344
--disable-fix-indents, 344
--disable-font-rescaling, 342
--disable-format-scene-breaks, 344
--disable-italicize-common-cases, 344
--disable-markup-chapter-headings, 344
--disable-remove-fake-margins, 345
--disable-renumber-headings, 344
--disable-unwrap-lines, 344
--dont-split-on-page-breaks, 340
--duplicate-links-in-toc, 346
--embed-all-fonts, 342
--embed-font-family, 342
--enable-heuristics, 344
--epub-flatten, 340
--epub-inline-toc, 340
--epub-max-image-size, 340
--epub-toc-at-end, 340
--epub-version, 340
--expand-css, 342
--extra-css, 342
--extract-to, 341
--filter-css, 342
--flow-size, 341
--font-size-mapping, 342
--from-opf, 347
--help, 339
--html-unwrap-factor, 344
--input-encoding, 340
--input-profile, 339
--insert-blank-line, 342
--insert-blank-line-size, 342
--insert-metadata, 345
--isbn, 347
--keep-ligatures, 342
--language, 347
--level1-toc, 346
--level2-toc, 346
--level3-toc, 346
--line-height, 342
--linearize-tables, 342
--list-recipes, 339
--margin-bottom, 343
--margin-left, 343
--margin-right, 343
--margin-top, 343
--max-toc-links, 346
--minimum-line-height, 343
--no-chapters-in-toc, 346
--no-default-epub-cover, 341
--no-svg-cover, 341
--output-profile, 340
--page-breaks-before, 346
--prefer-metadata-cover, 346
--preserve-cover-aspect-ratio, 341
--pretty-print, 341
--pubdate, 347
--publisher, 347
--rating, 347
--read-metadata-from-opf, 347
--remove-first-image, 346
--remove-paragraph-spacing, 343
--remove-paragraph-spacing-indent-size, 343
--replace-scene-breaks, 344
--search-replace, 345
--series, 347
--series-index, 347
--smarten-punctuation, 343
--sr1-replace, 345
--sr1-search, 345
--sr2-replace, 345
--sr2-search, 345
--sr3-replace, 345
--sr3-search, 345
--start-reading-at, 346
--subset-embedded-fonts, 343

--tags, 347
--timestamp, 348
--title, 348
--title-sort, 348
--toc-filter, 346
--toc-threshold, 347
--toc-title, 341
--transform-css-rules, 343
--transform-html-rules, 344
--unsmarten-punctuation, 344
--use-auto-toc, 347
--verbose, 348
--version, 340
-d, 348
-h, 339
-m, 347
-v, 348
option de ligne de commande ebook-edit
--detach, 348
--help, 348
--select-text, 348
--version, 348
-h, 348
option de ligne de commande ebook-meta
--author-sort, 349
--authors, 349
--book-producer, 349
--category, 349
--comments, 349
--cover, 349
--date, 349
--from-opf, 349
--get-cover, 349
--help, 349
--identifier, 349
--index, 350
--isbn, 350
--language, 350
--lrf-bookid, 350
--publisher, 350
--rating, 350
--series, 350
--tags, 350
--title, 350
--title-sort, 350
--to-opf, 350
--version, 350
-a, 349
-c, 349
-d, 349
-h, 349
-i, 350
-k, 349
-l, 350
-p, 350
-r, 350
-s, 350
-t, 350
option de ligne de commande ebook-polish
-H, 351
-U, 352
--add-soft-hyphens, 351
--compress-images, 351
--cover, 351
--embed-fonts, 351
--help, 351
--jacket, 351
--opf, 351
--remove-jacket, 351
--remove-soft-hyphens, 351
--remove-unused-css, 351
--smarten-punctuation, 351
--subset-fonts, 351
--upgrade-book, 352
--verbose, 352
--version, 352
-c, 351
-e, 351
-f, 351
-h, 351
-i, 351
-j, 351
-o, 351
-p, 351
-u, 351
option de ligne de commande ebook-viewer
--continue, 352
--detach, 352
--force-reload, 352
--full-screen, 352
--fullscreen, 352
--help, 352
--new-instance, 352
--open-at, 352
--raise-window, 352
--version, 352
-f, 352
-h, 352
option de ligne de commande fetch-ebook-metadata
-I, 353
--allowed-plugin, 353
--authors, 353
--cover, 353
--help, 353
--identifier, 353

--isbn, 353
--opf, 353
--timeout, 353
--title, 353
--verbose, 353
--version, 353
-a, 353
-c, 353
-d, 353
-h, 353
-i, 353
-o, 353
-p, 353
-t, 353
-v, 353
option de ligne de commande lrf2lrs
--dont-output-resources, 354
--help, 354
--output, 354
--verbose, 354
--version, 354
-h, 354
-o, 354
option de ligne de commande lrfviewer
--disable-hyphenation, 354
--help, 354
--profile, 354
--verbose, 354
--version, 354
--visual-debug, 354
--white-background, 355
-h, 354
option de ligne de commande lrs2lrf
--help, 355
--lrs, 355
--output, 355
--verbose, 355
--version, 355
-h, 355
-o, 355
option de ligne de commande web2disk
--base-dir, 355
--delay, 355
--dont-download-stylesheets, 355
--encoding, 355
--filter-regexp, 356
--help, 356
--match-regexp, 356
--max-files, 356
--max-recursions, 356
--timeout, 356
--verbose, 356
--version, 356
-d, 355
-h, 356
-n, 356
-r, 356
-t, 356
options (*attribut calibre.customize.conversion.InputFormatPlugin*), 279
options (*attribut calibre.customize.conversion.OutputFormatPlugin*), 280
options (*attribut calibre.ebooks.metadata.sources.base.Source*), 275
OSX_MAIN_MEM_VOL_PAT (*attribut calibre.devices.usbms.device.Device*), 291
output_encoding (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
OutputFormatPlugin (*classe dans calibre.customize.conversion*), 279

P

parse_feeds() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 45
parse_index() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 45
parsed() (*méthode calibre.ebooks.oeb.polish.container.Container*), 379
path_sep (*attribut calibre.devices.interface.DevicePlugin*), 282
Plugin (*classe dans calibre.customize*), 268
populate_article_metadata() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 45
popup_type (*attribut calibre.gui2.actions.InterfaceAction*), 296
post_yank_cleanup() (*méthode calibre.devices.interface.DevicePlugin*), 284
post_yank_cleanup() (*méthode calibre.devices.usbms.device.Device*), 292
postadd() (*méthode calibre.customize.FileTypePlugin*), 272
postconvert() (*méthode calibre.customize.FileTypePlugin*), 271
postdelete() (*méthode calibre.customize.FileTypePlugin*), 272
postimport() (*méthode calibre.customize.FileTypePlugin*), 271
postprocess_book() (*méthode calibre.customize.conversion.InputFormatPlugin*), 279
postprocess_book() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 46
postprocess_html() (*méthode calibre.web.feeds.news.BasicNewsRecipe*),

46
pref() (*méthode calibre.db.cache.Cache*), 373
prefer_results_with_isbn (*attribut calibre.ebooks.metadata.sources.base.Source*), 275
PreferencesPlugin (*classe dans calibre.customize*), 299
prepare_addable_books() (*méthode calibre.devices.interface.DevicePlugin*), 287
preprocess_html() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 46
preprocess_image() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 46
preprocess_raw_html() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 46
preprocess_regexps (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
pretty_all() (*dans le module calibre.ebooks.oeb.polish.pretty*), 382
pretty_css() (*dans le module calibre.ebooks.oeb.polish.pretty*), 381
pretty_html() (*dans le module calibre.ebooks.oeb.polish.pretty*), 381
pretty_xml() (*dans le module calibre.ebooks.oeb.polish.pretty*), 382
print_version() (*méthode de la classe calibre.web.feeds.news.BasicNewsRecipe*), 46
priority (*attribut calibre.customize.Plugin*), 269
priority (*attribut calibre.gui2.actions.InterfaceAction*), 296
PRODUCT_ID (*attribut calibre.devices.interface.DevicePlugin*), 281
PRODUCT_ID (*attribut calibre.devices.usbms.device.Device*), 290
publication_date() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 46
publication_type (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50

R

raw_data() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
re:test(), **169**
read_backup() (*méthode calibre.db.cache.Cache*), 373
Recette, **393**
recipe_disabled (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
recommendations (*attribut calibre.customize.conversion.InputFormatPlugin*), 279
recommendations (*attribut calibre.customize.conversion.OutputFormatPlugin*), 280
recursions (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
refresh_gui() (*méthode calibre.gui2.preferences.ConfigWidgetInterface*), 300
regexp, **394**
register() (*méthode calibre.gui2.preferences.ConfigWidgetBase*), 301
register_shortcut() (*méthode calibre.gui2.tweak_book.plugin.Tool*), 385
relpath() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
remove_attributes (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
remove_book() (*méthode calibre.devices.interface.BookList*), 289
remove_books() (*méthode calibre.db.cache.Cache*), 373
remove_books_from_metadata() (*méthode calibre.devices.usbms.driver.USBMS*), 295
remove_books_from_metadata() (*méthode de la classe calibre.devices.interface.DevicePlugin*), 286
remove_empty_feeds (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
remove_formats() (*méthode calibre.db.cache.Cache*), 373
remove_from_spine() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
remove_from_xml() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
remove_item() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
remove_items() (*méthode calibre.db.cache.Cache*), 373
remove_jacket() (*dans le module calibre.ebooks.oeb.polish.jacket*), 382
remove_javascript (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50

remove_stale_user_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
remove_tags (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 50
remove_tags_after (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
remove_tags_before (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
remove_unused_css() (*dans le module calibre.ebooks.oeb.polish.css*), 383
rename() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
rename_extra_files() (*méthode calibre.db.cache.Cache*), 374
rename_files() (*dans le module calibre.ebooks.oeb.polish.replace*), 381
rename_items() (*méthode calibre.db.cache.Cache*), 374
replace() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
replace_links() (*dans le module calibre.ebooks.oeb.polish.replace*), 381
replace_links() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
requires_version (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
reset() (*méthode calibre.devices.interface.DevicePlugin*), 283
reset() (*méthode calibre.devices.usbms.device.Device*), 291
resolve_internal_links (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
restart_critical (*attribut calibre.gui2.preferences.ConfigWidgetBase*), 300
restart_critical (*attribut calibre.gui2.preferences.ConfigWidgetInterface*), 300
restore_book() (*méthode calibre.db.cache.Cache*), 374
restore_defaults() (*méthode calibre.gui2.preferences.ConfigWidgetBase*), 301
restore_defaults() (*méthode calibre.gui2.preferences.ConfigWidgetInterface*), 300
restore_defaults_desc (*attribut calibre.gui2.preferences.ConfigWidgetInterface*), 300
restore_original_format() (*méthode calibre.db.cache.Cache*), 374
reverse_article_order (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
rewind_savepoint() (*méthode calibre.gui2.tweak_book.boss.Boss*), 386
RSS, **393**
run() (*méthode calibre.customize.CatalogPlugin*), 274
run() (*méthode calibre.customize.FileTypePlugin*), 271

S

safe_read_lock (*propriété calibre.db.cache.Cache*), 374
sanitize_callback() (*méthode calibre.devices.usbms.device.Device*), 293
sanitize_path_components() (*méthode calibre.devices.usbms.device.Device*), 293
save_book() (*méthode calibre.gui2.tweak_book.boss.Boss*), 387
save_original_format() (*méthode calibre.db.cache.Cache*), 374
save_settings() (*méthode calibre.customize.Plugin*), 269
save_settings() (*méthode calibre.ebooks.metadata.sources.base.Source*), 275
save_settings() (*méthode de la classe calibre.devices.interface.DevicePlugin*), 286
scale_news_images (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
scale_news_images_to_device (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
search() (*méthode calibre.db.cache.Cache*), 374
search_notes() (*méthode calibre.db.cache.Cache*), 374
serialize_item() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
set_all_user_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
set_conversion_options() (*méthode calibre.db.cache.Cache*), 375
set_cover() (*dans le module calibre.ebooks.oeb.polish.cover*), 383
set_cover() (*méthode calibre.db.cache.Cache*), 375
set_driveinfo_name() (*méthode calibre.devices.interface.DevicePlugin*), 287

set_driveinfo_name() (*méthode calibre.devices.usbms.driver.USBMS*), 293
set_field() (*méthode calibre.db.cache.Cache*), 375
set_identifier() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
set_identifiers() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
set_library_info() (*méthode calibre.devices.interface.DevicePlugin*), 287
set_link_map() (*méthode calibre.db.cache.Cache*), 375
set_metadata() (*méthode calibre.customize.MetadataWriterPlugin*), 273
set_metadata() (*méthode calibre.db.cache.Cache*), 375
set_modified() (*méthode calibre.gui2.tweak_book.boss.Boss*), 387
set_notes_for() (*méthode calibre.db.cache.Cache*), 375
set_option() (*méthode calibre.devices.interface.DevicePlugin*), 288
set_plugboards() (*méthode calibre.devices.interface.DevicePlugin*), 286
set_pref() (*méthode calibre.db.cache.Cache*), 376
set_progress_reporter() (*méthode calibre.devices.interface.DevicePlugin*), 284
set_progress_reporter() (*méthode calibre.devices.usbms.device.Device*), 291
set_spine() (*méthode calibre.ebooks.oeb.polish.container.Container*), 380
set_user_blacklisted_devices() (*méthode calibre.devices.interface.DevicePlugin*), 287
set_user_metadata() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
settings() (*méthode de la classe calibre.devices.interface.DevicePlugin*), 286
show_current_diff() (*méthode calibre.gui2.tweak_book.boss.Boss*), 387
show_editor() (*méthode calibre.gui2.tweak_book.boss.Boss*), 387
shutdown() (*méthode calibre.devices.interface.DevicePlugin*), 287
shutting_down() (*méthode calibre.gui2.actions.InterfaceAction*), 298
simultaneous_downloads (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
skip_ad_pages() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 46
SLOW_DRIVEINFO (*attribut calibre.devices.interface.DevicePlugin*), 282
smart_update() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
sort_index_by() (*méthode calibre.web.feeds.news.BasicNewsRecipe*), 47
Source (*classe dans calibre.ebooks.metadata.sources.base*), 274
specialize() (*méthode calibre.customize.conversion.InputFormatPlugin*), 279
specialize_css_for_output() (*méthode calibre.customize.conversion.OutputFormatPlugin*), 280
specialize_global_preferences() (*méthode calibre.devices.interface.DevicePlugin*), 287
specialize_options() (*méthode calibre.customize.conversion.OutputFormatPlugin*), 280
spine_items (*propriété calibre.ebooks.oeb.polish.container.Container*), 380
spine_iter (*propriété calibre.ebooks.oeb.polish.container.Container*), 381
spine_names (*propriété calibre.ebooks.oeb.polish.container.Container*), 381
split() (*dans le module calibre.ebooks.oeb.polish.split*), 382
split_if_is_multiple_composite() (*méthode calibre.db.cache.Cache*), 376
split_jobs() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
standard_field_keys() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 221
STANDARD_METADATA_FIELDS (*dans le module calibre.ebooks.metadata.book.base*), 222
start_plugin() (*méthode calibre.devices.interface.DevicePlugin*), 288
startup() (*méthode calibre.devices.interface.DevicePlugin*), 287
stop_plugin() (*méthode calibre.devices.interface.DevicePlugin*), 288
summary_length (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
supported_platforms (*attribut calibre.customize.conversion.InputFormatPlugin*), 278

supported_platforms (*attribut calibre.customize.conversion.OutputFormatPlugin*), 280
supported_platforms (*attribut calibre.customize.InterfaceActionBase*), 298
supported_platforms (*attribut calibre.customize.MetadataReaderPlugin*), 272
supported_platforms (*attribut calibre.customize.MetadataWriterPlugin*), 273
supported_platforms (*attribut calibre.customize.Plugin*), 268
supported_platforms (*attribut calibre.customize.PreferencesPlugin*), 299
supported_platforms (*attribut calibre.devices.usbms.driver.USBMS*), 293
supported_platforms (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
supports_collections() (*méthode calibre.devices.interface.BookList*), 289
supports_gzip_transfer_encoding (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
supports_restoring_to_defaults (*attribut calibre.gui2.preferences.ConfigWidgetBase*), 300
supports_restoring_to_defaults (*attribut calibre.gui2.preferences.ConfigWidgetInterface*), 300
sync_booklists() (*méthode calibre.devices.interface.DevicePlugin*), 286
sync_booklists() (*méthode calibre.devices.usbms.driver.USBMS*), 295
sync_preview_to_editor() (*méthode calibre.gui2.tweak_book.boss.Boss*), 387
synchronize_with_db() (*méthode calibre.devices.interface.DevicePlugin*), 288

T

tag_browser_context_action() (*méthode calibre.gui2.actions.InterfaceAction*), 298
tag_to_string() (*méthode de la classe calibre.web.feeds.news.BasicNewsRecipe*), 47
tags_older_than() (*méthode calibre.db.cache.Cache*), 376
template_css (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
template_to_attribute() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
temporary_file() (*méthode calibre.customize.Plugin*), 270
test_fields() (*méthode calibre.ebooks.metadata.sources.base.Source*), 276
THUMBNAIL_COMPRESSION_QUALITY (*attribut calibre.devices.interface.DevicePlugin*), 281
THUMBNAIL_HEIGHT (*attribut calibre.devices.interface.DevicePlugin*), 281
timefmt (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 51
timeout (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 52
title (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 52
to_html() (*méthode calibre.ebooks.metadata.book.base.Metadata*), 222
Tool (*classe dans calibre.gui2.tweak_book.plugin*), 384
toolbar_button_popup_mode (*attribut calibre.gui2.tweak_book.plugin.Tool*), 385
total_space() (*méthode calibre.devices.interface.DevicePlugin*), 285
total_space() (*méthode calibre.devices.usbms.device.Device*), 291
touched_fields (*attribut calibre.ebooks.metadata.sources.base.Source*), 274
type (*attribut calibre.customize.CatalogPlugin*), 273
type (*attribut calibre.customize.conversion.InputFormatPlugin*), 278
type (*attribut calibre.customize.conversion.OutputFormatPlugin*), 279
type (*attribut calibre.customize.FileTypePlugin*), 271
type (*attribut calibre.customize.InterfaceActionBase*), 298
type (*attribut calibre.customize.MetadataReaderPlugin*), 272
type (*attribut calibre.customize.MetadataWriterPlugin*), 273
type (*attribut calibre.customize.Plugin*), 269
type (*attribut calibre.customize.PreferencesPlugin*), 299
type (*attribut calibre.devices.interface.DevicePlugin*), 281
type (*attribut calibre.ebooks.metadata.sources.base.Source*), 274

U

unretire_note_for() (*méthode calibre.db.cache.Cache*), 376
upload_books() (*méthode calibre.devices.interface.DevicePlugin*), 285
upload_books() (*méthode calibre.devices.usbms.driver.USBMS*), 294
upload_cover() (*méthode calibre.devices.usbms.driver.USBMS*), 294
URL, **393**
USBMS (*classe dans calibre.devices.usbms.driver*), 293

use_embedded_content (*attribut calibre.web.feeds.news.BasicNewsRecipe*), 52
user_categories_for_books() (*méthode calibre.db.cache.Cache*), 376
user_feedback_after_callback (*attribut calibre.devices.interface.DevicePlugin*), 282
UserAnnotation (*attribut calibre.devices.interface.DevicePlugin*), 282

V

VENDOR_ID (*attribut calibre.devices.interface.DevicePlugin*), 281
VENDOR_ID (*attribut calibre.devices.usbms.device.Device*), 290
version (*attribut calibre.customize.MetadataReaderPlugin*), 272
version (*attribut calibre.customize.MetadataWriterPlugin*), 273
version (*attribut calibre.customize.Plugin*), 268
VIRTUAL_BOOK_EXTENSION_MESSAGE (*attribut calibre.devices.interface.DevicePlugin*), 282
VIRTUAL_BOOK_EXTENSIONS (*attribut calibre.devices.interface.DevicePlugin*), 282

W

WANTS_UPDATED_THUMBNAILS (*attribut calibre.devices.interface.DevicePlugin*), 281
WINDOWS_CARD_A_MEM (*attribut calibre.devices.usbms.device.Device*), 290
WINDOWS_CARD_B_MEM (*attribut calibre.devices.usbms.device.Device*), 291
WINDOWS_MAIN_MEM (*attribut calibre.devices.usbms.device.Device*), 290
windows_sort_drives() (*méthode calibre.devices.usbms.device.Device*), 292