
novelWriter

Guide de l'utilisateur

Version 2.7rc1

Veronica Berglyd Olsen, Jean-Michel Heras

lundi, 19 mai 2025 at 18:06

1	Présentation générale	3
1.1	Fonctionnalités principales	4
1.2	Captures d'écran	5
2	Premières lectures	7
2.1	Utiliser novelWriter	7
2.2	Organiser vos projets	8
2.3	Writing Aids	8
2.4	Détails supplémentaires et sujets techniques	9
3	Configuration et installation	11
3.1	Installation sous Windows	11
3.2	Installation sous Linux	11
3.3	Installation sur MacOS	13
3.4	Installation à partir de PyPi	13
4	Trucs et astuces	15
4.1	Gestion du projet	15
4.2	Astuces de mise en page	16
4.3	Organiser son texte	16
4.4	Autres outils	17
5	Glossaire	19
6	Comment ça marche	21
6.1	Mise en page et conception de l'interface graphique	21
6.2	Structure du projet	25
6.3	Élaboration d'un manuscrit	26
6.4	Stockage du projet	26
7	Vues du projet	29
7.1	L'arborescence du projet	30
7.2	L'arborescence du roman	33
7.3	Vue d'ensemble du roman	34
8	L'éditeur et l'afficheur	37
8.1	Édition d'un document	37

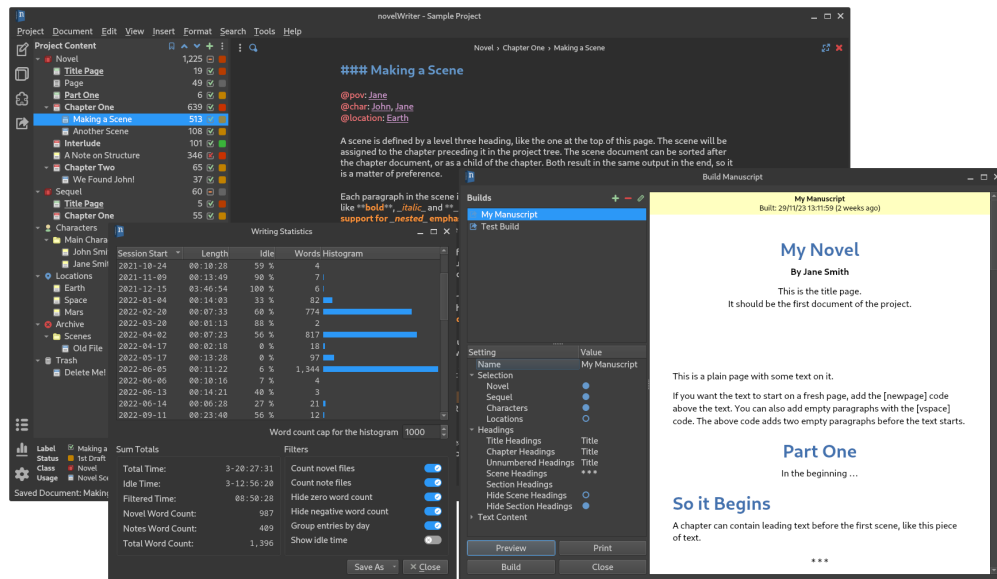
8.2	Visualisation d'un document	39
8.3	Rechercher & Remplacer	40
8.4	Remplacement automatique au fur et à mesure de la saisie	41
8.5	Mise en évidence du dialogue	42
9	Mise en forme du texte	43
9.1	Surlignage syntaxique	43
9.2	En-têtes	44
9.3	Paragraphe du texte	45
9.4	Mise en valeur du texte avec Markdown	46
9.5	Mise en forme avec les shortcodes	47
9.6	Commentaires et synopsis	47
9.7	Notes de bas de page	48
9.8	Texte ignoré	49
9.9	Tags et références	49
9.10	Alignement et indentation des paragraphes	50
9.11	Espace vertical et sauts de page	51
9.12	Insérer le nombre de mots dans le texte	51
10	Raccourcis clavier	53
10.1	Raccourcis de la fenêtre principale	53
10.2	Raccourcis de l'arborescence du projet	54
10.3	Raccourcis de l'éditeur de documents	54
10.4	Raccourcis de la fenêtre de visualisation des documents	57
11	Notes typographiques	59
11.1	Remarques spéciales sur les symboles	59
12	Gestion des projets	61
12.1	Créer un projet	61
12.2	Structure du projet	62
12.3	Documents de projets	65
12.4	Paramètres du projet	66
12.5	Sauvegarde	67
12.6	Statistiques d'écriture	67
13	Structure du roman	69
13.1	Importance des titres	69
14	Tags et références	73
14.1	Métadonnées dans novelWriter	73
14.2	Comment utiliser les tags	74
14.3	Comment utiliser les références	75
15	Compiler le manuscrit	79
15.1	L'outil de compilation de manuscrit	79
15.2	Paramètres de compilation	80
15.3	Création des manuscrits	83
15.4	Impression	84
16	Story Structure Comments	85
16.1	Usage	85
16.2	Output	86

17	Personnalisations	87
17.1	Dictionnaires de vérification orthographique	87
17.2	Syntaxe et thèmes de l'interface graphique	88
18	Modifications du format des projet	91
18.1	Format 1.5 Modifications	91
18.2	Format 1.4 Modifications	92
18.3	Format 1.3 Modifications	92
18.4	Format 1.2 Modifications	92
18.5	Format 1.1 Modifications	93
18.6	Format 1.0 Modifications	93
19	Décompte des mots et des textes	95
19.1	Décompte des mots du texte et statistiques	95
19.2	Décompte du manuscrit	96
20	Emplacement des fichiers	99
20.1	Configuration	99
20.2	Données d'application	99
21	Comment les données sont stockées	101
21.1	Structure du projet	101
21.2	Documents de projet	102
21.3	Métadonnées du projet	102
22	Executer depuis la source	105
22.1	Dépendances	105
22.2	Compiler et installer à partir des sources	106
22.3	Création des fichiers de traduction	106
22.4	Construction du projet d'exemple	107
22.5	Élaboration de la documentation	107
23	Exécution des tests	109
23.1	Dépendances	109
23.2	Test simple	109
23.3	Options avancées	110
	Index	111

Version : 2.7rc1

Mise à jour: lundi, 19 mai 2025 at 18:06

novelWriter est un éditeur de texte brut open source conçu pour écrire des romans assemblés à partir de documents texte individuels. Il utilise une syntaxe de formatage minimale inspirée de Markdown, et ajoute une syntaxe de métadonnées pour les commentaires, les synopsis et les références croisées. Il est conçu pour être un éditeur de texte simple qui permet d'organiser facilement le texte et les notes, en utilisant des fichiers texte lisibles par l'homme comme stockage pour plus de robustesse.



Le stockage du projet est adapté aux logiciels de contrôle de version, ainsi qu'aux outils de synchronisation de fichiers. Tout le texte est sauvegardé sous forme de fichiers de texte brut, et les données de votre projet sous les formats de données standard XML et JSON. Voir [Comment les données sont stockées](#) pour plus de détails.

Tout système d'exploitation disposant de Python 3 et des bibliothèques Qt 6 devrait pouvoir faire fonctionner novelWriter. Il fonctionne correctement sous Linux, Windows et MacOS, et des utilisateurs l'ont également testé sur d'autres plateformes. Voir [Configuration et installation](#) pour plus de détails.

Liens utiles

- Site web : <https://novelwriter.io>
- Documentation: <https://docs.novelwriter.io>
- Public Releases: <https://releases.novelwriter.io>
- Internationalisation: <https://crowdin.com/project/novelwriter>
- Code source : <https://github.com/vkbo/novelWriter>
- Source Releases: <https://github.com/vkbo/novelWriter/releases>
- Issue Tracker: <https://github.com/vkbo/novelWriter/issues>
- Feature Discussions: <https://github.com/vkbo/novelWriter/discussions>
- PyPi Project: <https://pypi.org/project/novelWriter>
- Social Media: <https://fosstodon.org/@novelwriter>

Présentation générale

A la base, novelWriter est un éditeur de texte brut multi-documents. L'idée est de vous permettre d'éditer votre texte sans avoir à vous préoccuper de la mise en forme jusqu'à ce que vous génériez un projet de document ou un manuscrit. Au lieu de cela, vous pouvez vous concentrer sur l'écriture dès le début.

Bien sûr, vous avez probablement besoin d'une certaine mise en forme pour votre texte. Au minimum, vous avez besoin d'accentuation. La plupart des gens sont familiers avec l'ajout d'emphase en utilisant `_underscores_` et `**asterisks**`. Ce standard de formatage vient de [Markdown](#) et est supporté par novelWriter. Il utilise également le formatage Markdown pour définir les titres du document. Si vous avez besoin d'un formatage plus spécialisé, des options de formatage supplémentaires sont disponibles en utilisant un format shortcode. Voir [Mise en forme avec les shortcodes](#) pour plus de détails.

Restrictions

novelWriter est conçu pour l'écriture de fictions, et les fonctions de formatage disponibles sont donc limitées à celles qui sont pertinentes pour cet usage. Il n'est *pas* adapté à la rédaction technique et n'est *pas* un éditeur Markdown complet.

Il n'est pas non plus conçu comme un outil d'organisation de la recherche en vue de l'écriture, et ne dispose donc pas des fonctions de formatage dont vous pourriez avoir besoin à cette fin. La fonction de notes de novelWriter est principalement destinée aux profils de personnages et aux ébauches d'intrigues.

Votre projet de roman dans novelWriter est organisé comme une collection de documents de texte brut séparés au lieu d'un seul grand document. L'idée est de faciliter la réorganisation de la structure de votre projet sans avoir à couper et coller du texte entre les chapitres et les scènes.

Il y a deux types de documents dans votre projet : [Documents du roman](#) sont des documents qui font partie de votre histoire. L'autre type de documents est [Notes de projet](#), qui sont destinés à vos notes sur vos personnages, votre construction du monde, et ainsi de suite.

Vous pouvez à tout moment diviser les documents individuels par leurs titres en plusieurs documents ou fusionner plusieurs documents en un seul. Il est ainsi plus facile d'utiliser des variantes de la méthode [Snowflake](#) pour l'écriture. Vous pouvez commencer par rédiger des documents plus importants axés sur

la structure, comme par exemple un document par acte, et les diviser ensuite sans effort en documents individuels de chapitres ou de scènes.

1.1 Fonctionnalités principales

Voici quelques caractéristiques clés de novelWriter.

Focus sur l'écriture

L'objectif de l'interface utilisateur est de vous permettre de vous concentrer sur l'écriture plutôt que de passer du temps à formater le texte. Le formatage est donc limité à un petit ensemble de balises de formatage pour des choses simples comme l'emphase du texte et l'alignement des paragraphes. Des shortcodes supplémentaires sont disponibles pour des cas de formatage particuliers lorsque vous en avez besoin.

Lorsque vous souhaitez vraiment vous concentrer sur l'écriture, vous pouvez basculer l'éditeur en **Mode sans distraction**, où seul le panneau de l'éditeur de texte est visible, et la vue de la structure du projet est cachée.

Gardez un œil sur vos notes

La fenêtre principale peut éventuellement afficher une visionneuse de documents à droite de l'éditeur. La visionneuse est destinée à afficher un autre document de scène, vos notes de personnage, vos notes d'intrigue ou tout autre document auquel vous pourriez avoir besoin de vous référer lors de l'écriture. Il ne s'agit pas d'un panneau de prévisualisation du document que vous êtes en train d'éditer, mais si vous le souhaitez, vous pouvez également l'utiliser à cette fin.

Organisez vos documents à votre guise

Vous pouvez diviser votre projet de roman en autant de documents individuels que vous le souhaitez. Lorsque vous transformez le projet en manuscrit, tous les documents sont collés ensemble dans l'ordre où ils apparaissent dans l'arborescence du projet. Vous pouvez utiliser aussi peu de documents texte que vous le souhaitez, mais en divisant le projet en chapitres et en scènes, vous pouvez facilement les réorganiser à l'aide de la fonction « glisser-déposer » de l'arborescence du projet. Vous pouvez également commencer avec moins de documents, puis les diviser en plusieurs documents en fonction des titres des chapitres et des scènes.

Support de projets de plusieurs romans

Les principales parties de votre projet sont réparties dans des dossiers spéciaux de premier niveau appelés dossiers « racine ». Le texte principal de votre histoire se trouve dans le dossier racine « Roman ». Vous pouvez avoir plusieurs dossiers de ce type dans un projet et les renommer comme vous le souhaitez. Cela vous permet de conserver une série de romans individuels avec les mêmes personnages et la même construction du monde dans le même projet, et de créer des manuscrits pour chacun d'entre eux.

Tracez les éléments de votre histoire

Toutes les notes de votre projet peuvent se voir attribuer un *tag* que vous pouvez ensuite *référencer* à partir de n'importe quel autre document ou note. En fait, vous pouvez ajouter une nouvelle balise sous chaque titre d'une note si vous avez besoin de pouvoir référencer des sections spécifiques de celle-ci, ou si vous voulez conserver plusieurs sujets dans la même note.

Obtenez une vue d'ensemble de votre histoire

Ce ne sont pas les documents eux-mêmes qui définissent les chapitres et les scènes de votre histoire, mais les titres qu'ils contiennent. Dans la **Vue d'ensemble** de la fenêtre principale, vous pouvez voir un aperçu de tous les titres de chapitres et de scènes de chaque dossier racine de roman de votre projet. S'ils contiennent des références, par exemple quel personnage se trouve dans quel chapitre et quelle scène, celles-ci sont répertoriées dans des colonnes supplémentaires.

Vous pouvez également ajouter un synopsis à chaque chapitre ou scène, qui peut également être listé ici. Vous avez la possibilité d'ajouter ou de supprimer des colonnes d'informations dans ce plan. Un sous-ensemble des informations du plan est également disponible dans la **Vue roman** comme vue alternative à l'arborescence du projet.

Obtenez une vue d'ensemble des éléments de votre histoire

Sous le panneau de visualisation du document, vous trouverez une série d'onglets qui montrent les différents éléments de l'histoire pour lesquels vous avez créé des étiquettes. Les onglets sont classés en **Personnages**, **Intrigues**, etc., en fonction des catégories que vous utilisez dans votre histoire. Ce panneau peut être caché pour libérer de l'espace lorsque vous n'en avez pas besoin.

Assembler votre manuscrit

Que vous souhaitiez assembler un manuscrit, exporter toutes vos notes ou générer un plan de vos chapitres et scènes avec un synopsis, vous pouvez utiliser l'outil **Compiler le manuscrit** pour le faire. Cet outil vous permet de sélectionner les informations que vous souhaitez inclure dans le document généré, ainsi que son formatage. Vous pouvez envoyer le résultat à une imprimante ou générer un fichier Open Document ou Word Document qui peut être ouvert par la plupart des traitements de texte de type bureautique. Vous pouvez également générer le résultat sous forme de HTML ou de Markdown, tous deux adaptés à une conversion ultérieure dans d'autres formats.

1.2 Captures d'écran

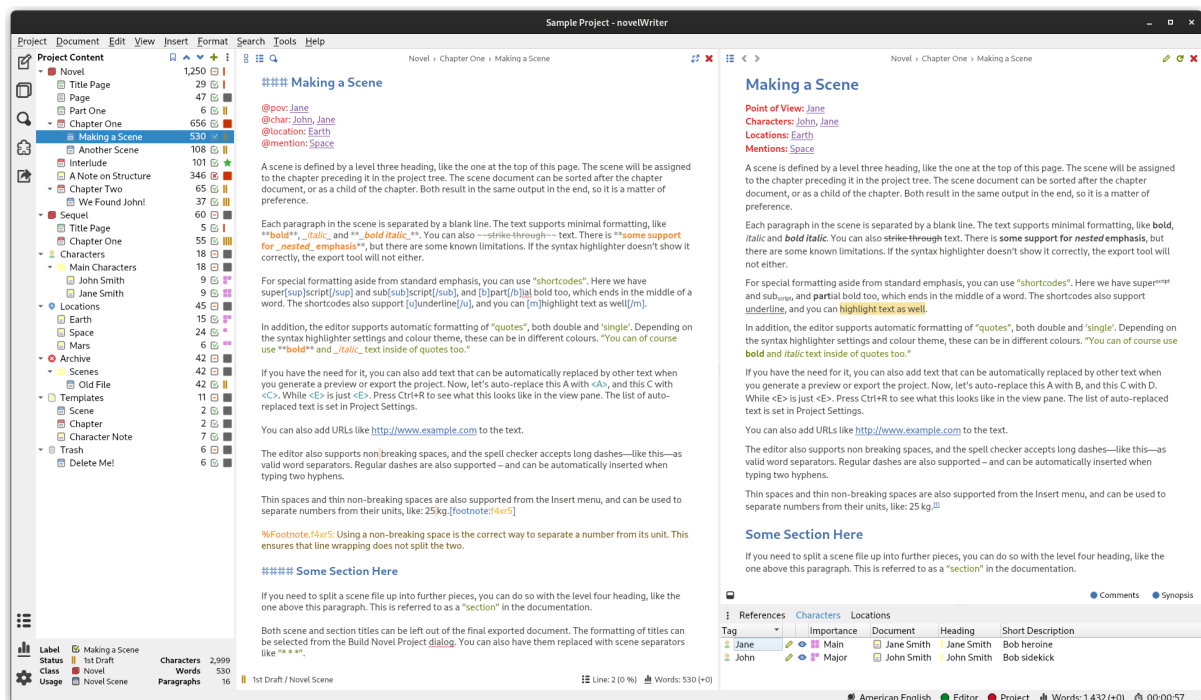


Figure1: novelWriter avec un thème de couleur claire

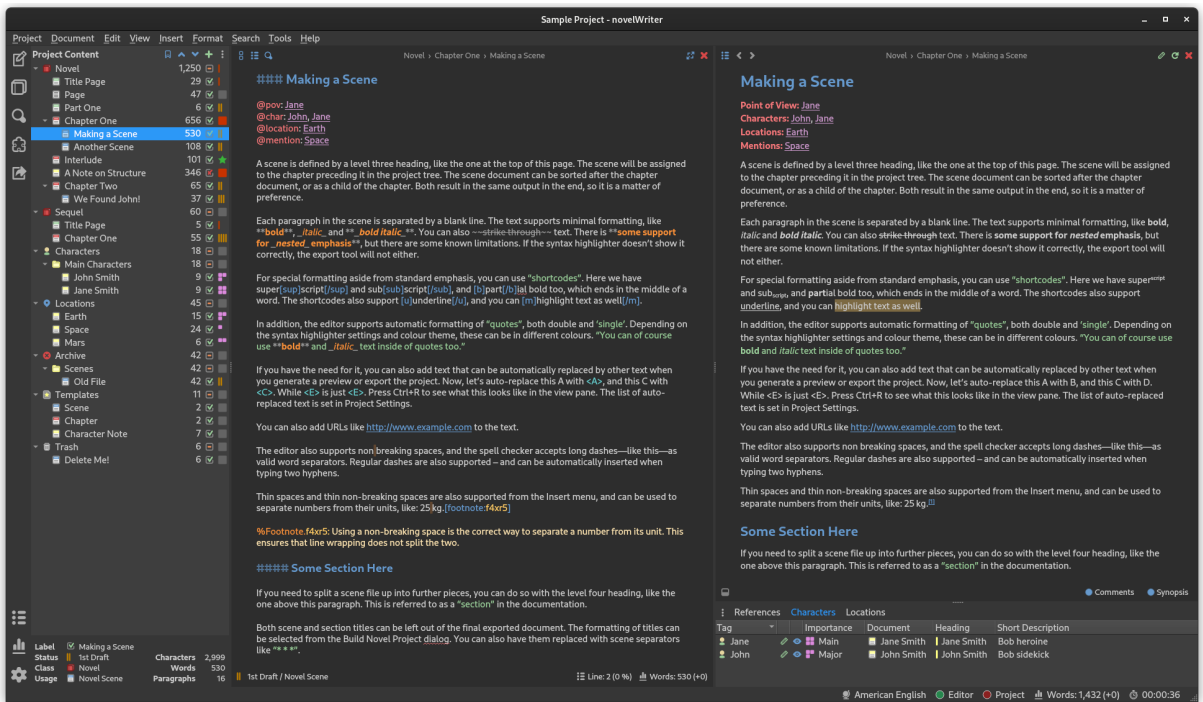


Figure2: novelWriter avec un thème de couleur sombre

Premières lectures

La documentation de novelWriter est assez complète. Il y a beaucoup de fonctionnalités auxquelles il faut s'habituer, mais vous n'avez pas besoin de toutes pour commencer.

Les chapitres ci-dessous intitulés « Informations essentielles » sont ceux que vous devez connaître pour utiliser correctement l'application. Par « correctement », on entend : de manière à ce que novelWriter comprenne la structure de base de votre texte. Il collecte de nombreuses informations à partir de votre texte et les utilise pour afficher la structure de celui-ci de différentes manières afin de vous aider à obtenir une vue d'ensemble de votre écriture.

Les chapitres intitulés « Lectures recommandées » contiennent des informations supplémentaires sur le fonctionnement des différentes parties de l'application et sur les fonctionnalités.

Les chapitres « Optionnel “ et ” Consulter » contiennent des informations supplémentaires ou des tables de consultation qui ne sont pas indispensables à l'utilisation de l'application.

2.1 Utiliser novelWriter

Afin d'utiliser novelWriter de manière efficace, vous devez connaître les bases de son fonctionnement. Les chapitres suivants expliquent les grands principes. Ils commencent par les bases et deviennent de plus en plus détaillés au fur et à mesure de la lecture.

Comment ça marche – Essential Information

Ce chapitre explique les bases du fonctionnement de l'application et ce qu'elle peut et ne peut pas faire.

Vues du projet – Recommended Reading

Ce chapitre explique plus en détail comment vous pouvez utiliser les composants de l'interface utilisateur pour organiser et visualiser votre travail de projet.

L'éditeur et l'afficheur – Recommended Reading

Ce chapitre explique plus en détail le fonctionnement de l'éditeur de texte et de l'affichage.

Mise en forme du texte – Essential Information

Ce chapitre traite de la mise en forme du texte. L'éditeur étant du texte brut, la mise en forme du texte nécessite un balisage de base. La structure de votre roman est également déduite de la

manière dont vous utilisez les titres. Les balises et les références sont mises en œuvre par des mots-clés spéciaux.

Raccourcis clavier – **Optional / Lookup**

Ce chapitre énumère tous les raccourcis clavier de novelWriter et leur fonction. La plupart des raccourcis sont également listés à côté de leurs entrées de menu dans l'application, ou dans les info-bulles. Ce chapitre est principalement destiné à servir de référence.

Notes typographiques – **Optional**

Ce chapitre vous donne un aperçu des symboles typographiques spéciaux disponibles dans novelWriter. La fonction de remplacement automatique peut gérer l'insertion des symboles de guillemets standard de votre langue, ainsi que d'autres caractères spéciaux. Si vous utilisez d'autres symboles, leur utilisation est expliquée ici.

2.2 Organiser vos projets

En plus de gérer une collection de fichiers de texte brut, novelWriter peut interpréter et cartographier la structure de votre roman et vous fournir des informations supplémentaires sur son déroulement et son contenu. Pour bénéficier de ces fonctionnalités, vous devez structurer votre texte d'une manière spécifique et ajouter des métadonnées à extraire.

Gestion des projets – **Essential Information**

Ce chapitre explique comment organiser le contenu de votre projet et comment mettre en place des sauvegardes automatiques de votre travail.

Structure du roman – **Essential Information**

Ce chapitre traite de la manière dont la structure de votre roman est encodée dans les documents textuels. Il explique comment les différents niveaux d'en-têtes sont utilisés, ainsi que certains formats spéciaux pour différents types d'en-têtes.

Tags et références – **Recommended Reading**

Ce chapitre explique comment organiser vos notes et comment fonctionne le système de balises et de références. Ce système vous permet d'établir des liens entre les documents de votre projet et d'afficher ces références dans l'interface de l'application.

Compiler le manuscrit – **Recommended Reading**

Ce chapitre explique le fonctionnement de l'outil **Manuscript Build**, la manière dont vous pouvez contrôler le formatage des titres de chapitre et la gestion des sauts de scène et de section.

2.3 Writing Aids

We are in the process of adding tools to novelWriter that can be used to track more information about your project to assist in your writing. These tools are entirely optional and will not be in your way if you choose to not use them.

Story Structure Comments – **Writing Tools**

Story structure comments are a way to add meta data that can be used to analyse the structure of your writing. These comments can be exported to spreadsheets for analysis.

2.4 Détails supplémentaires et sujets techniques

Les sections Détails supplémentaires et Sujets techniques contiennent des informations plus approfondies sur le fonctionnement des différents éléments de novelWriter. Ces informations ne sont pas essentielles pour commencer à utiliser novelWriter.

Configuration et installation

Des paquets prêts à l'emploi et des installateurs pour novelWriter sont disponibles pour toutes les plateformes principales, y compris Linux, Windows et MacOS, à partir de la page [Downloads](#). Voir ci-dessous les instructions d'installation supplémentaires pour chaque plateforme.

Vous pouvez également installer novelWriter à partir de l'index des paquets Python ([PyPi](#)). Voir [Installation à partir de PyPi](#). L'installation à partir de PyPi ne met pas en place de lanceurs d'icônes, vous devrez donc soit le faire vous-même, soit lancer novelWriter à partir de la ligne de commande.

La vérification orthographique dans novelWriter est assurée par une bibliothèque tierce appelée [Enchant](#). En général, elle devrait extraire automatiquement les dictionnaires de votre système d'exploitation. Cependant, sous Windows, ils doivent être installés manuellement. Voir [Dictionnaires de vérification orthographique](#) pour plus de détails.

3.1 Installation sous Windows

Vous pouvez installer novelWriter avec les dépendances Python et les bibliothèques intégrées en utilisant le fichier Windows Installer (setup.exe) depuis la page [Downloads](#), ou depuis la page [Releases](#) sur [GitHub](#). L'installation devrait être simple.

Si vous rencontrez des problèmes, essayez de désinstaller la version précédente et de procéder à une nouvelle installation. Si vous avez déjà installé une version par une autre méthode, désinstallez-la d'abord, car les installations multiples sont connues pour causer des problèmes.

3.2 Installation sous Linux

Un paquet Debian peut être téléchargé depuis la page [Downloads](#), ou depuis la page [Releases](#) sur [GitHub](#). Ce paquet devrait fonctionner sur Debian, Ubuntu et Linux Mint, au moins.

Si vous préférez, vous pouvez également ajouter le dépôt novelWriter sur Launchpad à votre gestionnaire de paquets. Les paquets Launchpad [sont signés par l'auteur](#).

3.2.1 Ubuntu

Vous pouvez ajouter le PPA Ubuntu et installer novelWriter avec les commandes suivantes.

```
sudo add-apt-repository ppa:vkbo/novelwriter
sudo apt update
sudo apt install novelwriter
```

Si vous voulez le PPA [Pre-Release](#) à la place, ajoutez le dépôt `ppa:vkbo/novelwriter-pre`.

3.2.2 Debian et Mint

Comme il s'agit d'un packaging purement Python, le PPA Launchpad peut en principe être utilisé sur Debian ou Mint. Cependant, la commande ci-dessus n'ajoutera pas la clé de signature, car elle est spécifique à Ubuntu.

Au lieu de cela, exécutez les commandes suivantes pour ajouter le dépôt et la clé :

```
sudo gpg --no-default-keyring --keyring /usr/share/keyrings/novelwriter-ppa-
↳keyring.gpg --keyserver hkps://keyserver.ubuntu.com:80 --recv-keys
↳F19F1FCE50043114
echo "deb [signed-by=/usr/share/keyrings/novelwriter-ppa-keyring.gpg] http://
↳ppa.launchpad.net/vkbo/novelwriter/ubuntu noble main" | sudo tee /etc/apt/
↳sources.list.d/novelwriter.list
```

Exécutez ensuite les commandes de mise à jour et d'installation comme pour Ubuntu :

```
sudo apt update
sudo apt install novelwriter
```

Astuce

Si vous obtenez un message d'erreur du type `gpg : failed to create temporary file` lors de l'importation de la clé depuis le serveur de clés Ubuntu, essayez de créer le dossier sur lequel l'erreur s'est produite, et importez à nouveau la clé :

```
sudo mkdir /root/.gnupg/
```

3.2.3 AppImage

Pour les distros Linux autres que celles mentionnées ci-dessus, l'option principale est [AppImage](#). Il s'agit d'images complètement autonomes pour l'application qui incluent l'environnement nécessaire pour faire fonctionner novelWriter. Elles peuvent bien sûr être exécutées sur n'importe quelle distribution Linux, si vous préférez cela aux paquets natifs.

Note

novelWriter ne supporte généralement pas les versions de Python qui ont atteint leur fin de vie. Si votre distro Linux utilise encore d'anciennes versions de Python et que novelWriter ne fonctionne pas, vous pouvez essayer l'AppImage à la place.

3.3 Installation sur MacOS

Vous pouvez installer novelWriter avec ses dépendances Python et bibliothèques intégrées en utilisant le fichier image de l'application DMG depuis la page [Downloads](#), ou depuis la page [Releases](#) sur [GitHub](#). L'installation devrait être simple.

- Téléchargez le fichier DMG et ouvrez-le. Faites ensuite glisser l'icône novelWriter dans le dossier *Applications* sur la droite. Cela l'installera dans votre *Applications*.
- La première fois que vous essayez de le lancer, il vous dira que le bundle ne peut pas être vérifié, appuyez simplement sur le bouton *Open* pour ajouter une exception.
- Si aucun bouton *Open* n'apparaît dans la boîte de dialogue, relancez l'application en faisant un clic droit sur l'application dans le Finder et en sélectionnant *Open* dans le menu contextuel.

Vous pouvez également accéder au menu contextuel par un clic d'option si vous disposez d'une souris à un bouton. Pour ce faire, maintenez la touche option de votre clavier enfoncée et cliquez sur l'application dans le Finder.

Note

Le DMG de novelWriter n'est pas signé parce qu'Apple ne fournit actuellement aucun moyen aux projets open source à but non lucratif de signer correctement leurs programmes d'installation. Le projet novelWriter n'a pas les moyens de payer des certificats de signature de logiciels commerciaux.

3.4 Installation à partir de PyPi

novelWriter est également disponible sur le Python Package Index, ou [PyPi](#). Cette méthode d'installation fonctionne sur tous les systèmes d'exploitation compatibles avec un environnement Python approprié.

Pour installer à partir de PyPi, vous devez d'abord avoir les commandes `python` et `pip` disponibles sur votre système. Vous pouvez télécharger Python depuis [python.org](#). Il est recommandé d'installer la dernière version. Si vous êtes sous Windows, assurez-vous également de sélectionner l'option « Add Python to PATH » pendant l'installation.

Pour installer novelWriter depuis PyPi, utilisez la commande suivante :

```
pip install novelwriter
```

Pour mettre à jour une installation existante, utilisez :

```
pip install --upgrade novelwriter
```

Lors de l'installation via `pip`, novelWriter peut être lancé depuis la ligne de commande avec :

```
novelwriter
```

Assurez-vous que l'emplacement d'installation de `pip` est dans votre variable `PATH`. Ce n'est pas toujours le cas par défaut, et vous pouvez alors obtenir une erreur « Not Found » lorsque vous exécutez la commande `novelwriter`.

CHAPTER 4

Trucs et astuces

Il s'agit d'une liste de petits conseils, que nous espérons utiles, sur la façon de tirer le meilleur parti de novelWriter.

Note

Cette section sera développée au fil du temps. Si vous souhaitez que quelque chose soit ajouté, n'hésitez pas à contribuer, ou à lancer une discussion sur la [Page des discussions](#) du projet.

4.1 Gestion du projet

Comment créer un projet à partir d'un modèle ?

Dans le formulaire **Créer un nouveau projet** de la boîte de dialogue d'accueil, vous pouvez choisir de « pré-remplir le projet » à partir du contenu d'un autre projet. Cette fonction est particulièrement utile si vous copiez un projet que vous avez dédié à être un projet modèle. Si vous avez une structure et des paramètres que vous souhaitez utiliser pour chaque nouveau projet, c'est la meilleure solution.

Comment fusionner plusieurs documents en un seul ?

Si vous avez besoin de fusionner une sélection de documents dans votre projet en un seul document, vous pouvez le faire en créant un nouveau dossier à cet effet, et en faisant glisser tous les documents que vous voulez fusionner dans ce dossier. Ensuite, vous pouvez faire un clic droit sur le dossier, sélectionner *Transformer* et *Fusionner les documents du dossier*.

Dans la boîte de dialogue qui s'ouvre, les documents seront dans le même ordre que dans le dossier, mais vous pouvez les réorganiser ici si vous le souhaitez. Voir [Séparation et fusion de documents](#) pour plus de détails.

Comment puis-je partager des étiquettes d'état ou d'importance entre les projets ?

Les étiquettes d'état ou d'importance que vous avez définies dans un projet peuvent être exportées à partir de **Paramètres du projet** dans les onglets de configuration respectifs. Vous pouvez ensuite importer ces étiquettes dans un autre projet.

4.2 Astuces de mise en page

Comment créer un tableau ?

Les outils de formatage disponibles dans novelWriter ne permettent pas de créer des structures complexes telles que des tableaux. Cependant, l'éditeur rend les tabulations de la même manière que les traitements de texte classiques. Vous pouvez définir la largeur d'un onglet dans les **Préférences**.

La touche de tabulation doit avoir la même distance dans l'éditeur que dans le visualiseur, de sorte que vous puissiez aligner du texte en colonnes à l'aide de la touche de tabulation, et que l'aspect soit le même lorsque vous regardez à côté de l'éditeur.

Cette méthode convient surtout à vos notes, car il n'est pas garanti que le résultat dans les documents exportés corresponde à la réalité. Surtout si vous n'utilisez pas la même police dans votre manuscrit que dans l'éditeur.

Comment forcer un saut de ligne lorsque les sauts de ligne sont ignorés dans mon manuscrit ?

Dans les **Paramètres de construction du manuscrit**, vous pouvez choisir d'ignorer les sauts de ligne dans les paragraphes de votre texte. Cependant, ces sauts de ligne sont parfois nécessaires. Comme par exemple sur la page de couverture où vous pouvez avoir besoin d'ajouter votre nom et votre adresse. Dans ce cas, vous pouvez ajouter `[br]` à l'endroit où vous souhaitez des sauts de ligne. Ces sauts de ligne ne peuvent être ignorés par aucun paramètre et seront toujours respectés.

Ajouté dans la version 2.6.

Comment désactiver l'indentation de la première ligne pour un paragraphe spécifique ?

Si l'indentation de la première ligne est activée, mais qu'un paragraphe spécifique ne doit pas être indenté, vous pouvez désactiver l'indentation en ajoutant explicitement l'alignement du texte. Par exemple, en ajoutant `<<` à la fin du paragraphe pour l'aligner à gauche. Les paragraphes alignés ne sont pas indentés.

Voir *Alignement et indentation des paragraphes* pour plus de détails.

4.3 Organiser son texte

Comment ajouter un texte d'introduction aux chapitres ?

Parfois, les chapitres sont précédés d'une courte préface, comme un court texte ou une citation pour planter le décor avant le début de la première scène.

Si vous ajoutez des fichiers distincts pour les chapitres et les scènes, le fichier de chapitre est l'endroit idéal pour ajouter ce type de texte. La séparation des fichiers de chapitres et de scènes vous permet également de faire des fichiers de scènes des documents enfants du chapitre.

Comment faire la distinction entre les ruptures de scènes douces et les ruptures de scènes dures ?

En fonction de votre style d'écriture, vous pouvez avoir besoin de faire une distinction entre les ruptures de scènes douces et les ruptures de scènes dures au sein des chapitres. Par exemple, si vous changez souvent de point de vue.

Dans ce cas, vous pouvez utiliser des en-têtes de scène différents pour les coupures de scène dures et douces. L'outil **Compiler le manuscrit** vous permettra de définir un format différent pour les scènes en utilisant les codes d'en-tête `###` et `###!` lors de la génération de votre manuscrit. Vous pouvez par exemple ajouter l'habituel « * * * » pour les coupures dures et sélectionner des coupures de scènes douces, qui inséreront simplement un paragraphe vide à leur place. Voir *Paramètres de compilation* pour plus de détails.

Ajouté dans la version 2.4.

4.4 Autres outils

Comment convertir mon projet au format yWriter ?

Il existe un outil qui vous permet de convertir un projet yWriter en un projet novelWriter, et vice versa.

L'outil est disponible à l'adresse peter88213.github.io/yw2nw

Documents du roman

Il s'agit de documents créés sous un *dossier racine* « Roman ». Ils se comportent différemment des *Notes de projet*, et ont plus de restrictions. Par exemple, ils ne peuvent pas exister dans des dossiers destinés uniquement aux notes de projet. Voir le chapitre *Structure du roman* pour plus de détails.

Dossier racine

Un « dossier racine » est un dossier de premier niveau de l'arborescence du projet dans novelWriter. Chaque type de dossier racine possède une icône spécifique pour l'identifier. Pour une vue d'ensemble des types de dossiers racines disponibles, voir *Structure du projet*.

En-têtes

Chaque niveau d'en-têtes dans *Documents du roman* a une signification spécifique en termes de structure de l'histoire. En d'autres termes, ils déterminent ce que novelWriter considère comme une partition, un chapitre, une scène ou une section de texte. Pour *Notes de projet*, les niveaux d'en-tête n'ont pas d'importance. Pour plus de détails sur les en-têtes dans les documents de roman, voir *Importance des titres*.

Index du projet

L'index du projet est un enregistrement de toutes les en-têtes d'un projet, avec toutes leurs métadonnées comme les commentaires de synopsis, les *tags* et les *références*. L'index du projet est mis à jour automatiquement, mais il peut aussi être régénéré manuellement à partir du menu *Outils* ou en appuyant sur F9.

Menu contextuel

Un menu contextuel est un menu qui apparaît lorsque vous faites un clic droit sur quelque chose dans l'interface utilisateur. Dans novelWriter, vous pouvez également ouvrir un menu contextuel en appuyant sur le raccourci clavier Ctrl+..

Mot-clé

Un mot-clé dans novelWriter est une commande spéciale que vous placez dans le texte de vos documents. Il ne s'agit pas de Markdown standard, mais ils sont utilisés dans novelWriter pour ajouter des informations qui sont interprétées par l'application. Par exemple, les mots-clés sont utilisés pour *tags* et *références*.

Les mots-clés doivent toujours être sur leur propre ligne, et le premier caractère de la ligne doit toujours être le caractère @. Le mot-clé doit également toujours être suivi du caractère :, et les valeurs transmises à la commande sont ajoutées à la suite, séparées par des virgules.

Notes de projet

Les notes de projet sont des documents libres qui peuvent être placés n'importe où dans votre projet. Vous ne devez pas utiliser ces documents pour les éléments de l'histoire, mais uniquement pour les notes. Les notes de projet sont les fichiers sources utilisés par le système de balises et de références. Voir le chapitre *Tags et références* pour plus de détails sur leur utilisation.

Référence

Une référence fait partie d'un ensemble de *mots-clés* qui peuvent être utilisés pour établir un lien avec un *tag* dans un autre document. Les mots-clés de référence sont spécifiques aux différents types de *dossier racine*. Une présentation complète est disponible dans le chapitre *Tags et références*.

Tag

Un tag est une valeur définie par l'utilisateur et attribuée comme tag à une section de votre *Notes de projet*. Elle est facultative et ne peut être définie qu'une seule fois par rubrique. Elle est définie en utilisant la syntaxe *mot-clé @tag : valeur*, où *valeur* est la partie définie par l'utilisateur. Chaque balise peut être référencée dans un autre fichier en utilisant l'un des mots-clés *référence*. Voir le chapitre *Tags et références* pour plus de détails.

Comment ça marche

Les principales fonctionnalités de novelWriter sont listées dans le chapitre *Présentation générale*. Dans ce chapitre, nous entrons dans les détails de leur implémentation. Il s'agit d'une vue d'ensemble. Plus loin dans cette documentation, ces fonctionnalités seront abordées plus en détail.

6.1 Mise en page et conception de l'interface graphique

L'interface utilisateur de novelWriter se veut aussi minimaliste que possible, tout en offrant les fonctionnalités utiles à l'écriture d'un roman.

La fenêtre principale ne comporte pas, par défaut, de barre d'outils d'édition, comme c'est le cas dans de nombreuses autres applications. Cela réduit l'encombrement et, comme les documents sont formatés avec des balises de style, la barre d'outils n'est pas nécessaire la plupart du temps. Néanmoins, une petite barre d'outils de formatage peut être affichée en cliquant sur le bouton le plus à gauche dans l'en-tête de l'éditeur de documents. Elle permet d'accéder rapidement aux codes de formatage standard.

La plupart des fonctions de formatage prises en charge sont accessibles par des raccourcis clavier pratiques. Ils sont également disponibles dans le menu principal sous **Format**, de sorte que vous n'avez pas à chercher les codes de formatage à chaque fois que vous en avez besoin. Pour référence, une liste de tous les raccourcis se trouve dans le chapitre *Raccourcis clavier*.

Note

novelWriter n'est pas conçu pour être un traitement de texte complet de type bureautique. Il ne prend pas en charge les images, les liens, les tableaux et autres structures et objets complexes souvent nécessaires pour de tels documents. Le formatage est limité aux titres, aux formats de texte de base en ligne, à l'alignement du texte et à quelques autres fonctions simples.

Sur le côté gauche de la fenêtre principale, vous trouverez une barre latérale. Cette barre comporte des boutons pour les vues standard entre lesquelles vous pouvez basculer, un lien rapide vers l'outil **Compiler le manuscrit**, ainsi qu'un ensemble d'outils liés au projet et un accès rapide aux paramètres dans la partie inférieure.

Ajouté dans la version 2.2: Un certain nombre de nouvelles options de formatage ont été ajoutées dans la version 2.2 pour permettre certains cas de formatage particuliers. En même temps, une petite barre d'outils de formatage a été ajoutée à l'éditeur. Elle est cachée par défaut, mais peut être ouverte en appuyant sur le bouton situé dans le coin supérieur gauche de l'en-tête de l'éditeur.

6.1.1 Arborescence du projet et vue de l'éditeur

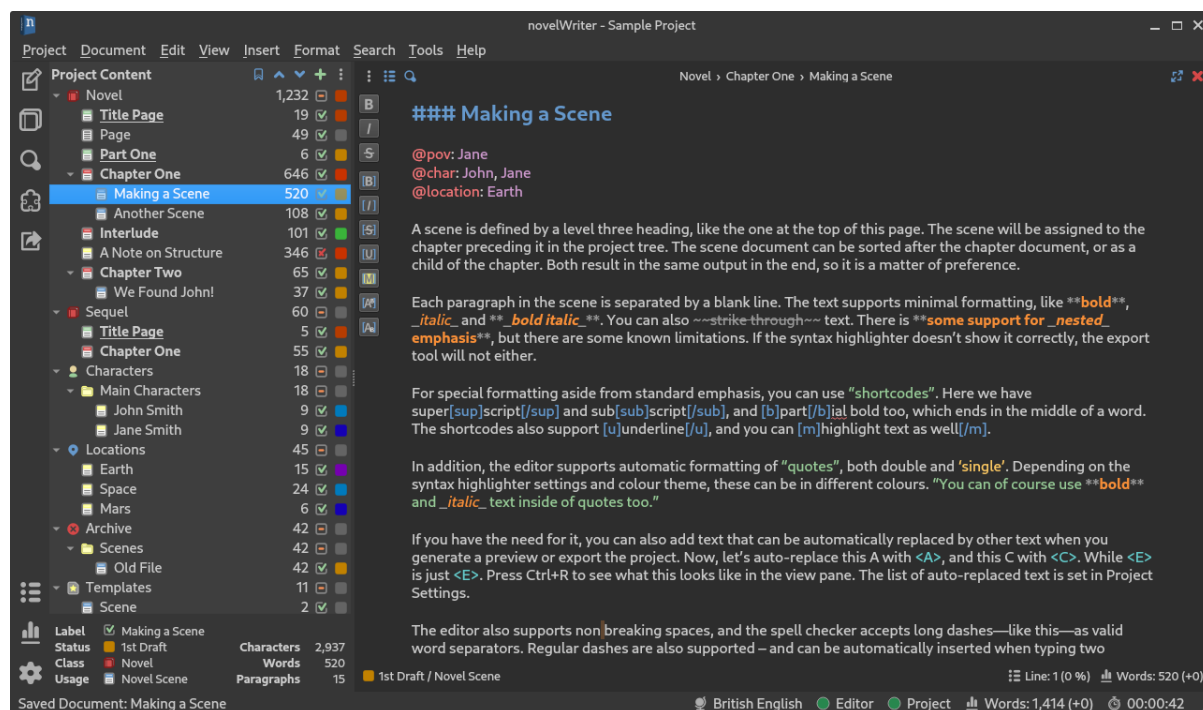


Figure1: Capture d'écran de l'arborescence du projet et de la vue de l'éditeur.

Lorsque l'application est en mode **Arborescence du projet**, la zone de travail de la fenêtre principale est divisée en deux, ou éventuellement trois, panneaux. Le panneau le plus à gauche contient l'arborescence du projet et tous les documents de votre projet. Le deuxième panneau est l'éditeur de documents.

Un troisième panneau facultatif sur le côté droit contient une visionneuse de documents qui peut afficher n'importe quel document de votre projet indépendamment de ce qui est ouvert dans l'éditeur de documents. Ce panneau n'est pas conçu comme une fenêtre de prévisualisation, bien que vous puissiez l'utiliser à cette fin si vous le souhaitez. Par exemple, si vous avez besoin de vérifier que les balises de formatage se comportent comme vous le souhaitez. Cependant, l'objectif principal de la visionneuse est de visualiser vos notes à côté de votre éditeur pendant que vous écrivez.

L'éditeur dispose également d'un **Mode Focus** que vous pouvez activer soit à partir du menu, soit à partir de l'icône dans l'en-tête de l'éditeur, soit en appuyant sur F8. Lorsque le **Mode Focus** est activé, tous les éléments de l'interface utilisateur autres que l'éditeur de document lui-même sont cachés.

L'arborescence du projet met en évidence, avec une couleur d'arrière-plan différente, le document actuellement ouvert dans l'éditeur.

Ajouté dans la version 2.6: Vous pouvez désormais faire glisser et déposer des documents de l'arborescence du projet sur les panneaux de l'éditeur ou de l'afficheur pour les ouvrir.

6.1.2 Vue du roman et vue de l'éditeur

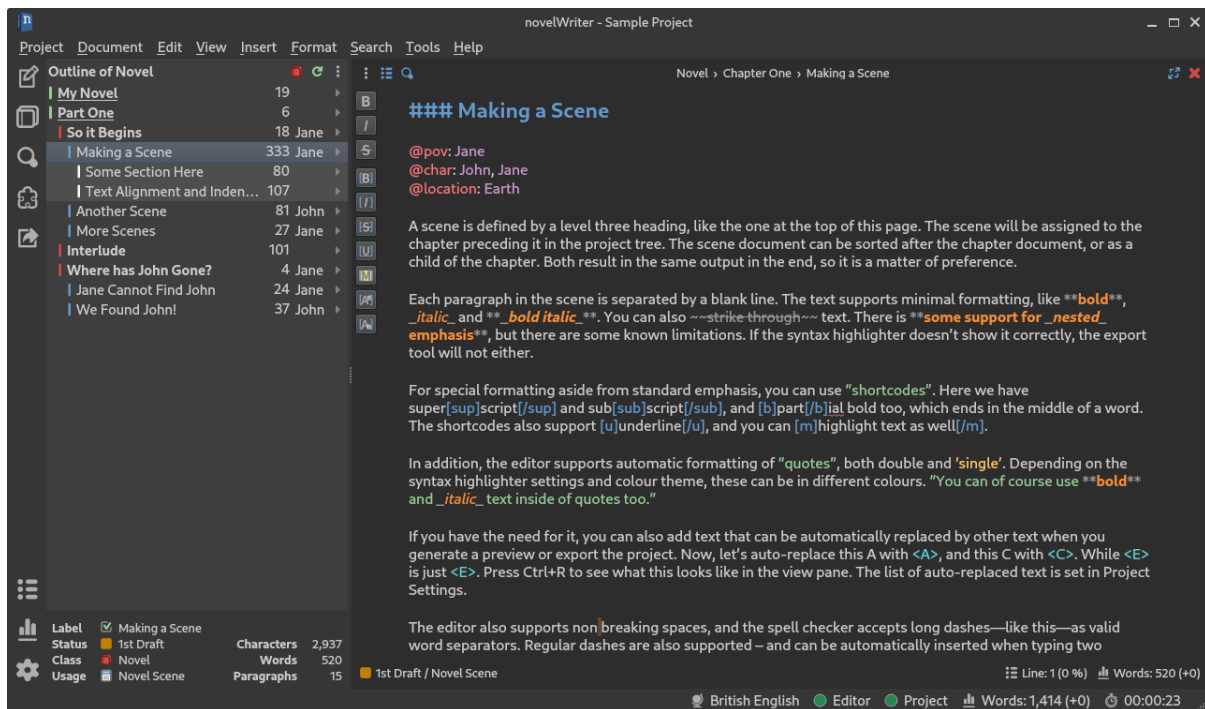


Figure2: Capture d'écran de l'arborescence des romans et de la vue de l'éditeur.

Lorsque l'application est en mode **Arborescence du roman**, l'arborescence du projet est remplacée par une vue d'ensemble de la structure de votre roman pour un *Novel dossier racine* spécifique. Au lieu de montrer des documents individuels, l'arbre montre maintenant tous les titres du texte de votre roman. Cela inclut les titres multiples au sein d'un même document.

Chaque titre est indenté en fonction du niveau du titre, et non de sa relation parent/enfant avec d'autres éléments de votre projet. Vous pouvez également ouvrir et modifier vos nouveaux documents à partir de cette vue. Toutes les rubriques contenues dans le document actuellement ouvert doivent être mises en évidence dans la vue pour indiquer celles qui appartiennent au même document.

Si vous avez plusieurs dossiers racine de type **Nouveau**, l'en-tête de la vue roman devient une liste déroulante. Vous pouvez alors passer de l'un à l'autre en cliquant sur le texte *Outline of ...*. Vous pouvez également cliquer sur le bouton de l'icône du roman situé à côté.

En général, la vue du roman doit être mise à jour lorsque vous apportez des modifications à la structure du roman, y compris les modifications du document courant dans l'éditeur. Les informations ne sont mises à jour que lorsque la sauvegarde automatique du document est déclenchée, ou lorsque vous appuyez manuellement sur **Ctrl+S** pour sauvegarder les modifications (vous pouvez ajuster l'intervalle de sauvegarde automatique dans **Préférences**). Vous pouvez également régénérer la vue du roman en entier en appuyant sur le bouton d'actualisation dans l'en-tête de la vue du roman.

Il est possible d'afficher une troisième colonne facultative dans la vue nouvelle. Les paramètres sont disponibles à partir du bouton de menu dans la barre d'outils.

Si vous cliquez sur l'icône triangulaire à droite de chaque élément, une infobulle s'affichera, indiquant toutes les métadonnées collectées pour cette rubrique.

6.1.3 Vue d'ensemble du roman

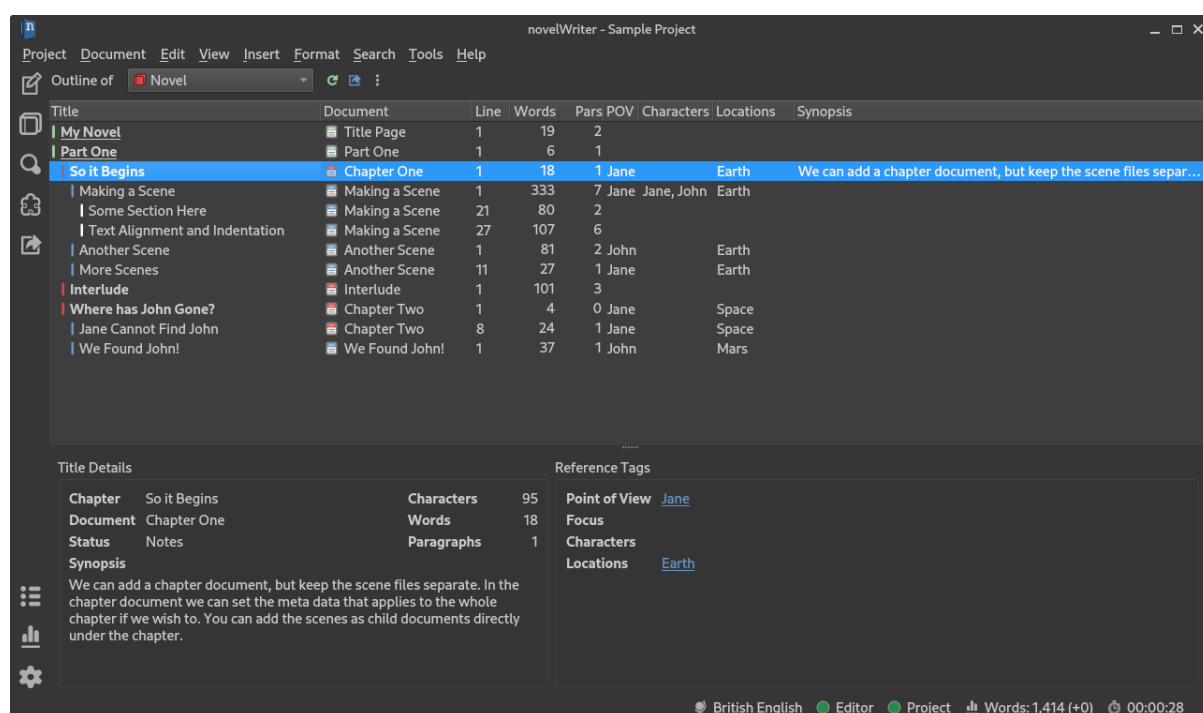


Figure3: Capture d'écran de la vue d'ensemble du roman.

Lorsque l'application est en mode **Vue d'ensemble du roman**, l'arbre, l'éditeur et l'afficheur sont remplacés par un grand tableau qui montre la structure complète du roman avec toutes les balises et les références listées. Pratiquement toutes les métadonnées collectées sont disponibles ici dans différentes colonnes.

Vous pouvez sélectionner le dossier racine du roman à afficher dans la liste déroulante, et vous pouvez sélectionner les colonnes à afficher ou à masquer à l'aide du bouton de menu. Vous pouvez également réorganiser les colonnes par glisser-déposer. L'application mémorise l'ordre et la taille des colonnes d'une session à l'autre et pour chaque projet individuel.

6.1.4 Thèmes

Par défaut, novelWriter utilise un thème de couleur claire. Vous pouvez également choisir un thème sombre standard avec des couleurs neutres, ou une série d'autres thèmes inclus, dans les **Préférences**.

Si vous le souhaitez, vous pouvez créer vos propres thèmes de couleurs, et même les ajouter à l'application. Voir *Syntaxe et thèmes de l'interface graphique* pour plus de détails.

Le changement de thème de couleur de l'interface graphique n'affecte pas les couleurs de l'éditeur et du visualiseur. Ils ont des couleurs séparées qui peuvent être sélectionnées à partir du paramètre « Thème de couleur du document » dans les **Préférences**. Elles sont séparées parce qu'il y a beaucoup plus d'options à choisir pour l'éditeur et le visualiseur.

Note

Si vous passez du mode clair au mode foncé dans l'interface graphique, vous devez également changer le thème de l'éditeur pour qu'il corresponde, sinon les icônes risquent d'être difficiles à voir dans l'éditeur et le visualiseur.

6.1.5 Recherche dans un projet

Un outil de recherche global est disponible dans la barre latérale. Il vous permet d'effectuer une recherche dans l'ensemble de votre projet. Cet outil n'offre pas de fonction de remplacement. Un outil de recherche et de remplacement est disponible dans l'éditeur de documents et agit sur le document ouvert.

Ajouté dans la version 2.4.

6.1.6 Changer de focus

Si la vue du projet ou du roman n'a pas le focus, presser **Ctrl+T** fait basculer le focus sur celle des deux qui est visible. Si l'une d'entre elles a déjà le focus, la pression sur la touche permet de passer de l'une à l'autre.

De même, en appuyant sur **Ctrl+E**, le focus passe à l'éditeur ou à l'afficheur du document, ou si l'un d'entre eux est déjà au focus, il passe de l'un à l'autre,

Ces deux raccourcis permettent de passer d'un élément de l'interface graphique à l'autre sans avoir à saisir la souris ou le pavé tactile.

6.2 Structure du projet

Il s'agit d'une brève introduction à la manière dont vous devez structurer vos projets d'écriture. Nous reviendrons plus en détail sur ce sujet ultérieurement.

L'intérêt principal de novelWriter est que vous êtes libre d'organiser les documents de votre projet comme vous le souhaitez, dans des sous-dossiers ou des sous-documents, et de répartir le texte entre ces documents de la manière qui vous convient. Tout ce qui importe à novelWriter est l'ordre linéaire dans lequel les documents apparaissent dans l'arborescence du projet (de haut en bas). Les chapitres, scènes et sections du roman sont déterminés par les titres de ces documents.

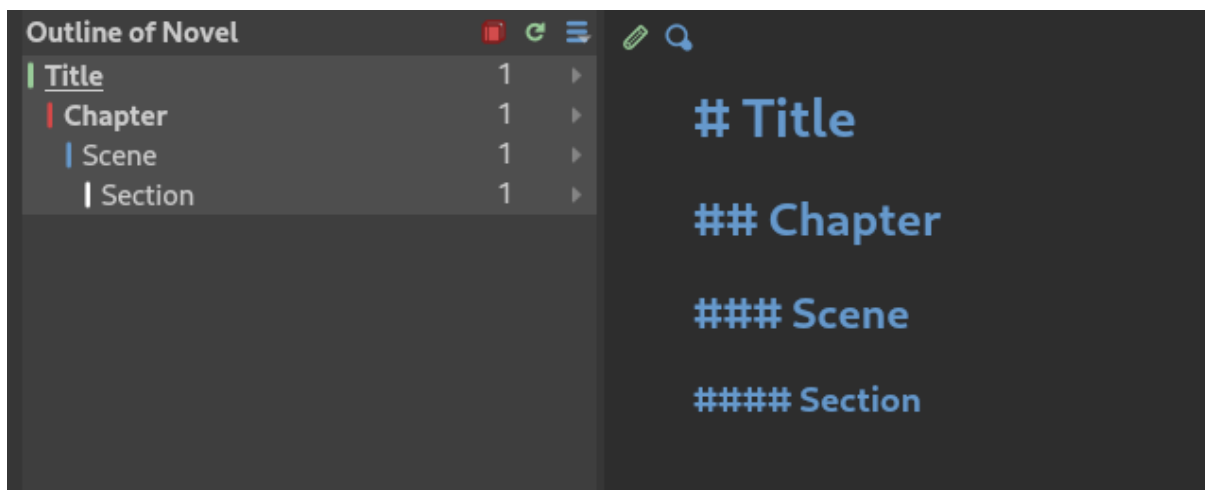


Figure4: Illustration de la correspondance entre les niveaux de rubriques et la structure du roman.

Les quatre niveaux de rubrique, **niveau 1** à **niveau 4**, sont traités comme suit :

- Le **niveau 1** est utilisé pour le titre du roman et pour les partitions.
- Le **niveau 2** est utilisé pour les titres de chapitres.
- Le **niveau 3** est utilisé pour les titres de scènes, éventuellement remplacés par des séparateurs.

- **Le niveau 4** est destiné aux titres de section à l'intérieur des scènes, si une telle granularité est nécessaire.

L'arborescence du projet sélectionnera une icône pour le document en fonction de la première rubrique qu'il contient.

Cette structure de niveaux de rubriques n'est prise en compte que pour les *documents du roman*. Pour *notes de projet*, les niveaux d'en-tête n'ont pas de signification structurelle, et vous êtes libre de les utiliser comme vous le souhaitez. Voir *Structure du roman* et *Tags et références* pour plus de détails.

Astuce

Vous pouvez ajouter des documents en tant qu'éléments enfants d'autres documents. C'est souvent plus utile que d'ajouter des dossiers, car vous pouvez de toute façon vouloir que l'en-tête de chapitre figure dans un document distinct de vos documents de scène individuels, afin de pouvoir réorganiser librement les documents de scène sans affecter l'emplacement des chapitres.

Ajouté dans la version 2.6: Les niveaux d'en-tête pour les parties, les chapitres et les scènes ne s'appliquent qu'à novelWriter. Lorsque vous générez votre manuscrit, les chapitres sont considérés comme le niveau d'en-tête le plus élevé, et les scènes comme le niveau inférieur. Les partitions sont insérées comme des éléments de texte dans la plupart des formats.

6.3 Élaboration d'un manuscrit

Le projet peut à tout moment être assemblé dans une gamme de formats différents grâce à l'outil **Compiler le manuscrit**. Nativement, novelWriter supporte [Open Document](#), Microsoft Word Document (connu sous le nom de [Office Open XML](#)), HTML5, et diverses saveurs de Markdown. Vous pouvez également générer un document PDF.

Le format HTML5 peut être converti par un certain nombre d'autres outils tels que [Pandoc](#), ou importé dans des traitements de texte si les formats Open Document ou Word Document ne conviennent pas. Le format Open Document est pris en charge par la plupart des applications bureautiques, tout comme le format Word Document. En outre, l'impression est également possible.

Pour un traitement avancé, vous pouvez exporter le contenu du projet vers un fichier JSON. Ceci est utile si vous souhaitez écrire votre propre script de traitement personnalisé, par exemple en Python, car le roman entier peut être lu dans un dictionnaire Python avec quelques lignes de code. Le fichier JSON peut être alimenté par du texte formaté en HTML ou par le texte brut tel qu'il a été tapé dans les documents du roman.

Voir [Compiler le manuscrit](#) pour plus de détails.

Ajouté dans la version 2.1: Vous pouvez désormais définir plusieurs définitions de construction dans l'outil **Compiler un manuscrit**. Cela vous permet de définir des paramètres spécifiques pour différents types de projets de documents, de documents de synthèse et de formats de manuscrits. Voir [Compiler le manuscrit](#) pour plus de détails.

6.4 Stockage du projet

Les fichiers d'un projet novelWriter sont stockés dans un dossier dédié au projet. La structure du projet est conservée dans un fichier à la racine de ce dossier appelé `nwProject.nwx`. Tous les fichiers de documents et les métadonnées associées sont stockés dans d'autres dossiers sous le dossier du projet.

Pour une description plus technique de la signification de tous les fichiers et de leur organisation, voir la section *Comment les données sont stockées*.

Ce mode de stockage des données a été choisi pour plusieurs raisons.

Tout d'abord, tout le texte que vous ajoutez à votre projet est enregistré directement dans le dossier de votre projet, dans des fichiers séparés. Seule la structure du projet et le texte que vous êtes en train d'éditer sont stockés en mémoire à un moment donné, ce qui signifie qu'il y a moins de risque de perdre des données si l'application ou votre ordinateur tombe en panne.

Deuxièmement, le fait d'avoir plusieurs petits fichiers signifie qu'il est très facile de les synchroniser entre ordinateurs à l'aide d'outils de synchronisation de fichiers standard.

Thirdly, if you use [version control](#) software to track the changes to your project, the file formats used for the files are well suited. All the JSON documents have line breaks and indents as well, which makes it easier to track them with version control software.

Note

Comme novelWriter doit garder la trace d'un grand nombre de fichiers et de dossiers lorsqu'un projet est ouvert, il peut ne pas fonctionner correctement sur certains systèmes de fichiers virtuels. Un fichier ou un dossier doit être accessible avec exactement le chemin avec lequel il a été sauvegardé ou créé. Un exemple où ce n'est pas le cas est la façon dont Google Drive est mappé sur les bureaux Linux Gnome en utilisant gvfs/gio.

Prudence

Vous ne devez pas ajouter vous-même des fichiers supplémentaires au dossier du projet. En règle générale, vous ne devez pas non plus modifier manuellement les fichiers qui s'y trouvent. Si vous devez vraiment éditer manuellement les fichiers texte, par exemple dans le cadre d'une tâche automatisée, vous devez reconstruire le *Index du projet* lorsque vous ouvrez à nouveau le projet.

L'édition de fichiers texte dans le dossier `content` est moins risquée car il s'agit de texte simple. Cependant, éditer le fichier XML du projet principal peut rendre le fichier du projet illisible et vous risquez de planter novelWriter et de perdre les informations sur la structure et les paramètres du projet.

CHAPTER 7

Vues du projet

Ce chapitre couvre plus en détail les différentes vues de projet disponibles dans novelWriter.

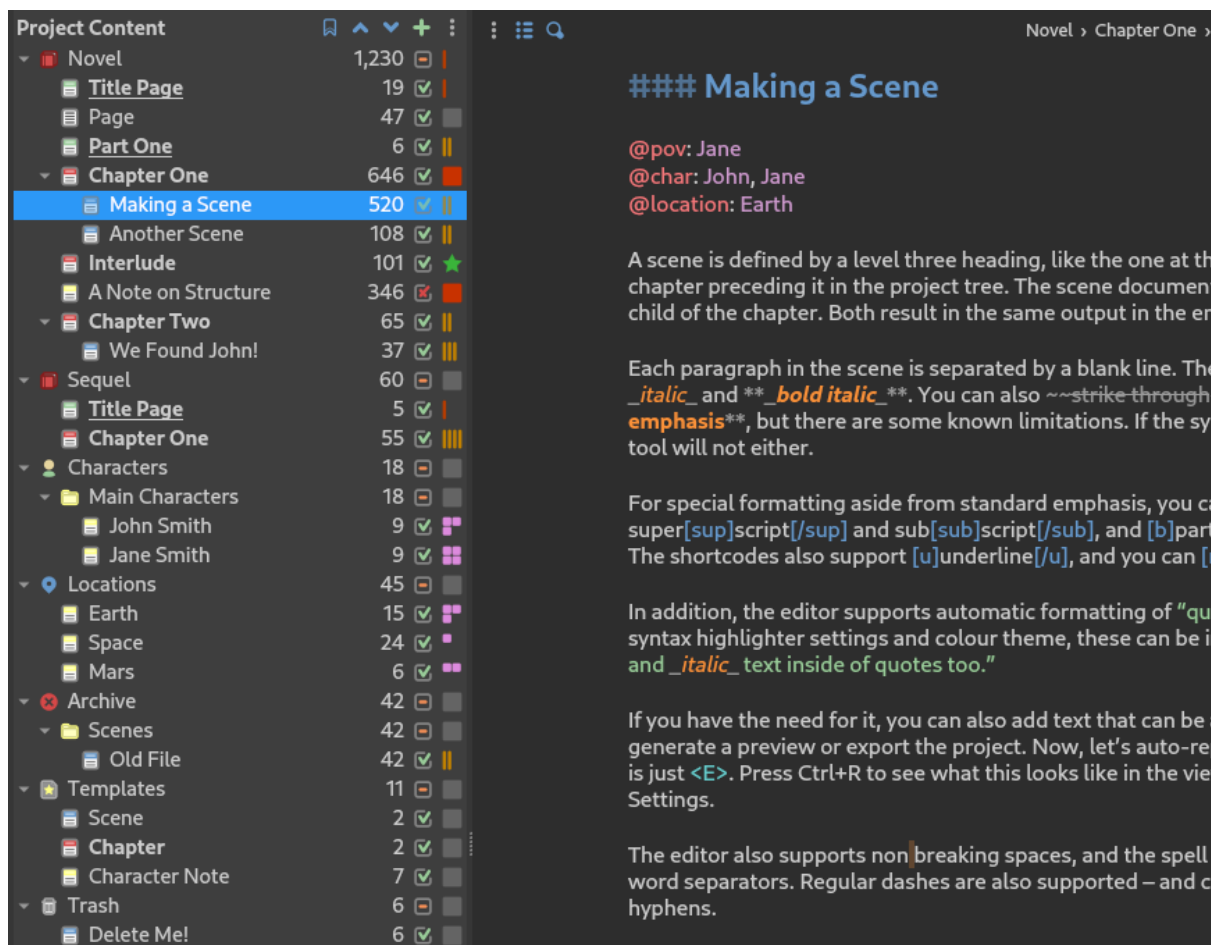


Figure1: L'arborescence du **Contenu du projet** telle qu'elle apparaît lors du chargement d'un exemple de projet.

7.1 L'arborescence du projet

La fenêtre principale contient l'arborescence du projet dans le panneau le plus à gauche. Elle montre la structure complète du projet et comporte quatre colonnes.

Colonne 1

La première colonne affiche l'icône et l'étiquette de chaque dossier, document ou note de votre projet. L'étiquette n'est pas la même que le titre de l'en-tête que vous avez défini dans le document. Toutefois, l'étiquette du document apparaît dans l'en-tête, au-dessus du texte du document lui-même, afin que vous sachiez à quel endroit du projet appartient un document ouvert. L'icône est sélectionnée en fonction du type d'élément et, pour les nouveaux documents, du niveau du premier titre dans le texte du document.

Colonne 2

La deuxième colonne indique le nombre de mots du document, ou la somme des mots des éléments enfants pour les dossiers et les documents avec sous-documents. Si les décomptes semblent incorrects, ils peuvent être mis à jour en reconstruisant le *index du projet* à partir du menu **Outils**, ou en appuyant sur F9.

Colonne 3

La troisième colonne indique si le document est considéré comme actif ou inactif dans le projet. Vous pouvez utiliser ce drapeau pour indiquer qu'un document est toujours dans le projet, mais qu'il ne doit pas être considéré comme une partie active de celui-ci. Lorsque vous exécutez l'outil **Compiler le manuscrit**, vous pouvez inclure ou exclure des documents en fonction de cet indicateur. Vous pouvez modifier cette valeur à partir du *menu contextuel* en cliquant avec le bouton droit de la souris.

Colonne 4

La quatrième colonne montre les étiquettes de statut ou d'importance définies par l'utilisateur que vous avez attribuées à chaque élément du projet. Voir *Importance et statut du document* pour plus de détails sur l'utilisation de ces étiquettes. Vous pouvez sélectionner ces étiquettes dans le *menu contextuel*, et les définir dans **Réglages du projet**.

Un clic droit sur un élément de l'arborescence du projet ouvre un menu contextuel sous le curseur, affichant une sélection d'actions pouvant être effectuées sur l'élément sélectionné.

En haut de l'arborescence du projet, vous trouverez une série de boutons.

- Le premier bouton est un bouton de liens rapides qui affiche un menu déroulant de tous les *dossiers racines* de votre projet. En sélectionnant l'un d'entre eux, vous vous déplacerez à la position correspondante dans l'arborescence. Vous pouvez également activer ce menu en appuyant sur Ctrl+L.
- Les deux boutons suivants permettent de déplacer des éléments vers le haut ou vers le bas dans l'arborescence du projet. C'est la seule façon de déplacer les dossiers racines.
- Le bouton suivant ouvre un menu déroulant permettant d'ajouter de nouveaux éléments à l'arborescence. Cela inclut les dossiers racines et les documents modèles. Vous pouvez également activer ce menu déroulant en appuyant sur Ctrl+N.
- Le dernier bouton est un menu d'actions supplémentaires que vous pouvez appliquer à l'arborescence du projet.

Sous l'arborescence du projet, vous trouverez un petit panneau de détails affichant les informations complètes de l'élément sélectionné. Ce panneau comprend également les derniers décomptes de paragraphes et de caractères, en plus du nombre de mots.

Astuce

Si vous souhaitez que l'étiquette d'un document soit identique à celle d'un en-tête, vous pouvez cliquer avec le bouton droit de la souris sur un en-tête du document lorsqu'il est ouvert dans l'éditeur et sélectionner *Définir comme nom du document* dans le menu contextuel.

7.1.1 Séparation et fusion de documents

Dans le sous-menu **Transformer** du menu contextuel d'un élément de l'arborescence du projet, vous trouverez plusieurs options permettant de modifier un document ou un dossier. Il s'agit notamment de passer d'un document à une note, mais aussi de les diviser en plusieurs documents ou de fusionner des éléments enfants en un seul document.

Découper des documents

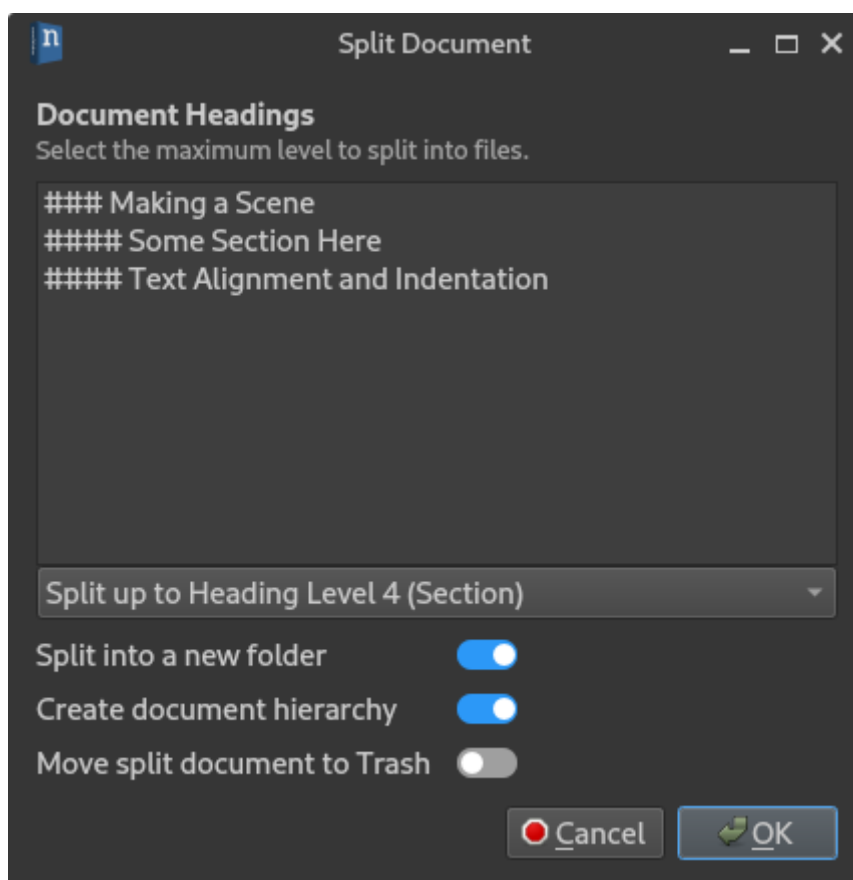


Figure2: La boîte de dialogue **découper un document**.

L'option **découper le document selon les titres** ouvre une boîte de dialogue qui vous permet de découper le document sélectionné en plusieurs nouveaux documents sur la base des en-têtes qu'il contient. Vous pouvez sélectionner le niveau de rubrique auquel le fractionnement doit être effectué dans la liste déroulante. La liste déroulante affiche un aperçu des rubriques qui seront scindées en nouveaux documents.

Vous avez la possibilité de créer un dossier pour ces nouveaux documents et de créer ou non une hiérarchie entre les documents. C'est-à-dire de placer les sections sous les scènes, et les scènes sous les chapitres.

Le document source **n'est pas** supprimé au cours du processus, mais vous avez la possibilité de laisser l'outil déplacer le document source dans le dossier *Corbeille*.

Fusionner des documents



Figure3: La boîte de dialogue **Fusionner des documents**.

Vous disposez de deux options pour fusionner des documents qui sont des éléments enfants d'un autre document. Vous pouvez soit **Intégrer les éléments enfants**, soit **Créer une fusion des éléments enfants**. La première option permet d'extraire tout le contenu des éléments enfants et de les fusionner dans le document parent, tandis que la seconde option crée un nouveau document au cours du processus.

Lorsque vous fusionnez des documents dans un dossier, seul ce dernier processus est possible, de sorte que seul le choix **Fusionner les documents du dossier** est disponible.

Dans les deux cas, la boîte de dialogue **Fusionner des documents** vous permet d'exclure les documents que vous ne souhaitez pas inclure et de les réorganiser si vous le souhaitez.

7.1.2 Importance et statut du document

Chaque document ou dossier de votre projet peut avoir un indicateur « État » ou « Importance ». Ce sont des indicateurs que vous contrôlez et définissez vous-même, et novelWriter ne les utilise pas pour quoi que ce soit. Pour modifier les étiquettes, allez dans leurs onglets respectifs dans **Paramètres du projet**.

L'indicateur « État » est destiné à marquer un *document de roman* comme étant par exemple un brouillon ou comme étant terminé, et l'indicateur « Importance » est destiné à marquer des notes sur les personnages, ou d'autres *notes de projet*, comme par exemple un personnage principal, majeur ou mineur, ou un élément de l'histoire.

L'utilisation de l'indicateur « État » ou « Importance » dépend du *dossier racine* dans lequel se trouve le document. S'il se trouve dans un dossier de type **Roman**, il utilise l'indicateur « État », sinon il utilise l'indicateur « Importance ».

7.1.3 Glisser-déposer de l'arborescence du projet

L'arborescence du projet permet d'effectuer un glisser-déposer pour réorganiser les documents et les dossiers. Le déplacement d'un document dans l'arborescence du projet affectera la position du texte lorsque vous assemblerez votre manuscrit dans l'outil **Compiler le manuscrit**.

Ajouté dans la version 2.2: Vous pouvez maintenant sélectionner plusieurs éléments dans l'arborescence du projet en maintenant la touche Ctrl ou Maj enfoncée pendant la sélection des éléments.

Les documents et leurs dossiers peuvent être réorganisés librement à l'intérieur de leur dossier racine. Si vous déplacez un **Document du roman** hors d'un dossier **roman**, il sera converti en **Notes de projet**. Les notes peuvent être déplacées librement entre tous les dossiers racine, mais gardez à l'esprit que si vous déplacez une note dans un dossier racine de type **roman**, son paramètre « Importance » sera remplacé par un paramètre « État ». Voir *Importance et statut du document*. L'ancienne valeur ne sera pas écrasée, et devrait être restaurée si vous la déplacez à nouveau à un moment donné.

Les dossiers racines de l'arborescence du projet ne peuvent pas être glissés et déposés. Si vous souhaitez les réorganiser, vous pouvez les déplacer vers le haut ou vers le bas à partir des boutons fléchés en haut de l'arborescence du projet, ou en appuyant sur Ctrl+Haut ou Ctrl+Bas lorsqu'ils sont sélectionnés.

Astuce

Vous pouvez faire glisser et déposer des documents sur le panneau de l'éditeur ou de l'afficheur pour les ouvrir.

7.2 L'arborescence du roman

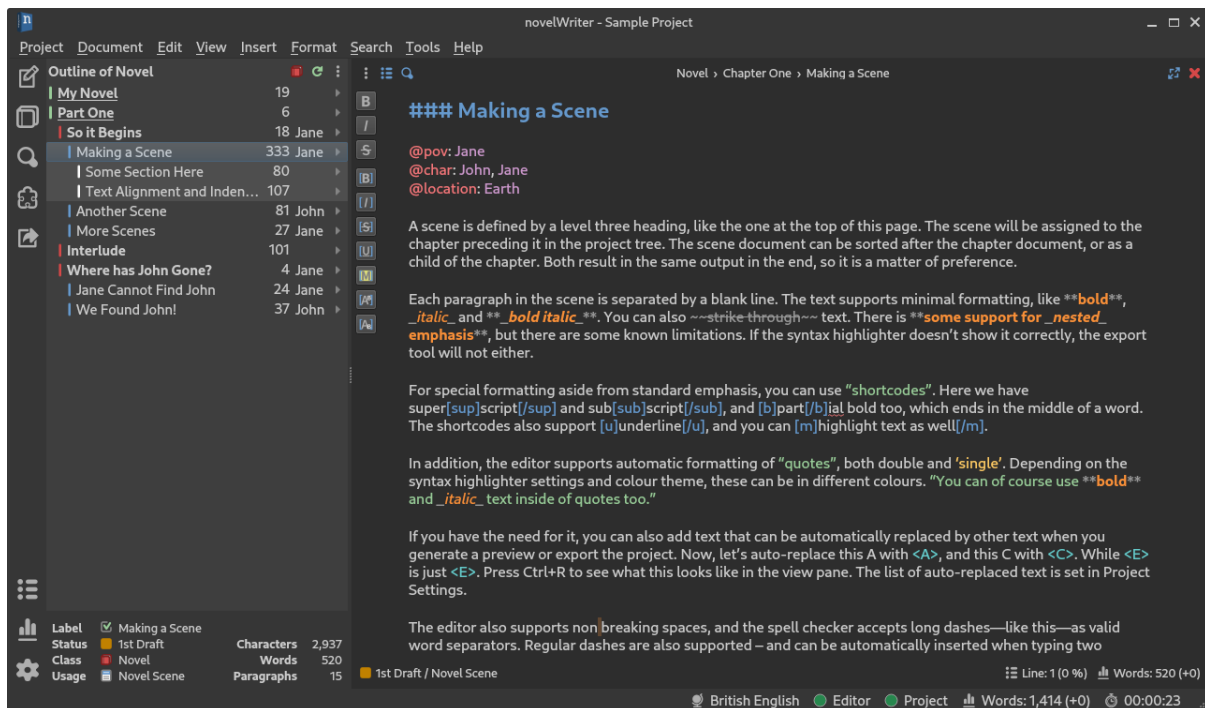


Figure4: Capture d'écran de l'arborescence d'un roman.

Une autre façon de visualiser la structure du projet est la vue roman. Vous pouvez passer à cette vue en sélectionnant le bouton *Vue Roman* dans la barre latérale. Cette vue est une version simplifiée de la vue

Vue d'ensemble du roman. Elle est pratique lorsque vous souhaitez parcourir la structure de l'histoire elle-même plutôt que les fichiers du document.

Note

Vous ne pouvez pas réorganiser les entrées dans la vue roman, ni ajouter de nouveaux documents, car cela impliquerait de restructurer le contenu des fichiers de documents eux-mêmes. Toute modification de ce type doit être effectuée dans l'arborescence du projet. Cependant, vous pouvez ajouter de nouveaux titres aux documents existants ou modifier les références, qui seront mises à jour dans cette vue lors de l'enregistrement du document.

7.3 Vue d'ensemble du roman

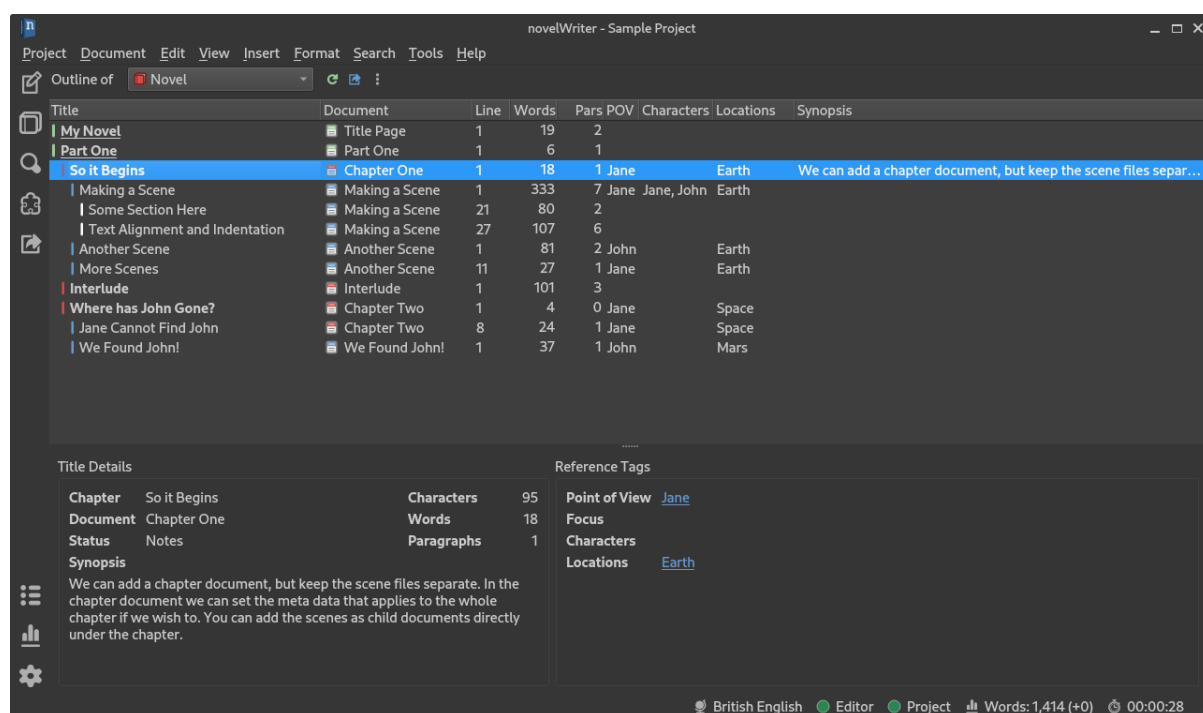


Figure5: Capture d'écran de l'affichage du plan du roman.

La **Vue d'ensemble du roman** du projet est disponible comme une autre option de vue dans la barre latérale. L'aperçu fournit une vue d'ensemble de la structure du roman, en affichant une hiérarchie arborescente des éléments du roman, c'est-à-dire les titres de niveau 1 à 4 représentant les partitions, les chapitres, les scènes et les sections.

Le document contenant le titre peut également être affiché dans une colonne séparée, ainsi que le numéro de ligne où le titre est défini. Un double-clic sur une entrée ouvre le document correspondant dans l'éditeur et permet de passer en mode **Vue de l'arborescence du projet**.

Vous pouvez sélectionner le dossier du roman à afficher dans le menu déroulant. Vous pouvez également choisir d'afficher une combinaison de tous les dossiers de romans.

Note

Comme la structure interne du roman ne dépend pas directement de la structure des dossiers et des

documents de l'arborescence du projet, cette vue ne sera pas nécessairement la même, selon la façon dont vous choisissez d'organiser vos documents. Voir la page *Structure du roman* pour plus de détails.

Diverses métadonnées et informations extraites de *tags* peuvent être affichées dans des colonnes du plan. Un ensemble de ces colonnes est visible par défaut, mais vous pouvez activer ou désactiver d'autres colonnes à partir du bouton de menu de la barre d'outils. L'ordre des colonnes peut également être réorganisé en les faisant glisser vers une autre position. Les paramètres des colonnes sont sauvegardés entre les sessions, par projet.

Note

La colonne **Titre** ne peut pas être désactivée ou déplacée.

Les informations affichées dans l'aperçu sont basées sur l'index du projet *index du projet*. Bien que novelWriter fasse de son mieux pour maintenir l'index à jour lorsque le contenu change, vous pouvez toujours le reconstruire manuellement en appuyant sur F9 si quelque chose ne va pas.

La vue d'ensemble elle-même peut être régénérée en appuyant sur le bouton d'actualisation. Par défaut, le contenu est actualisé chaque fois que vous basculez dans cette vue.

La colonne **Synopsis** de la vue d'ensemble tire ses informations d'un commentaire spécialement formaté. Voir *Commentaires et synopsis*.

L'éditeur et l'afficheur

Ce chapitre décrit plus en détail le fonctionnement de l'éditeur de documents et des panneaux de visualisation.

8.1 Édition d'un document

Pour éditer un document, double-cliquez dessus dans l'arborescence du projet, appuyez sur la touche **Return** lorsqu'il est sélectionné, ou faites-le glisser et déposez-le dans le panneau de l'éditeur. Cela ouvrira le document dans l'éditeur de documents. L'éditeur utilise une syntaxe de type Markdown pour certaines fonctionnalités, et une syntaxe spécifique à novelWriter pour d'autres. Le format de la syntaxe est décrit dans le chapitre *Mise en forme du texte*.

L'éditeur dispose d'un bouton d'agrandissement, qui permet de basculer en **Mode focus**, et d'un bouton de fermeture dans le coin supérieur droit. En haut à gauche, vous trouverez un bouton « Outils » qui ouvre une barre d'outils avec quelques boutons pour appliquer le formatage du texte, un menu déroulant pour naviguer entre les rubriques, et un bouton « Recherche » pour ouvrir la boîte de dialogue de recherche.

L'éditeur de documents et la visualisation affichent l'étiquette du document actuellement ouvert dans l'en-tête en haut du panneau d'édition ou de visualisation. En option, le chemin complet du projet vers le document peut être affiché. Cette option peut être définie dans les **Préférences**.

Astuce

En cliquant sur la barre de titre du document, vous sélectionnez le document dans l'arborescence du projet et révélez ainsi son emplacement, ce qui permet de le retrouver plus facilement dans un projet volumineux.

Tous les *références* de l'éditeur peuvent être ouverts dans le visualiseur en déplaçant le curseur sur l'étiquette et en appuyant sur **Ctrl+Retour**. Vous pouvez également faire un contrôle-clic avec votre souris.



Figure1: Capture d'écran du panneau de l'éditeur de documents.

8.1.1 Vérification de l'orthographe

Une bibliothèque tierce appelée Enchant est utilisée pour la vérification orthographique dans l'éditeur. Les commandes de vérification orthographique se trouvent dans le menu **Outils**. Vous pouvez également définir la langue de la vérification orthographique dans **Paramètres du projet**.

Cette bibliothèque de vérification orthographique prend en charge les mots personnalisés que vous pouvez ajouter en sélectionnant « Ajouter un mot au dictionnaire » dans le menu contextuel lorsqu'un mot est mis en évidence par le vérificateur orthographique comme étant mal orthographié. Les mots personnalisés sont gérés par projet et la liste des mots peut être modifiée à partir de l'outil **Lexique du projet** disponible dans le menu **Outils**.

Note

En général, les dictionnaires de vérification orthographique sont collectés à partir de votre système d'exploitation, mais ce n'est pas le cas sous Windows. Voir [Dictionnaires de vérification orthographique](#) pour savoir comment ajouter des dictionnaires de vérification orthographique sous Windows.

8.1.2 Auto-complétion de l'éditeur

Si vous tapez le caractère @ sur une nouvelle ligne, un menu contextuel apparaîtra, affichant les différents mots-clés disponibles. La liste se raccourcit au fur et à mesure que vous tapez. Une fois qu'un mot-clé a été sélectionné ou tapé, l'éditeur peut suggérer d'autres options basées sur le contenu de votre projet. Voir [L'auto-complétion des références](#) pour plus de détails.

Ajouté dans la version 2.2.

8.2 Visualisation d'un document

Tout document de l'arborescence du projet peut également être visualisé en parallèle dans une visionneuse de documents sur le côté droit. Pour visualiser un document, appuyez sur Ctrl+R, sélectionnez **Afficher ce document** dans le menu ou le menu contextuel, ou glissez-déposez le document sur le panneau de visualisation. Si vous disposez d'un bouton du milieu de la souris, un clic du milieu sur le document l'ouvrira également dans l'afficheur.

Le document visualisé ne doit pas nécessairement être le même que celui qui est en cours d'édition. Cependant, si vous *voyez* le même document, presser Ctrl+R depuis l'éditeur mettra à jour le document avec vos dernières modifications. Vous pouvez également appuyer sur le bouton de rechargement dans le coin supérieur droit du panneau de visualisation, à côté du bouton de fermeture, pour obtenir la même chose.

Dans la fenêtre de visualisation, les *références* deviennent des liens cliquables. En cliquant sur ces liens, le contenu de l'afficheur est remplacé par le contenu du document auquel la référence renvoie.

L'afficheur de documents conserve un historique des documents consultés, que vous pouvez parcourir à l'aide des boutons fléchés situés dans le coin supérieur gauche du visualiseur. Si votre souris possède des boutons de navigation vers l'avant et vers l'arrière, vous pouvez également les utiliser. Ils fonctionnent de la même manière que les fonctions de navigation vers l'avant et vers l'arrière d'un navigateur. Le bouton le plus à gauche est un menu déroulant qui permet de naviguer rapidement entre les rubriques du document. Le bouton d'édition situé à droite permet d'ouvrir le document visualisé dans l'éditeur.

En bas du panneau de visualisation se trouve un panneau **Références**. (S'il est caché, cliquez sur le bouton situé à gauche du pied de page pour le faire apparaître). Ce panneau contient un onglet Références avec

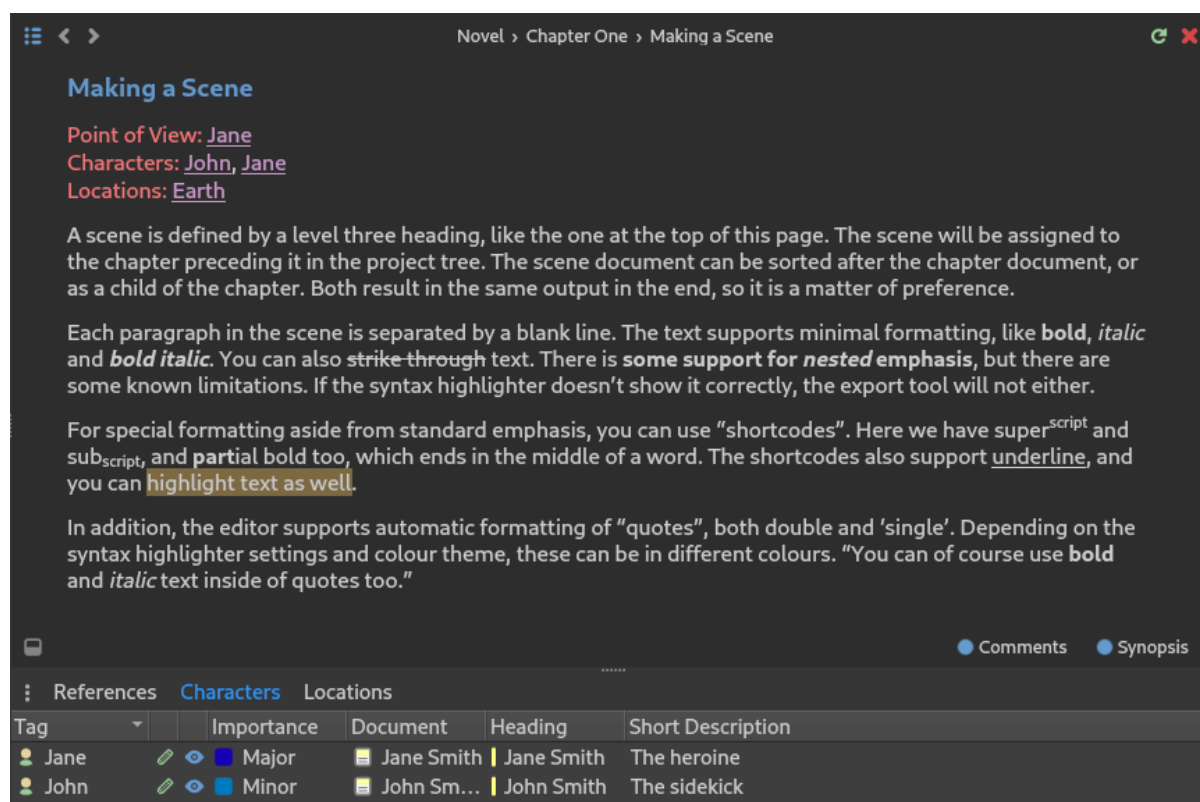


Figure2: Capture d'écran de la visualisation de documents.

des liens vers tous les documents qui renvoient à celui que vous êtes en train de visualiser, si un tel document a été défini. Si vous avez créé des dossiers racines et des étiquettes pour divers éléments de l'histoire, tels que les personnages et les points de l'intrigue, ceux-ci apparaîtront sous forme d'onglets supplémentaires dans ce panneau.

Note

Le panneau **Références** s'appuie sur un *index* à jour du projet. L'index est maintenu automatiquement. Cependant, si quelque chose manque ou semble erroné, l'index peut toujours être reconstruit en sélectionnant **Reconstruire l'index** dans le menu **Outils**, ou en appuyant sur F9.

Ajouté dans la version 2.2: Le panneau de référence a été redessiné et des onglets supplémentaires ont été ajoutés.

8.3 Rechercher & Remplacer

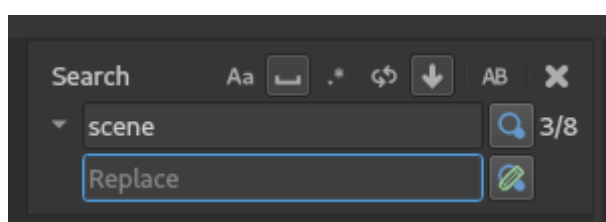


Figure3: Capture d'écran de la boîte de recherche de l'éditeur de documents.

L'éditeur de documents dispose d'un outil de recherche et de remplacement qui peut être activé avec **Ctrl+F** pour le mode recherche ou **Ctrl+H** pour le mode recherche et remplacement.

En appuyant sur **Retour** dans le champ de recherche, on recherche l'occurrence suivante du mot, et **Maj+Retour** l'occurrence précédente. En appuyant sur **Retour** dans la zone de remplacement, vous remplacerez le texte surligné et passerez au résultat suivant.

Il existe un certain nombre de paramètres pour l'outil de recherche, disponibles sous forme de commutateurs au-dessus de la boîte de recherche. Ils vous permettent de rechercher, dans l'ordre : la correspondance entre les majuscules et les minuscules, le mot entier uniquement, les expressions régulières, la recherche en boucle lorsque vous atteignez la fin du document et le passage au document suivant lorsque vous atteignez la fin du document. Il existe également un commutateur qui tente de faire correspondre la casse du mot au moment du remplacement. En d'autres termes, il essaiera de conserver le mot en majuscules, en minuscules ou en capitales pour qu'il corresponde au mot remplacé.

Voir aussi

Pour plus d'informations sur les possibilités de l'option Expression régulière, voir la documentation Qt pour la classe [QRegularExpression](#).

8.4 Remplacement automatique au fur et à mesure de la saisie

Quelques fonctions de remplacement automatique sont prises en charge par l'éditeur. Vous pouvez contrôler tous les aspects de la fonction de remplacement automatique dans les **Préférences**. Vous pouvez également désactiver entièrement cette fonction si vous le souhaitez.

Astuce

Si vous n'aimez pas l'auto-remplacement, tous les symboles insérés par cette fonctionnalité sont également disponibles dans le menu *Insertion*, et via *Raccourcis d'insertion*. Vous pouvez également utiliser une configuration *Compose Key*, ce qui signifie que vous n'avez pas du tout besoin de la fonction de remplacement automatique.

L'éditeur est capable de remplacer deux et trois traits d'union par des tirets courts et longs, les triples points par des points de suspension, et de remplacer les guillemets simples et doubles par des symboles de guillemets définis par l'utilisateur. Il tente également de déterminer s'il faut utiliser le symbole d'ouverture ou de fermeture, bien que cette fonction ne soit pas toujours exacte. En particulier, la distinction entre le guillemet simple fermant et l'apostrophe peut s'avérer délicate dans les langues qui utilisent le même symbole pour ces deux types de guillemets, comme c'est le cas en anglais.

Astuce

Si la fonction de remplacement automatique modifie un symbole alors que vous ne le souhaitez pas, appuyez sur **Ctrl+Z** une fois après le remplacement automatique pour l'annuler sans annuler le caractère que vous avez tapé avant lui.

8.5 Mise en évidence du dialogue

L'éditeur et l'afficheur peuvent optionnellement mettre en évidence les dialogues des personnages dans les documents de votre roman. Par défaut, cette fonctionnalité est configurée pour afficher le texte entre guillemets comme dialogue. Dans les **Préférences**, vous pouvez modifier ce paramètre pour prendre en charge d'autres marqueurs de dialogue qui utilisent des tirets pour indiquer le dialogue, comme ceux utilisés en espagnol, en portugais, en polonais, etc.

Différents styles sont pris en charge en modifiant les paramètres sous « Surlignage du texte » dans **Préférences**.

Il est également possible d'inclure la mise en évidence du dialogue dans vos manuscrits.

Mise en forme du texte

L'éditeur de texte novelWriter est un éditeur de texte simple qui utilise des codes de formatage pour définir les valeurs des métadonnées et permettre un certain formatage du texte. La syntaxe est basée sur Markdown, mais novelWriter n'est **pas** un éditeur Markdown. Il prend en charge le formatage de base comme les italiques, les gras et les biffures, ainsi que quatre niveaux de titres. Pour des besoins de formatage plus complexes, un ensemble de shortcodes peut être utilisé.

Outre les codes de formatage, novelWriter autorise les commentaires, une balise synopsis et un certain nombre de mots-clés et d'ensembles de valeurs utilisés pour *tags* et *références*. Certains codes s'appliquent également à des paragraphes entiers. Voir *Paragraphes du texte* pour plus de détails.

Les URL devraient également être mis en évidence et devenir cliquables. Cependant, seules les URL commençant par « http » ou « https » sont reconnues. Dans l'éditeur, vous devez maintenir la touche Ctrl enfoncée lorsque vous cliquez sur une URL pour la suivre.

9.1 Surlignage syntaxique

L'éditeur dispose d'une fonction de mise en évidence de la syntaxe qui vous permet de savoir si vous avez utilisé correctement les balises de formatage ou d'autres fonctions. Il modifie la couleur et la taille de police de vos titres, change la couleur du texte mis en évidence et peut également vous montrer les endroits où vous avez des dialogues dans votre texte.

Lorsque vous utilisez les mots-clés pour définir les balises et les références, celles-ci changent également de couleur. Les mots-clés corrects ont une couleur distincte, et les références elles-mêmes ont une couleur si elles sont valides. Les références non valides sont accompagnées d'une ligne d'erreur en pointillés. Il en va de même pour les balises dupliquées.

Il existe un certain nombre de thèmes de couleurs pour les surligneurs de syntaxe, à la fois pour les interfaces graphiques claires et foncées. Vous pouvez les sélectionner dans les **Préférences**.

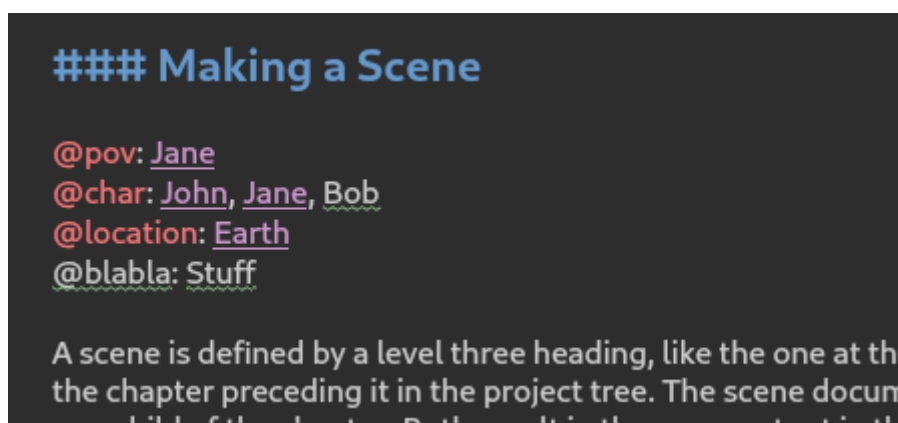


Figure1: Exemple de mise en évidence des références par la couleur. « Bob » n’est pas défini et “@blabla” n’est pas un type de référence valide.

9.2 En-têtes

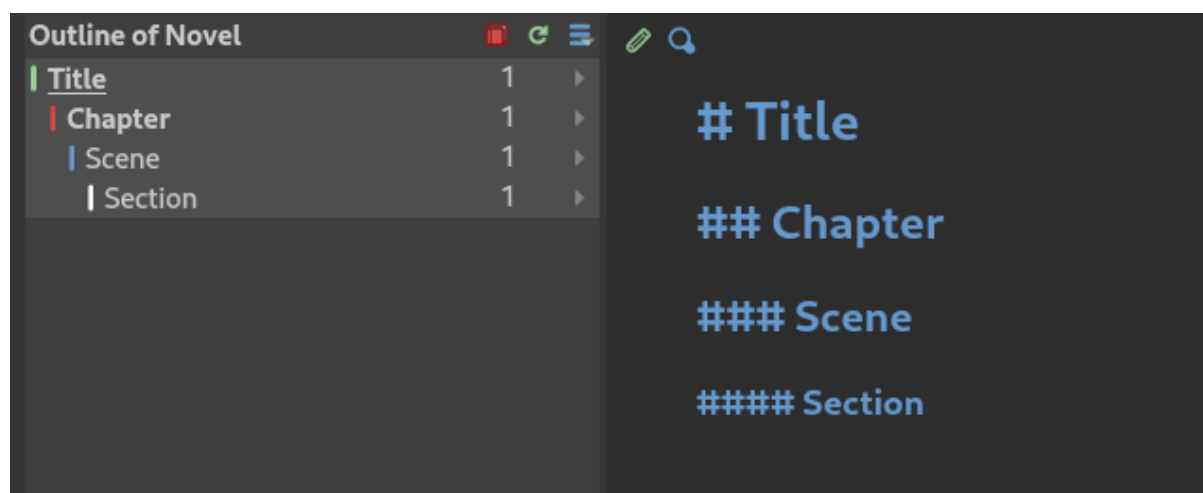


Figure2: Illustration de la correspondance entre les niveaux d’en-têtes et la structure du roman.

Quatre niveaux d’en-têtes sont autorisés. Pour *notes de projet*, ils sont libres d’être utilisés comme bon vous semble. En d’autres termes, novelWriter n’attribue pas de signification particulière aux différentes en-têtes. Cependant, pour *documents du roman*, ils indiquent le niveau structurel du roman et doivent être utilisés correctement pour produire le résultat escompté. Voir *Importance des titres* pour plus de détails.

Texte de titre

En-tête de niveau un. Pour les documents de type roman, le niveau indique le début d’une nouvelle partition. Les partitions sont utilisées lorsque vous souhaitez diviser votre histoire en « Partie 1 », « Partie 2 », etc. Vous pouvez également choisir de les utiliser pour diviser le texte en actes, puis de masquer ces titres dans votre manuscrit.

Texte de titre

En-tête de niveau deux. Pour les documents de type roman, le niveau indique le début d’un nouveau chapitre. Les numéros de chapitre peuvent être insérés automatiquement lors de la construction du manuscrit.

Texte de titre

En-tête de niveau trois. Pour les documents de type roman, le niveau indique le début d’une

nouvelle scène. Des numéros de scène ou des séparateurs de scène peuvent être insérés automatiquement lors de la construction du manuscrit. Vous pouvez donc utiliser le champ titre comme titre de travail pour vos scènes si vous le souhaitez, mais vous devez fournir un titre minimal.

Texte de titre

En-tête de niveau quatre. Pour les documents de type roman, le niveau indique le début d'une nouvelle section. Les titres de section peuvent être remplacés par des séparateurs ou ignorés lors de la construction du manuscrit.

Pour les en-têtes de niveau un à trois, l'ajout d'un ! modifie le sens de l'en-tête :

#! Texte de titre

Cela indique à l'outil de compilation que l'en-tête de niveau 1 est destiné à être utilisé pour le titre principal du roman ou du dossier de notes, comme par exemple le titre du roman sur la page de couverture. Lors de la construction du manuscrit, le style sera différent.

##! Texte de titre

Cela indique à l'outil de compilation de ne pas attribuer de numéro de chapitre à ce titre de chapitre si des numéros de chapitre automatiques sont utilisés. De tels titres sont utiles pour les prologues et les épilogues par exemple. Voir *Titres des chapitres non numérotés* pour plus de détails.

####! Texte de titre

Il s'agit d'une en-tête de scène alternatif qui peut être formaté différemment dans l'outil **Compiler le manuscrit**. Il est destiné à séparer les coupures de scène « douces » et « dures ». A part cela, il se comporte de la même manière qu'une en-tête de scène normale. Voir *Ruptures de scène dures et douces* pour plus de détails.

Note

L'espace après le caractère # ou ! est obligatoire. Le surligneur syntaxique change de couleur et de taille de police lorsque le titre est correctement formaté.

9.3 Paragraphes du texte

Un paragraphe de texte est indiqué par une ligne blanche. En d'autres termes, il faut deux sauts de ligne pour séparer deux fragments de texte en deux paragraphes. Les sauts de ligne simples sont traités comme des sauts de ligne à l'intérieur d'un paragraphe.

En outre, l'éditeur prend en charge quelques types d'espaces blancs supplémentaires :

- Un espace insécable peut être inséré avec **Ctrl+K, Espace**.
- Les espaces courtes sont également supportés, et peuvent être insérés avec **Ctrl+K, Maj+Espace**.
- Une espace courte insécable peut être insérée avec **Ctrl+K, Ctrl+Espace**.

Ce sont toutes des fonctions d'insertion, et le menu **Insertion** en propose d'autres. Les raccourcis clavier pour ces fonctions sont également listés dans *Raccourcis d'insertion*.

Les espaces insécables sont mis en évidence par le surligneur syntaxique avec un arrière-plan de couleur différente, en fonction du thème sélectionné.

Astuce

Les espaces insécables sont par exemple le type d'espace correct pour séparer un nombre de son unité. En général, les espaces insécables sont utilisés pour empêcher les algorithmes de retour à la ligne d'ajouter des sauts de ligne là où ils ne le devraient pas.

Note

Vous ne devez pas appliquer manuellement l'indentation du texte dans vos paragraphes dans l'éditeur. Ceux-ci peuvent être ajoutés automatiquement dans l'outil **Compilation du manuscrit** si vous souhaitez qu'ils figurent dans le manuscrit. Chaque nouveau paragraphe doit être séparé du contenu précédent par une ligne vide et sans indentation.

9.4 Mise en valeur du texte avec Markdown

Un ensemble minimal de styles d'accentuation de texte Markdown est pris en charge pour les paragraphes de texte.

texte

Le texte est rendu en italique.

texte

Le texte est rendu en gras.

~texte~

Barrer le texte.

Dans les guides Markdown, il est souvent recommandé de différencier le gras de l'italique en utilisant ****** pour le gras et pour l'italique, bien que Markdown supporte également **__** pour le gras et *** pour l'italique. Cependant, puisque la différenciation rend la mise en évidence et la conversion significativement plus simple et plus rapide, dans novelWriter il s'agit d'une règle et non d'une simple recommandation.

En outre, les règles suivantes s'appliquent :

1. Les balises de mise en valeur et de biffage n'autorisent pas les espaces entre les mots et la balise elle-même. Autrement dit, ****texte**** est valide, ****texte **** ne l'est pas.
2. Plus généralement, les délimiteurs doivent se trouver aux bords des mots. Autrement dit, « certains **textes en gras** ici » sont valides, « certains****** textes en gras****** ici » ne le sont pas.
3. Si vous utilisez à la fois ****** et pour envelopper le même texte, le trait de soulignement doit être à **l'intérieure**. Ceci est dû au fait que le trait de soulignement est également un caractère verbal valide, donc s'il est à l'extérieur, il viole la règle 2.
4. L'accentuation du texte ne s'étend pas au-delà des sauts de ligne. Si vous devez ajouter une emphase à plusieurs lignes ou paragraphes, vous devez l'appliquer à chacun d'entre eux à tour de rôle.
5. L'accentuation du texte ne peut être utilisée que dans les commentaires et les paragraphes. Les titres et les balises de métadonnées ne permettent pas la mise en forme, et toute marque de mise en forme sera rendue telle quelle.

Astuce

novelWriter supporte la syntaxe d'échappement standard pour les caractères d'accentuation au cas où l'éditeur ne comprendrait pas l'usage que vous voulez en faire. C'est-à-dire que *, _ et \~ généreront respectivement un *, _ et ~, sans les interpréter comme faisant partie du balisage.

9.5 Mise en forme avec les shortcodes

Pour des options de formatage supplémentaires, vous pouvez utiliser des shortcodes. Les shortcodes sont une forme de codes en ligne qui peuvent être utilisés pour modifier le format du texte qui suit le code d'ouverture et qui dure jusqu'à ce que la région de mise en forme soit terminée par un code de fermeture.

Ces shortcodes sont destinés à des cas de formatage spéciaux ou à des cas plus complexes qui ne peuvent pas être résolus avec de simples codes de formatage de type Markdown. Les shortcodes disponibles sont énumérés ci-dessous.

Table1: Formats des shortcodes

Syntaxe	Description
[b]texte[/b]	Le texte est rendu en gras.
[i]texte[/i]	Le texte est rendu en italique.
[s]texte[/s]	Le texte est rendu sous forme de texte barré.
[u]texte[/u]	Le texte est rendu sous forme de texte souligné.
[m]texte[/m]	Le texte est rendu en surbrillance.
[sup]texte[/sup]	Le texte est rendu en exposant.
[sub]texte[/sub]	Le texte est rendu sous forme d'indice.

Contrairement aux codes de style Markdown, ces codes peuvent être utilisés n'importe où dans un paragraphe. Même au milieu d'un mot si nécessaire. Vous pouvez également les combiner librement pour obtenir une mise en forme plus complexe.

Les shortcodes sont disponibles dans le menu **Format** et dans la barre d'outils de l'éditeur, qui peut être activée en cliquant sur l'icône la plus à gauche dans l'en-tête de l'éditeur.

Note

Les shortcodes ne sont pas traités tant que vous n'avez pas généré un aperçu ou un document manuscrit. Il n'y a donc pas de mise en évidence du texte entre les marqueurs de formatage. Il n'y a pas non plus de vérification de la logique de vos marqueurs. Vous devez vous assurer que les marqueurs de formatage d'ouverture et de fermeture se trouvent à l'endroit voulu.

Ajouté dans la version 2.2.

9.6 Commentaires et synopsis

En plus des fonctions de formatage ci-dessus, novelWriter permet également d'insérer des commentaires dans les documents. Le texte d'un commentaire est toujours ignoré par le compteur de mots. Le texte peut également être filtré lors de la construction du manuscrit ou de la visualisation du document.

Le premier mot d'un commentaire, suivi de deux points, peut faire partie d'une série de modificateurs qui indiquent que le commentaire est destiné à un usage spécifique. Par exemple, si le commentaire commence par **Synopsis:**, le commentaire est traité d'une manière spéciale et apparaîtra dans le *[Vue d'ensemble du roman](#)* dans une colonne dédiée. Le mot **synopsis** n'est pas sensible à la casse. S'il est correctement formaté, le surligneur syntaxique l'indiquera en modifiant la couleur du mot.

Les différents styles de commentaires sont les suivants :

% Le texte de votre commentaire ...

Il s'agit d'un commentaire. Le texte n'est pas rendu par défaut (il est possible d'y déroger), n'est pas visible dans la vue des documents et n'est pas pris en compte dans le décompte des mots. Il est destiné à vous permettre de prendre des notes dans votre texte pour votre propre intérêt, quel qu'il soit, et qui ne fait pas partie du texte de l'histoire. Voici le format général d'un commentaire.

%Synopsis: Votre texte de synopsis ...

Il s'agit d'un commentaire de synopsis. Il est généralement traité de la même manière qu'un commentaire normal, sauf qu'il est également capturé par l'algorithme d'indexation et affiché dans *[Vue d'ensemble du roman](#)*. Il peut également être filtré séparément lors de la construction du projet pour, par exemple, générer un document de synopsis de l'ensemble du projet.

%Short: Votre courte description ...

Il s'agit d'un commentaire de description courte. Il est identique au commentaire de synopsis (ils sont interchangeables), mais il est destiné à être utilisé pour les notes de projet. Le texte apparaît dans le panneau Référence, sous la vue de documents, dans la dernière colonne intitulée **Description sommaire**.

%Footnote.<key>: Votre texte de note de bas de page ...

Il s'agit d'un commentaire spécial attribué à un marqueur de note de bas de page. Voir *[Notes de bas de page](#)* pour savoir comment les utiliser dans votre texte.

Note

Un seul commentaire peut être marqué comme synopsis ou description sommaire pour chaque rubrique. Si plusieurs commentaires sont marqués comme synopsis ou Descriptions sommaires, le dernier sera utilisé et les autres ignorés.

9.7 Notes de bas de page

Les notes de bas de page sont ajoutées à l'aide d'un shortcode, associé à un commentaire correspondant au texte de la note. La correspondance se fait à l'aide d'une clé qui lie les deux. Si vous insérez une note de bas de page à partir du menu **Insérer**, une clé unique est générée pour vous.

La fonction d'insertion ajoutera le marqueur du shortcode de la note de bas de page à l'endroit où se trouve votre curseur dans le texte, créera le commentaire de la note de bas de page associée juste après le paragraphe et déplacera le curseur à cet endroit pour que vous puissiez immédiatement commencer à taper le texte de la note de bas de page.

Le commentaire de la note de bas de page peut se trouver n'importe où dans le document, de sorte que si vous souhaitez le placer, par exemple, au bas du texte, vous êtes libre de le faire.

Les clés de notes de bas de page ne doivent être uniques qu'à l'intérieur d'un document. Par conséquent, si vous copiez, déplacez ou fusionnez du texte, vous devez vous assurer que les clés ne sont pas dupliquées. Si vous utilisez les clés générées automatiquement à partir du menu **Insérer**, elles sont uniques parmi tous les documents indexés. Il n'est pas garanti qu'elles soient uniques par rapport aux notes de bas de

page du dossier Archives ou Corbeille, mais le risque de générer accidentellement la même clé deux fois dans un projet est relativement faible (1 sur 810 000).

Voici à quoi peut ressembler une note de bas de page insérée dans un paragraphe une fois terminée :

```
This is a text paragraph with a footnote[footnote:fn1] in the middle.
```

```
%Footnote.fn1: This is the text of the footnote.
```

Ajouté dans la version 2.5.

9.8 Texte ignoré

Si vous souhaitez ignorer complètement une partie du texte dans vos documents, mais que vous n'êtes pas prêt à l'effacer, vous pouvez ajouter %~ devant le paragraphe ou la ligne de texte. Cela permettra à novelWriter d'ignorer complètement le texte lors de la génération d'aperçus ou de la construction de manuscrits.

Il s'agit d'un meilleur moyen de supprimer le texte que de le convertir en commentaires normaux, car vous souhaiterez peut-être inclure des commentaires normaux dans vos prévisualisations ou votre projet de manuscrit.

Vous pouvez activer ou désactiver la fonction de texte ignoré pour un paragraphe en appuyant sur Ctrl+Maj+D sur votre clavier avec votre curseur quelque part dans le paragraphe.

Exemple :

```
%~ This text is ignored.
```

```
This text is a regular paragraph.
```

9.9 Tags et références

L'éditeur de documents prend en charge un ensemble de mots-clés utilisés pour définir des balises et faire des références entre les documents sur la base de ces balises.

Vous devez utiliser le mot-clé @tag: pour définir une balise. La balise peut être définie une fois par section définie par une en-tête. Si vous la définissez plusieurs fois dans la même rubrique, elle remplacera la précédente.

@tag: valeur

Un mot-clé de balise suivi de la valeur de la balise, comme par exemple le nom d'un personnage.

Les références peuvent être placées n'importe où dans une section et sont rassemblées en fonction de leur catégorie. Les références se trouvent sur le formulaire :

@mot clé: valeur1, valeur2, ..., valeurN

Un mot-clé de référence suivi d'une valeur ou d'une liste de valeurs séparées par des virgules.

Les balises et les références sont traitées en détail dans le chapitre *Tags et références*. Les mots-clés peuvent être insérés à la position du curseur dans l'éditeur via le menu **Insérer**. Si vous commencez à taper un @ sur une nouvelle ligne, un menu d'auto-complétion apparaîtra également pour suggérer des mots-clés.

9.10 Alignement et indentation des paragraphes

Le texte de tous les documents est par défaut aligné à gauche ou justifié, selon les paramètres définis dans les **Préférences**.

Vous pouvez remplacer l'alignement par défaut du texte sur des paragraphes individuels en spécifiant des balises d'alignement. Ces balises sont des doubles chevrons. Il peut s'agir de >> ou de <<. Vous les placez avant ou après le paragraphe, et elles « pousseront » le texte vers le bord vers lequel les crochets pointent. Cela devrait être assez intuitif.

L'indentation utilise une syntaxe similaire. Mais ici, vous utilisez un seul > ou < pour éloigner le texte du bord.

Exemples :

Table2: Alignement et indentation du texte

Syntaxe	Description
>> Texte aligné à droite	Le paragraphe est aligné à droite.
Texte aligné à gauche <<	Le paragraphe est aligné à gauche.
>> Texte centré <<	Le paragraphe est centré.
> Texte en retrait à gauche	Le texte a une marge gauche plus importante.
Texte en retrait à droite <	Le texte a une marge droite plus importante.
> Texte indenté gauche/droite <	Les deux marges du texte ont été augmentées.

Note

L'éditeur de texte n'affiche pas l'alignement et l'indentation en direct. Mais l'afficheur les affichera lorsque vous y ouvrirez le document. Ils seront bien sûr également reflétés dans le document généré par l'outil de création de manuscrits, à condition que le format prenne en charge l'alignement des paragraphes.

9.10.1 Alignement avec les sauts de ligne

Si le paragraphe comporte des retours à la ligne, les marqueurs de toutes les lignes sont combinés et utilisés pour l'ensemble du paragraphe. Pour le texte suivant, toutes les lignes seront centrées :

```
>> I am the very model of a modern Major-General
I've information vegetable, animal, and mineral
I know the kings of England, and I quote the fights historical
From Marathon to Waterloo, in order categorical <<
```

9.10.2 Alignement sur le retrait de la première ligne

Si vous avez activé le retrait de première ligne dans les paramètres de compilation de votre Manuscrit, vous voudrez probablement le désactiver pour le texte des versets. L'ajout de balises d'alignement entraînera la désactivation du retrait de première ligne pour ce paragraphe.

Le texte suivant sera toujours aligné sur la marge de gauche :


```
I am the very model of a modern Major-General <<
I've information vegetable, animal, and mineral
I know the kings of England, and I quote the fights historical
From Marathon to Waterloo, in order categorical
```

9.11 Espace vertical et sauts de page

Vous pouvez appliquer des sauts de page aux en-têtes de partition, de chapitre et de scène pour les documents de roman à partir de l'outil **Compiler le manuscrit**. Si vous avez besoin d'ajouter un saut de page ou un espacement vertical supplémentaire à d'autres endroits, il existe des codes spéciaux à cet effet.

L'ajout de plus d'un saut de ligne entre les paragraphes n'augmentera **pas** l'espace entre ces paragraphes lors de la compilation du projet. Pour ajouter de l'espace entre les paragraphes, ajoutez le texte `[vspace]` sur une ligne distincte, et l'outil de compilation insérera un paragraphe vide à sa place.

Si vous avez besoin de plusieurs paragraphes vides, il suffit d'ajouter deux points et un chiffre au code ci-dessus. Par exemple, en écrivant `[vspace:3]`, vous insérerez trois paragraphes vides.

Si vous devez ajouter un saut de page quelque part, placez le texte `[new page]` sur une ligne isolée avant le texte que vous souhaitez faire commencer sur une nouvelle page.

Note

Le code de saut de page est appliqué au texte qui le suit. Il ajoute une marque de « saut de page avant » au texte lors de l'exportation vers HTML ou Open Document. Cela signifie qu'une `[new page]` qui n'est suivie d'aucun texte n'entraînera pas de saut de page.

Exemple:

```
This is a text paragraph.
```

```
[vspace:2]
```

```
This is another text paragraph, but there will be two empty paragraphs
between them.
```

```
[new page]
```

```
This text will start on a new page if the build format has pages.
```

9.12 Insérer le nombre de mots dans le texte

La page de couverture d'un manuscrit indique normalement le nombre de mots. Toute valeur statistique collectée par novelWriter peut être insérée dans n'importe quel document. Vous pouvez générer le code pour cela à partir du menu **Insérer** sous **Compte de mots/caractères**.

La valeur insérée est le nombre réel pour l'ensemble de votre manuscrit, elle n'est donc pas renseignée tant que vous n'avez pas exécuté l'outil **Compilation du manuscrit**. Jusqu'à ce moment-là, ils apparaîtront sous la forme d'un « 0 » dans le panneau de visualisation.

CHAPTER 10

Raccourcis clavier

La plupart des fonctionnalités de novelWriter sont disponibles sous forme de raccourcis clavier. Il s'agit d'une liste de référence de ces raccourcis. La plupart d'entre eux sont également listés dans l'interface utilisateur de l'application.

Note

Sous MacOS, remplacez Ctrl par Cmd.

10.1 Raccourcis de la fenêtre principale

Raccourci	Description
F1	Ouvrir le manuel d'utilisation en ligne
F5	Ouvrir l'outil Compiler le manuscrit
F6	Ouvrir l'outil Statistiques d'écriture
F8	Basculer Mode focus
F9	Reconstruire l'index du projet
F11	Basculer en mode plein écran
Ctrl+,	Ouvrir la boîte de dialogue Préférences
Ctrl+E	Changement ou basculement du focus de l'éditeur ou du visualisateur
Ctrl+T	Changement de focalisation de l'arborescence du projet ou de la vue du roman
Ctrl+Q	Quitter novelWriter
Ctrl+Maj+,	Ouvrir la boîte de dialogue Paramètres du projet
Ctrl+Maj+O	Ouvrir la boîte de dialogue de bienvenue pour ouvrir ou créer un projet
Ctrl+Maj+S	Enregistrer le projet en cours
Ctrl+Maj+T	Basculer vers la vue d'ensemble
Ctrl+Maj+W	Fermer le projet en cours
Maj+F1	Ouvrir le manuel d'utilisation local (PDF) s'il est disponible
Maj+F6	Ouvrir la boîte de dialogue Détails du projet

10.2 Raccourcis de l'arborescence du projet

Raccourci	Description
F2	Modifier le libellé de l'élément sélectionné
Retour	Ouvrir le document sélectionné dans l'éditeur
Alt+Haut	Sauter ou aller à l'élément précédent au même niveau dans l'arborescence
Alt+Bas	Sauter ou aller à l'élément suivant au même niveau dans l'arborescence
Alt+Gauche	Sauter à l'élément parent dans l'arborescence
Alt+Droite	Sauter au premier élément enfant dans l'arborescence du projet
Ctrl+.	Ouvrir le menu contextuel de l'élément sélectionné
Ctrl+L	Ouvrir le menu Liens rapides
Ctrl+N	Ouvrir le menu Créer un nouvel élément
Ctrl+O	Ouvrir le document sélectionné dans l'éditeur
Ctrl+R	Ouvrir le document sélectionné dans le visualisateur
Ctrl+Haut	Déplacer l'élément sélectionné d'un cran dans l'arborescence
Ctrl+Bas	Déplacer l'élément sélectionné d'un cran dans l'arborescence
Ctrl+Maj+Suppr	Déplacer l'élément sélectionné vers la corbeille

10.3 Raccourcis de l'éditeur de documents

10.3.1 Raccourcis de recherche de texte

Raccourci	Description
F3	Trouver la prochaine occurrence du mot recherché
Ctrl+F	Ouvrir la recherche et rechercher le mot sélectionné
Ctrl+G	Trouver la prochaine occurrence du mot recherché
Ctrl+H	Ouvrir remplacer et rechercher le mot sélectionné (Mac Cmd+=)
Ctrl+Maj+1	Remplacer l'occurrence sélectionnée et passer à la suivante
Ctrl+Maj+G	Trouver l'occurrence précédente du mot recherché
Ctrl+Maj+F	Ouvrir la recherche de projet et recherchez le mot sélectionné
Maj+F3	Trouver l'occurrence précédente du mot recherché

10.3.2 Raccourcis de mise en forme du texte

Raccourci	Description
Ctrl+'	Mettre le texte sélectionné, ou le mot sous le curseur, entre guillemets simples
Ctrl+''	Mettre le texte sélectionné, ou le mot sous le curseur, entre guillemets
Ctrl+/	Modifier le format des commentaires pour le bloc ou le texte sélectionné
Ctrl+0	Supprimer le format d'un bloc ou d'un texte sélectionné
Ctrl+1	Modifier le format du bloc au niveau de titre 1
Ctrl+2	Modifier le format du bloc en titre de niveau 2
Ctrl+3	Modifier le format du bloc en titre de niveau 3
Ctrl+4	Modifier le format du bloc en titre de niveau 4
Ctrl+5	Aligner le bloc à gauche
Ctrl+6	Aligner le bloc au centre
Ctrl+7	Aligner le bloc à droite
Ctrl+8	Ajouter une marge gauche au bloc
Ctrl+9	Ajouter une marge droite au bloc
Ctrl+B	Formater le texte sélectionné, ou le mot sous le curseur, en gras
Ctrl+D	Formater le texte sélectionné, ou le mot sous le curseur, en le barrant
Ctrl+I	Formater le texte sélectionné, ou le mot sous le curseur, avec de l'italique
Ctrl+Maj+/'	Supprimer le format d'un bloc ou d'un texte sélectionné
Ctrl+Maj+D	Basculer le format de texte ignoré pour le bloc ou le texte sélectionné

10.3.3 Autres raccourcis de l'éditeur

Raccourci	Description
F7	Réexécuter le correcteur orthographique sur le document
Ctrl+.	Ouvrir le menu contextuel à l'emplacement actuel du curseur
Ctrl+A	Sélectionner tout le texte dans le document
Ctrl+C	Copier le texte sélectionné dans le presse-papiers
Ctrl+K	Activer les commandes d'insertion (voir la liste dans <i>Raccourcis d'insertion</i>)
Ctrl+R	Ouvrir ou recharger le document en cours dans le visualisateur
Ctrl+S	Enregistrer le document en cours
Ctrl+V	Coller le texte du presse-papiers à la position du curseur
Ctrl+W	Fermer le document en cours
Ctrl+X	Couper le texte sélectionné dans le presse-papiers
Ctrl+Y	Rétablir la dernière annulation
Ctrl+Z	Annuler les dernières modifications
Ctrl+Retour-arrière	Supprimer le mot précédant le curseur
Ctrl+Suppr	Supprimer le mot après le curseur
Ctrl+F7	Vérifier l'orthographe
Ctrl+Retour	Ouvrir la balise ou la référence sous le curseur dans le visualiseur
Ctrl+Maj+A	Sélectionner tout le texte du paragraphe en cours

10.3.4 Raccourcis d'insertion

Un ensemble de fonctions d'insertion sont également disponibles par le biais de raccourcis, mais elles nécessitent une double combinaison de séquences de touches. La fonction d'insertion est activée avec **Ctrl+K**, suivi d'une touche ou d'une combinaison de touches pour le contenu inséré.

Raccourci	Description
Ctrl+K, Espace	Insérer un espace insécable
Ctrl+K, _	Insérer un tiret long
Ctrl+K, .	Insérer un point de suspension
Ctrl+K, '	Insérer une apostrophe
Ctrl+K, *	Insérer une puce de liste
Ctrl+K, %	Insérer un symbole pour mille
Ctrl+K, ~	Insérer un tiret (de la même largeur qu'un chiffre)
Ctrl+K, -	Insérer un tiret court
Ctrl+K, 1	Insérer un guillemet simple gauche
Ctrl+K, 2	Insérer un guillemet simple à droite
Ctrl+K, 3	Insérer un double guillemet gauche
Ctrl+K, 4	Insérer un double guillemet droit
Ctrl+K, C	Insérer un mot-clé @char
Ctrl+K, E	Insérer un mot-clé @entity
Ctrl+K, F	Insérer un mot-clé @focus
Ctrl+K, G	Insérer un mot-clé @tag
Ctrl+K, H	Insérer un bref commentaire de description
Ctrl+K, L	Insérer un mot-clé @location
Ctrl+K, M	Insérer un mot-clé @mention
Ctrl+K, O	Insérer un mot-clé @object
Ctrl+K, P	Insérer un mot-clé @plot`
Ctrl+K, S	Insérer un commentaire de synopsis
Ctrl+K, T	Insérer un mot-clé @time
Ctrl+K, V	Insérer un mot-clé @pov
Ctrl+K, X	Insérer un mot-clé @custom
Ctrl+K, Ctrl+Espace	Insérer un espace fin insécable
Ctrl+K, Ctrl+_	Insérer une barre horizontale
Ctrl+K, Ctrl+'	Insérer une prime
Ctrl+K, Ctrl+"	Insérer une double prime
Ctrl+K, Ctrl+*	Insérer une fleur (puce alternative)
Ctrl+K, Ctrl+-	Insérer une puce à tiret (puce alternative)
Ctrl+K, Ctrl+D	Insérer un signe de division
Ctrl+K, Ctrl+O	Insérer un symbole de degré
Ctrl+K, Ctrl+X	Insert a times sign
Ctrl+K, Maj+Espace	Insérer un espace fin

10.4 Raccourcis de la fenêtre de visualisation des documents

Raccourci	Description
Alt+Gauche	Reculer dans l'historique des vues
Alt+Droite	Avancer dans l'historique des vues
Ctrl+C	Copier le texte sélectionné dans le presse-papiers
Ctrl+Maj+A	Sélectionner tout le texte du paragraphe en cours
Ctrl+Maj+R	Fermer le panneau de visualisation des documents

Notes typographiques

novelWriter prend en charge certains symboles typographiques qui ne sont généralement pas disponibles dans de nombreux éditeurs de texte. Cela inclut par exemple les guillemets unicode, les tirets, les points de suspension, les espaces fins, etc. Tous ces symboles sont disponibles dans le menu **Insérer**, et via des raccourcis clavier. Voir *Raccourcis d'insertion*.

Ce chapitre fournit des informations supplémentaires sur la manière dont novelWriter gère ces symboles.

11.1 Remarques spéciales sur les symboles

Cette section contient des notes supplémentaires sur les symboles spéciaux disponibles.

11.1.1 Tirets et points de suspension

Lorsque la fonction de remplacement automatique est activée (voir *Remplacement automatique au fur et à mesure de la saisie*), les traits d'union multiples sont automatiquement convertis en tirets courts et longs, et les trois points en points de suspension. Le dernier remplacement automatique peut toujours être annulé avec la commande d'annulation **Ctrl+Z**, ramenant le texte à ce que vous aviez tapé avant le remplacement automatique.

En outre, le « tiret chiffré » est disponible. Il s'agit d'un tiret de la même largeur que les chiffres de la même police, pour la plupart des polices. Il permet de bien aligner les chiffres dans les colonnes lorsque vous devez utiliser un tiret.

11.1.2 Guillemets simples et doubles

Tous les différents guillemets listés sur la page Wikipedia [Quotation Mark](#) sont disponibles, et peuvent être sélectionnés comme symboles auto-remplacés pour les touches de guillemets simples et doubles. Les réglages se trouvent dans les **Préférences**.

Normalement, le texte entre guillemets est mis en évidence par l'éditeur. Il s'agit d'une commodité pour mettre en évidence les dialogues entre les personnages. Cette fonctionnalité peut être désactivée dans les **Préférences** si elle n'est pas souhaitée.

L'éditeur fait la distinction entre le texte entre guillemets droits et les symboles de guillemets sélectionnés par l'utilisateur. Cela permet au rédacteur de reconnaître les parties du texte qui n'utilisent pas les symboles de guillemets choisis. Deux fonctions pratiques du menu **Format** peuvent être utilisées pour reformater une section de texte sélectionnée avec les symboles de guillemets corrects.

11.1.3 Simple et double prime

Les symboles de prime simple et double sont disponibles dans le menu **Insérer**. Ces symboles sont les symboles corrects à utiliser pour les symboles d'unités pour les pieds, les pouces, les minutes et les secondes. L'utilisation de ces symboles est décrite plus en détail sur la page [Prime](#) de Wikipedia. Ils ressemblent beaucoup aux guillemets simples et doubles et peuvent être rendus de la même manière par la police, mais ils ont des codes différents. En les utilisant correctement, vous éviterez que les fonctions de remplacement automatique et de mise en évidence des dialogues ne se méprennent sur leur signification dans le texte.

11.1.4 Modificateur Lettre Apostrophe

La fonction de remplacement automatique considère tout guillemet droit simple orienté vers la droite comme un symbole de guillemet, même s'il s'agit d'une apostrophe. Cela s'applique également à l'outil de mise en évidence de la syntaxe, qui peut supposer que la première apostrophe qui suit est le symbole de fermeture d'une région de texte entre guillemets.

Pour contourner ce problème, il existe une apostrophe alternative. Il s'agit d'un caractère Unicode spécial qui n'est pas considéré comme une ponctuation, mais comme un modificateur. Il est généralement rendu de la même manière que les guillemets simples de droite, en fonction de la police de caractères. Il existe un article de Wikipédia sur la [lettre modificatrice apostrophe](#) avec plus de détails.

Note

Lors de l'exportation à l'aide de l'outil **Compiler un manuscrit**, ces apostrophes seront automatiquement remplacées par le symbole de guillemet simple de droite correspondant, comme cela est généralement recommandé. Par conséquent, cela n'a pas vraiment d'importance si vous ne les utilisez que pour corriger la mise en évidence de la syntaxe.

11.1.5 Symboles spéciaux d'espacement

Quelques variations du caractère espace régulier sont prises en charge. La manière typographique correcte de séparer un nombre de son unité est d'utiliser une [thin space](#). Sa largeur est généralement égale aux deux tiers de celle d'une espace normale. Pour les nombres et les unités, il doit en outre s'agir d'un espace insécable, c'est-à-dire que l'habillage du texte ne doit pas ajouter de saut de ligne sur cet espace particulier.

Un espace régulier peut également être transformé en espace insécable si nécessaire.

Tous les espaces insécables sont mis en évidence par un arrière-plan de couleur différente pour faciliter leur repérage dans le texte. La couleur dépend du thème de couleur sélectionné.

Les espaces fins et insécables sont convertis en codes HTML correspondants lors de l'exportation au format HTML.

Gestion des projets

Vos textes dans novelWriter sont organisés en projets. Chaque projet est censé contenir un roman et les notes associées. Si vous avez plusieurs romans dans une série, avec les mêmes personnages et des notes partagées, il est également possible de les conserver tous dans le même projet.

12.1 Créer un projet

De nouveaux projets peuvent être créés à partir du menu **Projet** en sélectionnant **Créer ou ouvrir un projet**. Cela ouvrira la boîte de dialogue **Bienvenue**, où vous pourrez sélectionner le bouton *Nouveau* qui vous aidera à créer un nouveau projet. Cette boîte de dialogue est également affichée lorsque vous démarrez novelWriter.

Un projet novelWriter nécessite un dossier dédié pour stocker ses fichiers sur le système de fichiers local. Si vous êtes intéressé par les détails, vous pouvez consulter le chapitre *Comment les données sont stockées*.

Une liste des projets récemment ouverts est maintenue et affichée dans la boîte de dialogue **Bienvenue**. Un projet peut être supprimé de cette liste en le sélectionnant et en appuyant sur la touche Suppr ou en faisant un clic droit sur le projet et en sélectionnant l'option **Supprimer le projet**.

Les paramètres spécifiques au projet sont disponibles dans **Réglages du projet** dans le menu **Projet**. Voir plus de détails ci-dessous dans la section *Paramètres du projet*.

Les détails du texte du roman du projet, y compris le nombre de mots et une table des matières avec le nombre de mots et de pages, sont disponibles dans la boîte de dialogue **Détails du roman**. Des statistiques sur le projet sont également disponibles dans l'outil **Compiler le manuscrit**.

12.1.1 Modèles de projets

Dans la boîte de dialogue de bienvenue, vous pouvez également créer un nouveau projet à partir d'un projet existant. Si vous souhaitez utiliser une structure spécifique pour tous vos nouveaux projets, vous pouvez créer un projet dédié qui servira de modèle et choisir de copier un projet existant à partir de l'option :guilabel: « Pré-remplir le projet » du formulaire **Nouveau projet**.



Figure1: La liste des projets (à gauche) et le formulaire de nouveau projet (à droite) de la boîte de dialogue Bienvenue.

12.2 Structure du projet

Les projets sont structurés en un ensemble de dossiers de premier niveau appelés « dossiers racines ». Ils sont visibles dans l'arborescence du projet à gauche de la fenêtre principale.

Les *documents du roman* vont dans un dossier racine de type **Roman**. Les *Notes de projet* vont dans les autres dossiers racines. Ces autres types de dossiers racines sont destinés à recevoir vos notes sur les différents éléments de votre histoire. Leur utilisation est bien sûr entièrement facultative.

Un nouveau projet peut ne pas contenir tous les dossiers racines, mais vous pouvez ajouter ceux que vous souhaitez à partir de la barre d'outils de l'arborescence du projet.

Chaque dossier racine est associé à un ou plusieurs *référence mot-clé* qui sont utilisés pour y faire référence à partir d'autres documents et notes. L'utilisation prévue de chaque type de dossier racine est indiquée ci-dessous. Cependant, à part le dossier **Roman**, aucune restriction n'est appliquée par l'application sur ce que vous mettez dans ces dossiers. Vous pouvez les utiliser comme bon vous semble.

Le système de dossiers racine est étroitement lié au fonctionnement du système de balises et de références. Pour plus de détails, voir le chapitre *Tags et références*.

12.2.1 Types de dossiers racines

Roman

Il s'agit du type de dossier racine pour le texte qui sera intégré au roman ou aux romans finaux. Cette classe de documents a d'autres règles et caractéristiques que les notes de projet. Voir *Structure du roman* pour plus de détails.

Intrigue

Il s'agit du type de dossier racine dans lequel les intrigues principales peuvent être décrites. Il est facultatif, mais l'ajout d'au moins quelques notes peut être utile pour étiqueter les éléments de l'intrigue pour la **Vue d'ensemble**. Les balises de ce dossier peuvent être référencées en utilisant le mot-clé *@plot*.

Personnages

Les notes des personnages sont placées dans ce type de dossier racine. Elles sont particulièrement importantes si vous voulez utiliser la **Vue d'ensemble** pour voir quel personnage apparaît à quel endroit, quelle partie de l'histoire est racontée du point de vue d'un personnage spécifique, ou pour vous concentrer sur l'intrigue d'un personnage particulier.

Les noms de personnages peuvent également être insérés, par exemple, dans les titres de chapitres lorsque vous créez votre manuscrit. Les tags de ce type de dossier peuvent être référencés en utilisant le mot-clé `@pov`` pour les personnages de point de vue, `@focus`` pour un personnage de focalisation, ou le mot-clé `@char`` pour n'importe quel autre personnage présent.

Lieux

Le type de dossier lieux est destiné aux différents lieux de la scène que vous souhaitez suivre. Les balises de ce dossier peuvent être référencées en utilisant le mot-clé `@location``.

Chronologie

Si l'histoire a plusieurs lignes temporelles ou des sauts dans le temps au sein d'une même intrigue, ce type de dossier peut être utilisé pour en assurer le suivi. Les tags de ce type de dossier peuvent être référencés en utilisant le mot-clé `@time``.

Objets

Les objets importants de l'histoire, par exemple les objets physiques qui changent souvent de mains, peuvent être suivis ici. Les étiquettes de ce type de dossier peuvent être référencées à l'aide du mot-clé `@object``.

Entités

Votre intrigue comporte-t-elle de nombreuses organisations ou entreprises puissantes ? Ou d'autres entités qui font partie de l'intrigue ? Elles peuvent être organisées ici. Les tags dans ce type de dossier peuvent être référencés en utilisant le mot-clé `@entity``.

Personnalisé

Le type de dossier racine personnalisé peut être utilisé pour suivre tout ce qui n'est pas couvert par les options ci-dessus. Les balises de ce type de dossier peuvent être référencées en utilisant le mot-clé `@custom``.

Les dossiers racines sont étroitement liés au système de balises et de références. Chaque type de dossier correspond à des catégories de balises qui peuvent être utilisées pour les référencer. Pour plus d'informations sur les balises listées, voir [Comment utiliser les références](#).

Il y a aussi un mot-clé `@mention` qui peut être utilisé pour référencer n'importe quel tag. Voir [Comment utiliser les références](#) pour plus de détails.

Note

Vous pouvez renommer les dossiers racine comme vous le souhaitez. Cependant, cela ne change pas le mot-clé de référence ni ce qu'il fait.

12.2.2 Documents supprimés

Les documents supprimés sont placés dans un dossier racine **Corbeille** spécial. Les documents du dossier corbeille peuvent alors être supprimés de façon permanente, soit individuellement, soit en vidant la corbeille à partir du menu. Les documents contenus dans la corbeille sont supprimés de l'*index du projet* et ne peuvent pas être référencés.

Un document ou un dossier peut être déplacé vers la corbeille à partir du menu **Projet**, ou en appuyant sur **Ctrl+Maj+Suppr**. Les dossiers racines ne peuvent être supprimés que lorsqu'ils sont vides.

12.2.3 Documents archivés

Si vous ne souhaitez pas supprimer un document ou le placer dans le dossier **Corbeille** où il risque d'être supprimé accidentellement, mais que vous voulez quand même le retirer de l'arborescence principale de votre projet, vous pouvez créer un dossier racine **Archives** et l'y déplacer. Cela a le même effet que de le déplacer dans la **Corbeille**, mais il ne risque pas d'être supprimé.

Vous pouvez faire glisser n'importe quel document dans ce dossier et préserver ses paramètres. Le document sera toujours exclu de l'outil **Compiler le manuscrit**. Il est également supprimé de l'index *index du projet*, de sorte que les balises et les références qui y sont définies n'apparaîtront nulle part ailleurs.

12.2.4 Utilisation des dossiers dans l'arborescence du projet

Les dossiers ordinaires, ceux qui ne sont pas des dossiers racines, n'ont aucune signification structurelle pour le projet. Lorsque novelWriter traite les documents d'un projet, comme par exemple lorsque vous créez un manuscrit à partir de celui-ci, ces dossiers sont ignorés. Seul l'ordre des documents a de l'importance.

Les dossiers sont là uniquement pour vous permettre d'organiser les documents en sections significatives et de pouvoir les réduire et les cacher dans l'arborescence du projet lorsque vous ne travaillez pas sur ces documents.

Outre l'utilisation de dossiers, il est également possible d'ajouter des documents enfants à d'autres documents. Ceci est particulièrement utile lorsque vous créez des chapitres et des scènes. Si vous ajoutez des documents de scène distincts, vous devez également ajouter des documents de chapitre distincts, même s'ils ne contiennent qu'un titre de chapitre. Vous pouvez ensuite ajouter des documents de scène en tant qu'éléments enfants aux chapitres.

12.2.5 Documents récupérés

Si novelWriter se plante ou se termine sans sauvegarder l'état du projet, ou si vous utilisez un outil de synchronisation de fichiers qui se désynchronise, il peut y avoir des fichiers dans le dossier de stockage du projet qui ne sont pas suivis dans le fichier de base du projet. Ces fichiers, lorsqu'ils sont découverts, sont récupérés et ajoutés au projet lorsqu'un projet est ouvert.

Les fichiers découverts sont analysés à la recherche de métadonnées qui donnent des indices sur l'emplacement antérieur du document dans le projet. La routine de chargement du projet essaiera de les replacer aussi près que possible de cet emplacement, s'il existe toujours. En général, le document sera ajouté à la fin du dossier où il se trouvait auparavant. Si ce dossier n'existe pas, il essaiera de l'ajouter au type de dossier racine correct. S'il ne parvient pas à déterminer quel est le bon dossier racine, le document sera ajouté au dossier racine **Nouveau**. Enfin, si un **Nouveau** n'existe pas, il en sera créé un.

Si le titre du document peut être récupéré, le mot « Récupéré : » sera ajouté comme préfixe pour indiquer qu'il peut nécessiter une attention supplémentaire. Si le titre ne peut être déterminé, le document sera nommé d'après sa clé interne, qui est une chaîne de caractères et de chiffres.

12.2.6 Fichier de verrouillage du projet

Pour éviter la perte de données causée par des conflits de fichiers lorsque les projets novelWriter sont synchronisés via des outils de synchronisation de fichiers, un fichier de verrouillage de projet est écrit dans le dossier de stockage du projet lorsqu'un projet est ouvert. Si vous essayez d'ouvrir un projet dans lequel un tel fichier est déjà présent, vous recevrez un avertissement et des informations sur l'endroit où novelWriter pense que le projet est également ouvert. Vous aurez la possibilité d'ignorer cet avertissement et de continuer à ouvrir le projet à vos risques et périls.

Note

Si, pour une raison quelconque, novelWriter ou votre ordinateur tombe en panne, le fichier de verrouillage peut subsister même si aucune autre instance ne maintient le projet ouvert. Dans ce cas, il est prudent d'ignorer l'avertissement relatif au fichier de verrouillage lors de la réouverture du projet.

Avertissement

Si vous choisissez d'ignorer l'avertissement et de continuer à ouvrir le projet, et que plusieurs instances du projet sont en fait ouvertes, vous risquez de provoquer des incohérences et de créer des fichiers de projet divergents, ce qui peut entraîner une perte de données et des fichiers orphelins. Il est peu probable que vous perdiez du texte à moins que les deux instances aient le même document ouvert dans l'éditeur, et novelWriter essaiera de résoudre les incohérences du projet la prochaine fois que vous l'ouvrirez.

12.3 Documents de projets

De nouveaux documents peuvent être créés à partir de la barre d'outils dans l'arborescence du projet, ou en appuyant sur **Ctrl+N**. Cela ouvrira le menu de création d'un nouvel élément et vous permettra de choisir parmi un certain nombre de documents et de dossiers prédéfinis. Il vous sera demandé de donner un nom au nouvel élément.

Vous pouvez toujours renommer un élément en sélectionnant **Renommer un élément** dans le menu **Projet**, ou en appuyant sur **F2** lorsqu'un document ou un dossier est sélectionné.

D'autres paramètres pour les documents et dossiers du projet sont disponibles dans le menu contextuel que vous pouvez activer en cliquant avec le bouton droit de la souris sur un élément de l'arborescence. Le sous-menu **Transformer** comprend des options pour convertir, diviser ou fusionner des documents. Voir *Séparation et fusion de documents* pour plus de détails sur ces deux dernières options.

12.3.1 Modèles de documents

Si vous souhaitez créer des documents modèles à utiliser lors de la création de nouveaux documents de projet, comme par exemple un modèle de note de caractère, vous pouvez ajouter un dossier racine **Modèles** à votre projet. Tout document ajouté à ce dossier racine apparaîtra dans le menu **Ajouter un élément** de la barre d'outils de l'arborescence du projet. Lorsqu'il est sélectionné, un nouveau document est créé dont le contenu est copié à partir du modèle choisi.

Ajouté dans la version 2.3.

12.3.2 Nombre de mots

Un décompte des caractères, des mots et des paragraphes est maintenu pour chaque document, ainsi que pour chaque section d'un document suivant un *titre*. Le nombre de mots et le changement de mots dans la session en cours sont affichés dans le pied de page de tout document ouvert dans l'éditeur, et toutes les statistiques sont affichées dans le panneau de détails sous l'arborescence du projet pour tout document sélectionné dans l'arborescence du projet ou des romans.

Le décompte des mots n'est pas mis à jour en temps réel, mais s'exécute en arrière-plan toutes les quelques secondes tant que le document est activement édité.

Le nombre total de mots du projet est affiché dans la barre d'état. Le décompte total dépend de la somme des valeurs de l'arborescence du projet, qui dépendent à leur tour d'un *index du projet* à jour. Si le décompte semble erroné, un recomptage complet des mots du projet peut être lancé en reconstruisant l'index du projet. Soit à partir du menu **Outils**, soit en appuyant sur F9.

Les règles de comptage sont décrites plus en détail dans *Décompte des mots et des textes*.

12.4 Paramètres du projet

Les **Paramètres du projet** sont accessibles depuis le menu **Projet**, ou en appuyant sur Ctrl+Maj+,. Cela ouvrira une boîte de dialogue, avec une série d'onglets.

12.4.1 Onglet Paramètres

L'onglet **Paramètres** contient les paramètres relatifs au nom du projet, à l'auteur et à la langue.

Le **Nom du projet** peut être modifié ici. Il est utilisé pour le titre de la fenêtre principale et pour la génération des fichiers de sauvegarde. Gardez donc à l'esprit que si vous modifiez ce paramètre, les noms des fichiers de sauvegarde changeront également.

Vous pouvez également modifier les paramètres **Auteurs** et **Langue du projet**. Ces paramètres ne sont utilisés que lors de la construction du manuscrit, pour certains formats. Le paramètre de langue est également utilisé lors de l'insertion de texte dans les documents dans l'afficheur, comme par exemple les étiquettes pour les mots-clés et les commentaires spéciaux.

Si votre projet est rédigé dans une langue différente de la langue de vérification orthographique principale, vous pouvez remplacer le paramètre par défaut ici. La langue du projet peut également être modifiée à partir du menu **Outils**.

Vous pouvez également remplacer le paramètre de sauvegarde automatique du projet si vous le souhaitez.

12.4.2 Onglets Statut et Importance

Chaque document ou dossier de type **Roman** peut recevoir une étiquette « Statut » accompagnée d'une icône colorée avec une forme optionnelle sélectionnée dans une liste de formes prédéfinies. Chaque document ou dossier des autres types peut recevoir une étiquette « Importance » avec les mêmes options de personnalisation.

Ces étiquettes sont là uniquement pour votre commodité, et vous n'êtes pas obligé de les utiliser pour que d'autres fonctionnalités fonctionnent. Aucune autre partie de novelWriter n'accède à ces informations. L'intention est de les utiliser pour indiquer à quel stade d'achèvement se trouve chaque document de roman, ou à quel point le contenu d'une note est important pour l'histoire. Vous n'êtes pas obligé de les utiliser de cette manière, c'est juste ce pour quoi elles ont été conçues, mais vous pouvez en faire ce que vous voulez.

Les étiquettes d'état et d'importance peuvent être exportées et importées afin que vous puissiez les partager entre les projets ou définir un ensemble standard pour tous vos projets d'écriture. Lorsque vous importez des étiquettes dans un projet, elles sont toujours ajoutées en tant que *nouvelles* étiquettes.

Voir aussi *Importance et statut du document*.

Note

Le statut ou le niveau d'importance actuellement utilisé par un ou plusieurs documents ne peut pas être supprimé, mais il peut être modifié.

12.4.3 Onglet Remplacement automatique

Un ensemble de mots-clés de remplacement automatiquement peut être ajouté dans cet onglet. Les mots-clés de la colonne de gauche seront remplacés par le texte de la colonne de droite lorsque les documents seront ouverts dans l'afficheur. Ils seront également appliqués aux constructions de manuscrits.

La fonction de remplacement automatique remplace le texte entre crochets qui se trouve dans cette liste. Le surligneur syntaxique ajoutera une couleur alternative au texte correspondant à la syntaxe, mais il ne vérifie pas si le texte se trouve dans cette liste.

Note

Un mot-clé ne peut pas contenir d'espaces. Les crochets sont ajoutés par défaut et, lorsqu'ils sont utilisés dans le texte, ils font partie du mot-clé à remplacer. Cela permet de s'assurer que certaines parties du texte ne sont pas remplacées involontairement par le contenu de la liste.

12.5 Sauvegarde

Un système de sauvegarde automatique est intégré à novelWriter. Pour l'utiliser, un chemin d'accès à l'endroit où les fichiers de sauvegarde doivent être stockés doit être indiqué dans les **Préférences**. Le chemin d'accès est par défaut un dossier nommé « Backups » dans votre répertoire personnel.

Les sauvegardes peuvent être exécutées automatiquement lorsqu'un projet est fermé, ce qui implique également qu'elles sont exécutées lorsque l'application elle-même est fermée. Les sauvegardes sont des fichiers zip datés des fichiers du projet dans le dossier du projet (les fichiers qui ne font pas strictement partie du projet sont ignorés). Les archives zip sont stockées dans un sous-dossier du chemin de sauvegarde. Le sous-dossier aura le même nom que le **nom du projet** tel que défini dans *Paramètres du projet*.

La fonction de sauvegarde, lorsqu'elle est configurée, peut également être exécutée manuellement à partir du menu **Outils**. Il est également possible de désactiver les sauvegardes automatiques pour un projet donné dans **Paramètres du projet**.

Note

Pour que la sauvegarde puisse être exécutée, le **Nom du projet** doit être défini dans **Paramètres du projet**. Cette valeur est utilisée pour générer le nom et le chemin des sauvegardes. Sans cette valeur, la sauvegarde ne s'exécutera pas du tout, mais elle produira un message d'avertissement.

12.6 Statistiques d'écriture

Lorsque vous travaillez sur un projet, un fichier journal enregistre le moment où vous l'avez ouvert, le moment où vous l'avez fermé, et le nombre total de mots de vos documents et notes à la fin de la session, à condition que la session ait duré plus de 5 minutes, ou que le nombre total de mots ait changé. Pour plus de détails sur le fichier journal, voir *Comment les données sont stockées*.

Un outil permettant de visualiser le contenu du fichier journal est disponible dans le menu **Outils** sous **Statistiques d'écriture**. Vous pouvez également le lancer en appuyant sur F6, ou le trouver dans la barre latérale.

L'outil affichera une liste de toutes vos sessions, ainsi qu'un ensemble de filtres à appliquer aux données. Vous pouvez également exporter les données filtrées vers un fichier JSON ou vers un fichier CSV qui

peut être ouvert par un tableur comme par exemple Libre Office Calc ou Excel.

Ajouté dans la version 1.2: À partir de la version 1.2, le fichier journal enregistre également le temps d'inactivité de la session. La définition de l'inactivité ici est que la fenêtre principale de novelWriter perd le focus, ou que l'utilisateur n'a pas fait de changements au document actuellement ouvert depuis cinq minutes. Le nombre de minutes peut être modifié dans les **Préférences**.

12.6.1 Chronomètre de session

Un chronomètre de session est visible par défaut dans la barre d'état. L'icône affiche une horloge lorsque vous êtes actif et une icône de pause lorsque vous êtes considéré comme « inactif » selon les critères mentionnés ci-dessus.

Si vous ne souhaitez pas voir le chronomètre, vous pouvez cliquer une fois dessus pour le masquer. L'icône restera visible. Cliquez à nouveau sur l'icône pour afficher à nouveau le chronomètre.

Ajouté dans la version 2.6: Depuis la version 2.6, un clic sur le texte ou l'icône du chronomètre dans la barre d'état permet de basculer sa visibilité.

CHAPTER 13

Structure du roman

Ce chapitre traite de la structure d'un projet de roman.

Il existe deux types de documents dans un projet, les **Documents de roman** et les **Notes de projet**. Les documents inédits actifs ne peuvent se trouver que dans un dossier racine de type **Nouveau**. Vous pouvez également les déplacer vers **Archives** et **Corbeille** bien sûr, où ils deviennent inactifs.

L'arborescence du projet peut distinguer les différents niveaux d'en-tête des nouveaux documents à l'aide d'icônes colorées, et éventuellement ajouter de l'emphase sur l'étiquette. L'accentuation peut être activée dans les **Préférences**.

13.1 Importance des titres

Les sous-dossiers des dossiers racines n'ont aucun impact sur la structure du roman lui-même. La structure est plutôt dictée par le niveau des titres des documents.

Quatre niveaux de titres sont supportés, indiqués par le nombre de signes (#) précédant le titre. Voir aussi la section *Mise en forme du texte* pour plus de détails sur la syntaxe du balisage.

Note

Les niveaux de titres ne sont pas seulement importants lors de la génération du manuscrit, ils sont également utilisés par l'indexeur lors de la construction de l'arborescence dans la **Vue d'ensemble** ainsi que dans l'**Arborescence du roman**. Chaque rubrique commence également une nouvelle région où de nouvelles étiquettes et références peuvent être définies. Voir *Tags et références* pour plus de détails.

La syntaxe des quatre types de titres de base, et des trois types spéciaux, est listée dans la section *En-têtes*. La signification des quatre niveaux pour la structure de votre roman est la suivante :

Titre de niveau 1 : Section

Ce niveau de titre signifie que le texte se réfère à un titre de niveau supérieur. Il est utile lorsque vous souhaitez diviser le manuscrit en livres, parties ou actes. Ces titres ne sont pas obligatoires. Le titre du roman lui-même doit utiliser le niveau de titre spécial #! couvert dans *En-têtes*.

Niveau de niveau 2 : Chapitre

Ce niveau de titre correspond à un chapitre. Chaque fois que vous voulez commencer un nouveau chapitre, vous devez ajouter un tel titre. Si vous choisissez de diviser votre manuscrit en un document par scène, vous avez besoin d'un seul document de chapitre avec seulement l'en-tête. Vous pouvez bien entendu ajouter un synopsis et des mots-clés de référence au document de chapitre. Si vous souhaitez ouvrir le chapitre par une citation ou un autre texte d'introduction qui ne fait pas partie d'une scène, c'est également dans ce document que vous placerez ce texte.

Niveau de titre 3 : Scène

Ce niveau de titre correspond à une scène. Vous devez fournir un texte de titre, mais celui-ci peut être remplacé par un séparateur de scène ou simplement ignoré lors de la construction de votre manuscrit. Si vous avez besoin de faire la distinction entre une scène dure et une scène douce, il existe un format alternatif pour les scènes que vous pouvez utiliser pour faire cette distinction. Le formatage est couvert dans *En-têtes*. Voir aussi *Ruptures de scène dures et douces*.

Niveau de titre 4 : Section

Ce niveau d'en-tête peut être utilisé pour diviser une scène, généralement appelée « section » dans la documentation et l'interface utilisateur. Ces sections peuvent être utiles si vous souhaitez changer de référence au milieu de la scène, par exemple si vous changez de point de vue. Vous êtes libre d'utiliser les sections comme vous le souhaitez, et vous pouvez les filtrer dans le manuscrit final.

Les sauts de page peuvent être automatiquement ajoutés avant les titres, les parties, les chapitres et les scènes à partir de l'outil **Compiler le manuscrit** lorsque vous créez votre projet dans un format qui prend en charge les sauts de page. Si vous voulez des sauts de page à d'autres endroits, vous devez les spécifier manuellement. Voir *Espace vertical et sauts de page*.

Astuce

Il existe de nombreuses options pour traiter les titres de roman lors de la construction du manuscrit. Par exemple, les numéros de chapitre peuvent être appliqués automatiquement, de même que les numéros de scène si vous le souhaitez dans un projet de manuscrit. Vous pouvez également insérer des noms de personnages dans les titres de chapitres. Voir la page *Compiler le manuscrit* pour plus de détails.

Note

Depuis la version 2.6, les niveaux d'en-tête internes à novelWriter ne correspondent pas directement aux niveaux d'en-tête des documents manuscrits. Dans les documents manuscrits, les chapitres sont considérés comme l'en-tête de premier niveau, et les partitions deviennent des paragraphes de texte simple avec une police plus grande.

Ajouté dans la version 2.6.

13.1.1 Titre du roman et page de garde

Il est recommandé d'ajouter un document au premier niveau de chaque dossier racine **Nouveau** avec le titre du roman. Vous devez modifier le code de format de l'en-tête de niveau 1 avec un ! afin de le rendre comme un titre de document qui est exclu de toute table des matières automatique dans un document de construction de manuscrit.

Vous pouvez également ajouter le nom et l'adresse de l'auteur au-dessus de ce titre si le format de

manuscrit que vous utilisez l'exige, et ajouter un espace supplémentaire avant le titre.

Il s'agit de la page de titre que novelWriter génère automatiquement pour un nouveau projet à partir de la version 2.6 :

```
Jane Doe[br]
Address 1[br]
Address 2 <<

[vspace:5]

#! My Novel

>> **By Jane Doe** <<

>> Word Count: [field:textWords] <<
```

Le titre est par défaut centré sur la page. Vous pouvez ajouter plus de texte à la page si vous le souhaitez, comme par exemple le nom de l'auteur et ses coordonnées.

Si vous voulez une page de texte supplémentaire après la page de titre, commençant sur une nouvelle page, vous pouvez ajouter [new page] sur une ligne à part, et continuer le texte après elle. Cela insérera un saut de page avant le texte. Voir aussi *Espace vertical et sauts de page*.

13.1.2 Titres des chapitres non numérotés

Si vous utilisez la numérotation automatique pour vos chapitres, mais que vous voulez garder certains chapitres spéciaux séparés, vous pouvez ajouter un ! au code de formatage des titres de niveau 2 pour indiquer à l'outil de construction d'ignorer ces chapitres lors de l'ajout des numéros.

```
##! Unnumbered Chapter Title
```

```
Chapter Text
```

L'outil **Compiler le manuscrit** dispose également d'une fonction de formatage distincte pour ces titres de chapitre. Voir la page *Compiler le manuscrit* pour plus de détails. Lors de la construction d'un document d'un format qui supporte les sauts de page, les chapitres non numérotés peuvent également avoir un saut de page ajouté comme pour les chapitres normaux.

13.1.3 Ruptures de scène dures et douces

Si vous avez besoin de deux façons différentes de styliser les scènes dans votre manuscrit, par exemple si vous voulez insérer des séparateurs de scène différents pour les ruptures de scène douces et dures, il existe un format de scène alternatif disponible pour les titres de scène avec un ! ajouté au code de formatage.

```
### Soft Scene Transition
```

```
A soft scene break.
```

```
###! Hard Scene Transition
```

```
A hard scene break.
```

Il existe une fonction de formatage distincte pour ces titres dans l'outil **Compiler le manuscrit**.

Ajouté dans la version 2.4.

CHAPTER 14

Tags et références

Dans novelWriter, il n'y a pas de formulaires ou de tableaux à remplir pour définir les personnages, les lieux ou d'autres éléments de votre histoire. Au lieu de cela, vous créez des *notes de projet* que vous pouvez marquer comme représentant ces éléments de l'histoire en créant un *tag*. Lorsque vous souhaitez lier un élément de votre histoire à une note définissant un élément de l'histoire, comme un personnage, vous créez un *référence* qui renvoie à cette étiquette. Vous pouvez également créer des liens croisés entre les notes de votre projet de la même manière.

C'est peut-être l'une des caractéristiques qui différencie novelWriter d'autres applications similaires. Il n'est donc pas toujours évident pour les nouveaux utilisateurs de comprendre comment cela est censé fonctionner, c'est pourquoi ce chapitre vise à expliquer plus en détail comment utiliser ce système de balises et de références.

Astuce

Si le système de balises et de références vous semble difficile à suivre à la seule lecture de ce chapitre, vous pouvez créer un nouveau projet dans le formulaire Nouveau projet de la boîte de dialogue **Bienvenue** et sélectionner « Créer un projet d'exemple » dans l'option « Pré-remplir le projet ». Le projet d'exemple contient plusieurs exemples de balises et de références.

14.1 Métadonnées dans novelWriter

La structure de votre projet novelWriter est déduite des *en-têtes* des documents, et non des documents eux-mêmes. Voir *Importance des titres* pour plus de détails. Par conséquent, les métadonnées sont également associées aux en-têtes, et non aux documents directement.

Si vous divisez votre projet en documents distincts pour chaque scène, cette distinction peut ne pas avoir d'importance. Cependant, l'utilisation de documents à une échelle structurelle plus large présente plusieurs avantages lorsque vous démarrez votre projet. Par exemple, il peut être plus judicieux de définir toutes les scènes, et même les chapitres, dans un seul document au départ, ou peut-être un document par acte. Vous pouvez par la suite diviser ces documents en utilisant la fonction de division de document. Voir *Séparation et fusion de documents* pour plus de détails.

Vous pouvez faire de même avec vos notes. Vous pouvez traiter chaque en-tête comme un élément indépendant de vos notes qui peut être référencé ailleurs. Ainsi, vous pouvez rassembler tous vos personnages mineurs ou de second plan dans un seul fichier de notes, tout en étant en mesure de les référencer individuellement en les séparant par des en-têtes et en leur attribuant à chacun une balise. Vous pouvez également présenter votre intrigue dans un seul document et attribuer une balise à chaque étape de l'intrigue dans des rubriques distinctes.

14.2 Comment utiliser les tags

Un « tag » dans novelWriter est un mot ou une phrase que vous définissez comme appartenant à une rubrique. Les balises sont définies à l'aide de la commande `@tag` *mot-clé*.

Le format de base d'une balise est `@tag: NomDuTag`.

Un autre format de balise est `@tag: NomDuTag | NomAffiché`.

NomDuTag (Requis)

Il s'agit d'un identifiant unique de votre choix. C'est la valeur que vous utiliserez plus tard pour faire référence à ce document ou à une section du document.

NomAffiché (Optionnel)

Il s'agit d'un nom d'affichage facultatif utilisé pour la balise. Lorsque vous construisez votre manuscrit, vous pouvez par exemple insérer le nom du personnage du point de vue directement dans les titres des chapitres. Par défaut, la valeur `NomDuTag` est utilisée dans ces titres, mais si vous utilisez un format abrégé en interne dans votre projet, vous pouvez utiliser ceci pour spécifier un format plus approprié pour les titres de votre manuscrit.

Vous ne pouvez définir qu'une seule balise** par rubrique, et la balise doit être unique dans **tous** les documents du projet.

Une fois qu'une balise a été définie, elle peut être référencée dans de nouveaux documents ou faire l'objet de références croisées dans d'autres notes. Les balises apparaîtront également dans la **Vue en plan** et dans le panneau des références sous le visualiseur de documents lorsqu'un document est ouvert dans le visualiseur.

Le surligneur syntaxique vous indiquera que le mot-clé est correctement utilisé et que la balise est autorisée, c'est-à-dire qu'elle est unique. Les balises dupliquées devraient être détectées si l'index est à jour. Une balise non valide est signalée par une ligne verte et ne reçoit pas la même couleur que les balises valides.

La balise est la seule partie de ces notes que novelWriter utilise. Le reste du contenu du document est là pour que vous puissiez l'utiliser comme bon vous semble. Bien entendu, le contenu des documents peut être ajouté au manuscrit ou à un document de synthèse. Si vous souhaitez compiler un seul document de toutes vos notes, vous pouvez le faire à partir de l'outil **Compiler le manuscrit**.

Ajouté dans la version 2.2: Les balises ne sont plus sensibles à la casse. Les balises sont affichées par défaut avec la majuscule utilisée lors de la définition de la balise, mais vous n'êtes pas obligé d'utiliser la même majuscule lorsque vous y faites référence ultérieurement.

Ajouté dans la version 2.3: Les balises peuvent avoir un nom d'affichage facultatif pour les constructions de manuscrits.

Ajouté dans la version 2.6: Vous pouvez désormais ajouter des balises aux documents du roman. Celles-ci peuvent être utilisées pour les références croisées entre les chapitres et les scènes, ainsi qu'à partir des notes si vous le souhaitez.

Exemple d'un titre avec une tag pour un personnage de l'histoire :


```
# Character: Jane Doe
```

```
@tag: Jane | Jane Doe
```

```
Some information about the character Jane Doe.
```

Lorsque cela est fait dans un document dans un *Dossier racine* de type « Personnages », la balise est automatiquement traitée comme un caractère disponible dans votre projet avec la valeur " Jane », et vous pourrez y faire référence dans n'importe lequel de vos autres documents en utilisant les mots-clés de référence pour les personnages. Il apparaîtra également dans l'onglet Personnage du panneau Référence sous le visualiseur de document, et dans le menu d'auto-complétion de référence dans l'éditeur lorsque vous remplissez des références. Voir *Visualisation d'un document* et *L'auto-complétion des références*.

C'est le type de dossier racine qui définit la catégorie d'éléments de l'histoire sous laquelle la balise est indexée. Voir la section *Structure du projet* pour une vue d'ensemble des types de dossiers racines disponibles. Ils sont également abordés dans la section suivante.

14.3 Comment utiliser les références

Chaque en-tête de n'importe quel niveau de votre projet peut contenir des références à des balises définies dans les notes du projet. Ces références sont rassemblées par l'indexeur et utilisées pour générer la **Vue d'ensemble**, entre autres choses.

Les références sont définies sous la forme d'un *mot-clé* et d'une liste de balises correspondantes. Les mots-clés valides sont listés ci-dessous. Le format d'une ligne de référence est @keyword: value1, [value2] ... [valeurN]. Tous les mots-clés de référence autorisent des valeurs multiples.

@pov

Le personnage de point de vue pour la section en cours. La cible doit être une balise note dans un dossier racine de type **Personnages**.

@focus

Le personnage qui est au centre de l'attention pour la section en cours. Ceci peut être utilisé dans les cas où le focus n'est pas un personnage de point de vue. La cible doit être une balise note dans un dossier racine de type **Personnages**.

@char

Autres personnages de la section courante. La cible doit être une étiquette de note dans un dossier racine de type **Personnages**. Cela ne doit pas inclure le point de vue ou le personnage principal si ces références sont utilisées.

@plot

L'intrigue ou la sous-intrigue développée dans la section en cours. La cible doit être une balise note dans un dossier racine de type **Intrigue**.

@time

Les lignes de temps concernées par la section en cours. La cible doit être une balise note dans un dossier racine de type **Chronologie**.

@location

L'endroit où se déroule la section en cours. La cible doit être une balise note dans un dossier racine de type **lieux**.

@object

Objets présents dans la section courante. La cible doit être une étiquette de note dans un dossier

racine de type **Objets**.

@entity

Entités présentes dans la section courante. La cible doit être une balise note dans un dossier racine de type **Entités**.

@custom

Références personnalisées dans la section en cours. La cible doit être une étiquette de note dans un dossier racine de type **personnalisé**. Les dossiers personnalisés sont destinés à toute autre catégorie de notes que vous souhaiteriez utiliser.

@mention

Tout ce qui est mentionné, mais qui n'est pas présent dans la section actuelle. Il est destiné aux cas où vous révélez des détails sur un personnage ou un lieu dans une scène sans qu'il en fasse partie. Cela peut être utile pour vérifier la cohérence ultérieurement. N'importe quelle balise dans n'importe quel dossier de notes racine peut être listée sous mentions.

@story

Cette fonction est utilisée pour référencer un document de roman, comme une scène ou un chapitre, à partir d'un autre endroit de votre projet.

L'outil de mise en évidence de la syntaxe avertit l'utilisateur que les balises et les références sont utilisées correctement et que les balises référencées existent.

Note

Le surligneur peut se tromper si l'index des balises définies n'est pas à jour. Si c'est le cas, appuyez sur F9 pour le régénérer, ou sélectionnez **Reconstruire l'index** dans le menu **Outils**. En général, l'index d'un document est régénéré lorsqu'il est sauvegardé, ce qui ne devrait pas être nécessaire.

Astuce

Si vous ajoutez une référence dans l'éditeur à une balise qui n'existe pas encore, vous pouvez cliquer dessus avec le bouton droit de la souris et sélectionner **Créer une note pour la balise**. Cela générera automatiquement une nouvelle note de projet avec la nouvelle balise définie. Pour que cela soit possible, un dossier racine pour cette catégorie de références doit déjà exister.

Une note peut également faire référence à une autre note, de la même manière que les documents de romans. Lorsque la note est ouverte dans l'éditeur de documents, les références deviennent des liens cliquables, ce qui facilite le suivi des connexions dans l'intrigue. Vous pouvez suivre les liens dans l'éditeur de documents en les cliquant avec la souris tout en maintenant la touche Ctrl enfoncée. Les liens cliqués sont toujours ouverts dans le panneau de visualisation.

Les notes de projet n'apparaissent pas dans la **Vue d'ensemble**, de sorte que les références entre les notes n'ont de sens que si vous voulez pouvoir naviguer d'une note à l'autre en cliquant, ou bien sûr si vous voulez simplement mettre en évidence que deux notes sont liées.

Astuce

Si vous faites des références croisées entre les notes et que vous exportez votre projet en tant que document HTML à l'aide de l'outil **Compiler le manuscrit**, les références croisées deviennent

également des liens cliquables dans le document HTML exporté.

Exemple de document de roman avec références aux personnages et aux intrigues :

Chapter 1

@pov: Jane

Scene 1

@char: John, Sam

@plot: Main

Once upon a time ...

14.3.1 L'auto-complétion des références

Un menu contextuel d'auto-complétion s'affiche automatiquement dans l'éditeur de document lorsque vous tapez le caractère @ sur une nouvelle ligne. Il vous proposera d'abord des mots-clés de balise ou de référence à ajouter, puis, après l'ajout de :, des références à partir de la liste des balises que vous avez déjà définies.

Vous pouvez utiliser l'auto-complétion pour ajouter plusieurs références avec un , entre elles, et même en taper de nouvelles. De nouvelles références peuvent être créées en cliquant dessus avec le bouton droit de la souris et en sélectionnant **Créer une note pour le tag** dans le menu.

Ajouté dans la version 2.2.

CHAPTER 15

Compiler le manuscrit

Vous pouvez à tout moment compiler un manuscrit, une ébauche de vos notes ou tout autre type de document à partir du texte de votre projet. Tout cela est géré par l'outil **Compiler le manuscrit**. Vous pouvez l'activer à partir de la barre latérale, du menu **Outils**, ou en appuyant sur F5.

Ajouté dans la version 2.1: Cet outil est nouveau pour la version 2.1. Un outil plus simple était utilisé pour les versions précédentes. Cet outil plus simple ne vous permettait de définir qu'un seul ensemble d'options pour la compilation, mais il offrait à peu près les mêmes fonctionnalités.

15.1 L'outil de compilation de manuscrit

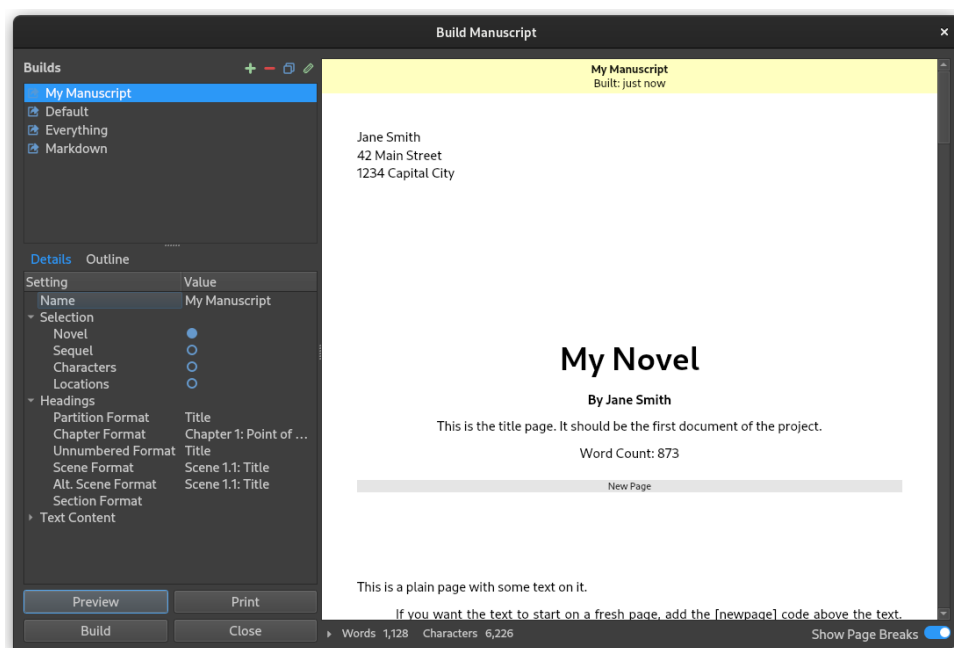


Figure1: La fenêtre principale de l'outil **Compiler le manuscrit**.

La fenêtre principale de l'outil **Compiler le manuscrit** contient une liste de toutes les structures que vous

avez définies, une sélection de paramètres et quelques boutons permettant de générer un aperçu, d'ouvrir la boîte de dialogue d'impression ou d'exécuter la compilation pour créer un document manuscrit.

15.1.1 Plan et nombre de mots

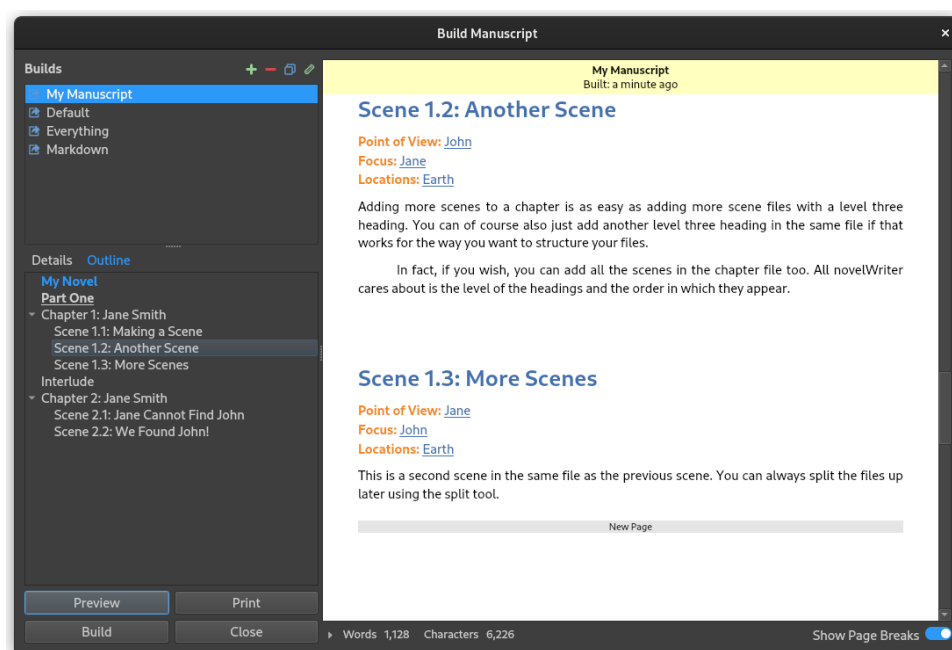


Figure2: La fenêtre principale de l'outil **Compiler le manuscrit** avec le **Plan** visible.

L'onglet **Plan** sur la gauche vous permet de naviguer dans les rubriques du document de prévisualisation. Il affiche les titres de niveau scène pour les documents de roman, et les titres de niveau 2 pour les notes.

Un panneau escamotable indiquant le nombre de mots et de caractères est également disponible sous le document de prévisualisation. Ces chiffres sont calculés à partir du texte que vous avez inclus dans le document et sont plus précis que ceux disponibles dans l'arborescence du projet puisqu'ils sont comptés *après le formatage*.

Pour une description détaillée de la façon dont ils sont comptés, voir *Décompte des mots et des textes*.

15.2 Paramètres de compilation

Chaque définition de construction peut être éditée en l'ouvrant dans la boîte de dialogue **Paramètres de construction du manuscrit**, soit en double-cliquant, soit en la sélectionnant et en appuyant sur le bouton d'édition dans la barre d'outils.

Astuce

Vous pouvez garder la boîte de dialogue **Paramètres de construction du manuscrit** ouverte pendant que vous testez les différentes options, et simplement cliquer sur le bouton *Appliquer*. Vous pouvez tester le résultat de vos réglages en appuyant sur le bouton *Aperçu* dans la fenêtre principale **Compiler le manuscrit**. Lorsque vous êtes satisfait du résultat, vous pouvez fermer les paramètres.

15.2.1 Sélection des documents

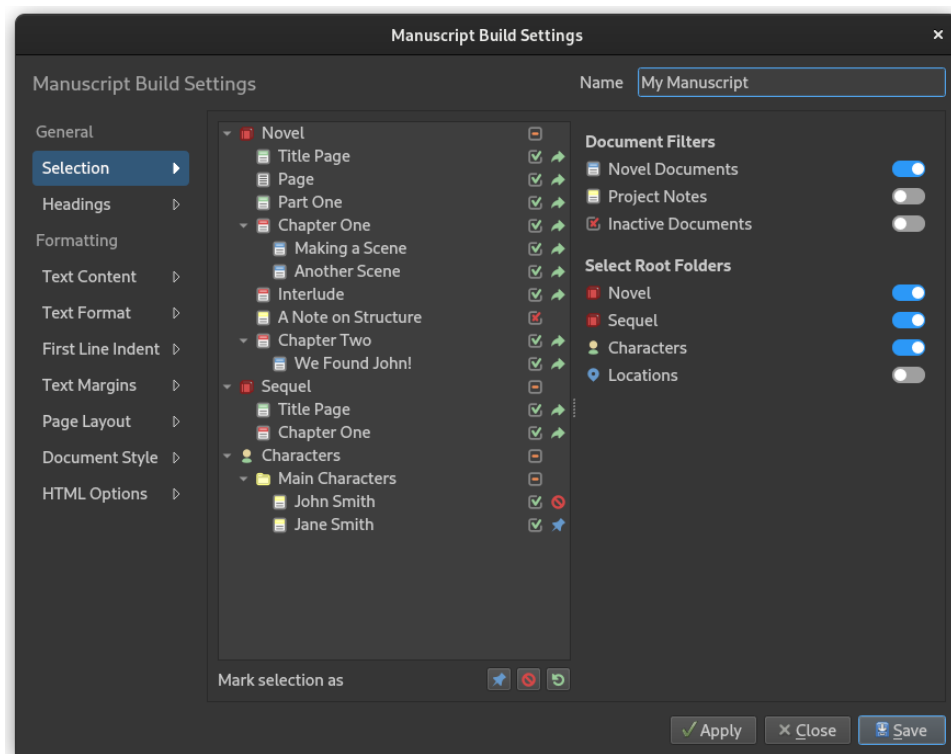


Figure3: La page **Sélection** de la boîte de dialogue **Paramètres de construction du manuscrit**.

La page **Sélection** de la boîte de dialogue **Paramètres de construction du manuscrit** vous permet d'affiner les documents inclus dans la construction. Ils sont indiqués par une icône de flèche verte dans la dernière colonne. Sur la droite, vous disposez d'options de filtrage pour sélectionner le contenu d'un type spécifique, et d'un ensemble de commutateurs pour les dossiers racine à inclure.

Vous pouvez remplacer le résultat de ces filtres en marquant un ou plusieurs documents et en choisissant de les inclure ou de les exclure explicitement à l'aide des boutons situés sous l'arborescence. Le dernier bouton peut être utilisé pour réinitialiser l'annulation et redonner le contrôle aux paramètres du filtre.

Dans la figure, l'icône de la flèche verte et l'icône de l'épingle bleue indiquent les documents inclus, et l'icône de l'interdiction rouge indique qu'un document est explicitement exclu.

15.2.2 Mise en forme des en-têtes

La page **En-têtes** de la boîte de dialogue **Paramètres de construction du manuscrit** vous permet de définir le format des titres de vos *Documents du roman*. Par défaut, le titre est simplement copié tel quel, comme l'indique le format `{Title}`. Vous pouvez modifier ce format pour, par exemple, ajouter des numéros de chapitre et de scène, ou insérer des noms de personnages, comme indiqué dans la figure ci-dessus.

Cliquer sur le bouton d'édition à côté d'un format copiera la chaîne de formatage dans la boîte d'édition où elle peut être modifiée, et où un surligneur de syntaxe aidera à indiquer quelles parties sont automatiquement générées par l'outil de construction. Le bouton *Insérer* est une liste déroulante de ces formats, et en sélectionner un l'insérera à la position du curseur.

Tout texte que vous ajoutez et qui n'est pas surligné en couleur restera dans vos titres formatés. `{Titre}` sera toujours remplacé par le texte de l'en-tête de vos documents.

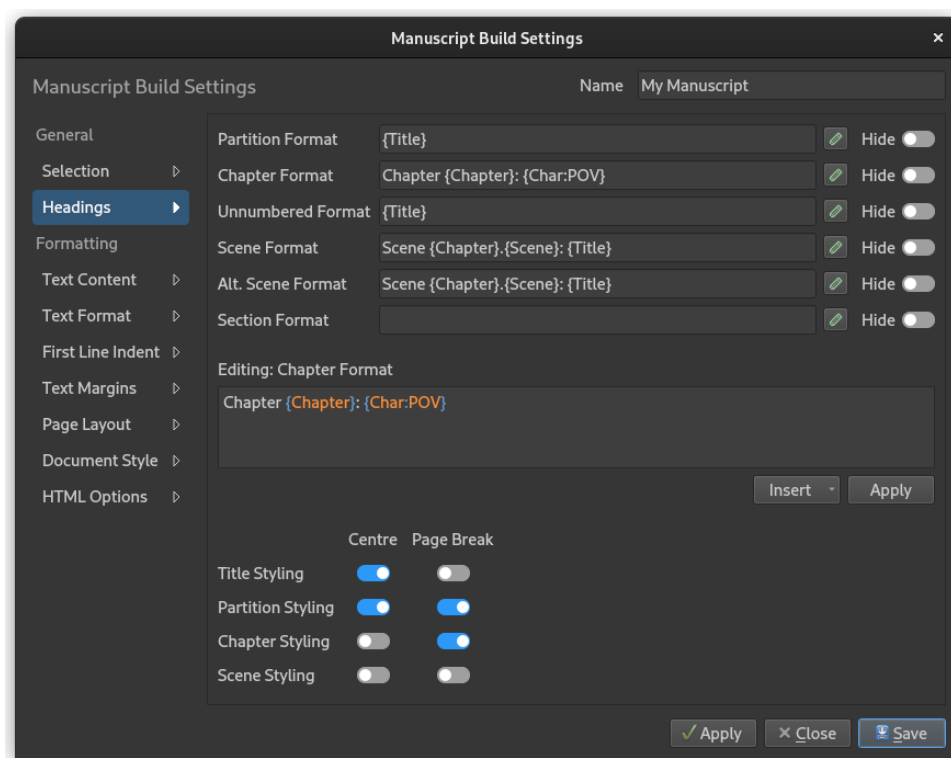


Figure4: La page **En-têtes** de la boîte de dialogue **Paramètres de construction du manuscrit**.

Vous pouvez prévisualiser le résultat de ces chaînes de formatage en cliquant sur *appliquer*, puis en cliquant sur *aperçu* dans la fenêtre principale de l'outil **Compiler le manuscrit**.

Séparateurs de scènes

Si vous ne voulez pas de titres pour vos scènes (ou pour vos sections si vous en avez), vous pouvez laisser les cases de mise en forme vides. Dans ce cas, un paragraphe vide sera inséré entre les scènes ou les sections, ce qui créera un vide dans le texte. Vous pouvez également activer le paramètre *Cacher*, qui les ignorera complètement. En d'autres termes, il n'y aura même pas d'espace supplémentaire inséré.

Alternativement, si vous voulez un texte séparateur entre eux, comme le commun * * *, vous pouvez entrer le texte séparateur désiré comme format. Si le format est un morceau de texte statique, il sera toujours traité comme un séparateur.

Scènes dures et douces

Si vous souhaitez faire la distinction entre les ruptures de scène douces et dures, vous pouvez utiliser le format alternatif de l'en-tête de scène dans votre texte. Vous pouvez ensuite donner à ces titres un format différent dans les paramètres **En-têtes**.

Voir *En-têtes* pour plus d'informations sur la façon de formater les titres dans votre texte.

15.2.3 Paramètres de sortie

Les pages **Contenu**, **Format** et **Sortie** de la boîte de dialogue **Paramètres de construction du manuscrit** contrôlent un certain nombre d'autres paramètres pour la sortie. Certains d'entre eux ne s'appliquent qu'à des formats de sortie spécifiques, ce qui est indiqué par les titres de section sur les pages de paramètres.

15.3 Création des manuscrits

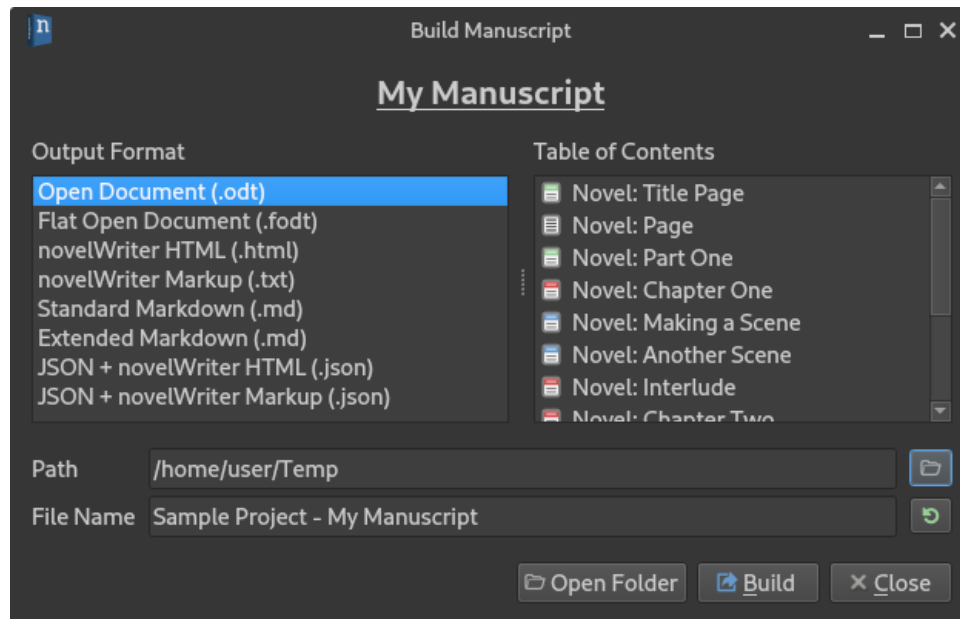


Figure5: Le dialogue **compiler le manuscrit** est utilisé pour créer les manuscrits proprement dits.

Lorsque vous appuyez sur le bouton *Construire* dans la fenêtre principale de l'outil **Compiler le manuscrit**, une boîte de dialogue spéciale s'ouvre. C'est là que vous choisissez le format de sortie désiré et l'endroit où écrire le fichier.

La partie gauche de la boîte de dialogue contient une liste de tous les formats de fichiers disponibles, et la partie droite, une liste des documents qui sont inclus en fonction de la définition de la construction que vous avez sélectionnée. Vous pouvez choisir un chemin de sortie et définir un nom de fichier de base. L'extension du fichier sera ajoutée automatiquement.

Pour générer le document manuscrit, appuyez sur le bouton *Construire*. Une petite barre de progression indique l'avancement de la construction, mais pour les petits projets, elle peut passer très vite.

15.3.1 Formats de fichiers

Actuellement, quatre formats de documents sont pris en charge.

Open Document Format

L'outil de construction peut produire soit un fichier `.odt`, soit un fichier `.fodt`. Ce dernier n'est qu'une version plate du format du document sous la forme d'un fichier XML unique. La plupart des éditeurs de texte riche supportent le premier, et seulement quelques uns le second.

Format de document Microsoft Word

Le format Microsoft Word Document permet d'écrire un seul fichier `.docx`. Il utilise un format assez basique qui devrait être compatible avec la plupart des éditeurs de texte riche.

Format de document portable (PDF)

Le PDF est généré à partir d'une copie du document de prévisualisation et devrait avoir les mêmes capacités de formatage que la prévisualisation. Il est identique à celui qui est produit lorsque vous sélectionnez l'option d'impression et que vous imprimez au format PDF.

novelWriter HTML

Le format HTML écrit un seul fichier `.htm` avec un formatage de style minimal. Le document

HTML peut être traité par des outils de conversion de documents tels que [Pandoc](#), importé dans un traitement de texte ou imprimé à partir d'un navigateur.

Markdown standard/étendu

Le format Markdown existe en version standard et en version étendue. La *seule* différence en termes de fonctionnalité de novelWriter est la prise en charge du texte barré, qui n'est pas prise en charge par la version standard.

novelWriter Markup

Il s'agit simplement d'une concaténation des documents du projet sélectionnés par les filtres dans un fichier `.txt`. Les documents sont empilés dans l'ordre où ils apparaissent dans l'arborescence du projet, avec les commentaires, les balises, etc. inclus s'ils sont sélectionnés. C'est un format utile pour exporter le projet et le réimporter plus tard dans novelWriter.

Ajouté dans la version 2.6: Les options de sortie Microsoft Word et PDF ont été ajoutées.

15.3.2 Formats supplémentaires

En plus des formats de document ci-dessus, les formats HTML et Markup de novelWriter peuvent également être enveloppés dans un fichier JSON. Ces fichiers auront une entrée pour les métadonnées et une entrée pour le corps du texte. Pour le HTML, les styles CSS utilisés par la prévisualisation sont également inclus.

Le corps du texte est enregistré dans une liste à deux niveaux. La liste extérieure contient une entrée par document, dans l'ordre où ils apparaissent dans l'arborescence du projet. Chaque document est ensuite divisé en une liste, avec une entrée par paragraphe qu'il contient.

Ces fichiers sont principalement destinés au post-traitement par script pour ceux qui souhaitent cette option. Un fichier JSON peut être importé directement dans un objet dict de Python ou dans un tableau de PHP, pour ne citer que quelques options.

15.4 Impression

Le bouton *Imprimer* vous permet d'imprimer le contenu de la fenêtre de prévisualisation. Vous pouvez soit imprimer sur l'une des imprimantes de votre système, soit sélectionner le format PDF à partir de l'icône de l'imprimante dans la boîte de dialogue d'impression.

Note

Dans tous les cas, le format de papier doit être défini par défaut, quel que soit le format par défaut de votre système. Si vous souhaitez le modifier, vous devez le sélectionner dans la boîte de dialogue **Aperçu avant impression**.

CHAPTER 16

Story Structure Comments

As of version 2.7, story structure annotations using the %story style of comment is supported. To use the feature, make the first word of a comment story, followed by a period, a structure term, a colon, a space and the text for that term.

Here's an example:

```
%story.term: text
```

Ajouté dans la version 2.7.

16.1 Usage

The story term can be anything that you want to track in the manuscript. This construct is intended to make it easier to extract metadata from a work to perform a structural analysis of the story.

There are probably as many ways to examine story structure as there are authors and editors combined. For this reason the story tag is flexible. You can use any terms you want and track any aspect of the story that serves your purposes.

An example method has been advanced by Shawn Coyne in *The Story Grid*. This method asserts that a story is composed of « beats », and that each beat has an inciting incident, a complication, a crisis, and a resolution. One might capture these elements of a beat where a character overcomes their fear of giving a speech as:

```
### Scene
```

```
%Synopsis: Carol overcomes her fear of giving a speech.
```

```
%Story.incite: Carol is pleased to be invited to a conference to see her boss.
↳ deliver a keynote.
```

```
%Story.complication: Carol's boss calls in sick and asks her to deliver a big
↳ speech.
```

```
%Story.crisis: Carol has a fear of appearing on stage.
```

(suite sur la page suivante)

(suite de la page précédente)

```
%Story.resolution: Carol engages the help of a coach who helps her overcome
↳her fears and delivers a great speech.
```

Other analytical models propose tracking a scene's pace, how it affects the mood of the story, or which element(s) of the story's genre are being satisfied. An author can use this mechanism to track any element of a scene. Some examples include time of day, how much time passes in the scene, or even the physical form of a shape-shifting character. If a story involves magic, one could track which wand a main character has in hand. It's up to the author.

When the story and other scene metadata is extracted into a tabular form, it is possible to get a comprehensive overview of the story and to identify possible issues. For example, so many fast-paced scenes without a break that readers might become fatigued or over-stimulated.

16.2 Output

The story structure comments can be included in the manuscript, and are formatted similarly to the synopsis comments:

Scene 1.2: Scene

Synopsis: Carol overcomes her fear of giving a speech.

Story Structure (Incite): Carol is pleased to be invited to a conference to see her boss deliver a keynote.

Story Structure (Complication): Carol's boss calls in sick and asks her to deliver a big speech.

Story Structure (Crisis): Carol has a fear of appearing on stage.

Story Structure (Resolution): Carol engages the help of a coach who helps her overcome her fears and delivers a great speech.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus diam nibh, tincidunt et quam at, fringilla malesuada risus. Nullam eleifend, sem nec varius tincidunt, urna mi varius dolor, sit amet gravida risus eros at purus. Etiam vehicula hendrerit elit, sit amet pulvinar dolor viverra sed. Curabitur metus ex, gravida at sodales sed, tristique eget diam. Suspendisse ultricies lorem sed ullamcorper rutrum. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis tempus magna mi, sit amet sagittis arcu viverra ut.

Figure1: A set of story structure comments as shown in the Manuscript tool.

When you export your project data from the Outline View, all story structure terms are added as columns to the exported file, which can then be opened in the spread sheet software of your choice.

Vous pouvez personnaliser novelWriter de plusieurs façons. Actuellement, vous pouvez ajouter de nouveaux thèmes d'interface graphique, vos propres thèmes de syntaxe et installer des dictionnaires supplémentaires.

17.1 Dictionnaires de vérification orthographique

novelWriter utilise [Enchant](#) comme outil de vérification orthographique. Selon votre système d'exploitation, il peut ou non charger automatiquement tous les dictionnaires de vérification orthographique installés.

17.1.1 Linux et MacOS

Sous Linux et MacOS, il vous suffit généralement d'installer les dictionnaires hunspell, aspell ou myspell sur votre système, comme vous le faites pour d'autres applications. Consultez la documentation de votre distro ou de votre système d'exploitation pour savoir comment procéder. Ces dictionnaires devraient apparaître comme des langues de vérification orthographique disponibles dans novelWriter.

17.1.2 Windows

Pour Windows, l'anglais est inclus dans l'installation. Pour les autres langues, vous devez télécharger et ajouter les dictionnaires vous-même.

Outil d'installation

Un petit outil pour vous aider se trouve sous **Outils > Ajouter des dictionnaires**. Il importera des dictionnaires de vérification orthographique à partir des extensions Free Office ou Libre Office. Les dictionnaires sont alors installés dans l'emplacement d'installation de la bibliothèque Enchant et devraient donc fonctionner pour toute application qui utilise Enchant pour la vérification orthographique.

Installation manuelle

Si vous préférez le faire manuellement ou si vous voulez utiliser une source différente de celles mentionnées ci-dessus, vous devez obtenir des fichiers de dictionnaires compatibles avec votre langue.

Vous avez besoin de deux fichiers se terminant par `.aff` et `.dic`. Ces fichiers doivent ensuite être copiés à l'emplacement suivant :

`C:\Users\<USER>\AppData\Local\enchant\hunspell`

Cela suppose que votre profil d'utilisateur est stocké dans `C:\Users\<USER>`. Le dernier ou les deux derniers dossiers peuvent ne pas exister, vous devrez donc les créer.

Vous pouvez trouver les différents dictionnaires sur le site web [Free Desktop](#).

Note

Le lien [Free Desktop](#) pointe vers un référentiel, et ce qui peut ressembler à des liens de fichiers à l'intérieur du dossier du dictionnaire sont en fait des liens vers des pages web. Si vous cliquez avec le bouton droit de la souris et que vous les téléchargez, vous obtiendrez des fichiers HTML, pas des dictionnaires !

Pour télécharger les fichiers du dictionnaire, cliquez avec le bouton droit de la souris sur l'étiquette « plain » à la fin de chaque ligne et téléchargez-la.

17.2 Syntaxe et thèmes de l'interface graphique

L'ajout de vos propres thèmes d'interface graphique et de syntaxe est relativement facile, bien qu'il soit nécessaire d'éditer manuellement les fichiers de configuration avec les valeurs de couleur. Les thèmes sont définis par de simples fichiers de configuration en texte brut contenant des métadonnées et des paramètres de couleur.

Pour créer vos propres versions, copiez d'abord l'un des fichiers existants sur votre ordinateur local et modifiez-le à votre guise.

- Les thèmes syntaxiques existants sont stockés dans `novelwriter/assets/syntax`.
- Les thèmes GUI existants sont stockés dans `novelwriter/assets/themes`.
- Les thèmes d'icônes existants sont stockés dans `novelwriter/assets/icons`.

N'oubliez pas de changer également le nom de votre thème en modifiant le paramètre `name` en haut du fichier, sinon vous ne pourrez pas les distinguer dans les **Préférences**.

Pour que novelWriter puisse localiser les fichiers de thèmes personnalisés, vous devez les copier dans l'emplacement *Données d'application* dans votre espace personnel ou utilisateur. Il devrait y avoir un dossier nommé `syntax` pour les thèmes de syntaxe, juste `themes` pour les thèmes d'interface graphique, et `icons` pour les thèmes d'icônes. Ces dossiers sont créés la première fois que vous démarrez novelWriter.

Une fois les fichiers copiés, ils devraient apparaître dans les **Préférences** avec le nom que vous avez défini dans le fichier.

Ajouté dans la version 2.0: La valeur `icontheme` a été ajoutée aux thèmes GUI. Assurez-vous de définir cette valeur dans les thèmes personnalisés existants. Sinon, novelWriter essaiera de deviner votre thème d'icônes, et ne choisira peut-être pas le plus approprié.

17.2.1 Thème d'interface graphique et d'icônes personnalisé

Le fichier `.conf` d'un thème GUI est constitué des paramètres suivants :

```
[Main]
name          = My Custom Theme
description   = A description of my custom theme
author       = Jane Doe
credit       = John Doe
url          = https://example.com
license      = CC BY-SA 4.0
licenseurl   = https://creativecommons.org/licenses/by-sa/4.0/
icontheme    = typicons_light

[Palette]
window        = 100, 100, 100
windowtext    = 100, 100, 100
base          = 100, 100, 100
alternatebase = 100, 100, 100
text          = 100, 100, 100
tooltipbase   = 100, 100, 100
tooltiptext   = 100, 100, 100
button        = 100, 100, 100
buttontext    = 100, 100, 100
brighttext    = 100, 100, 100
highlight     = 100, 100, 100
highlightedtext = 100, 100, 100
link          = 100, 100, 100
linkvisited   = 100, 100, 100

[GUI]
helptext      = 0, 0, 0
fadedtext     = 128, 128, 128
errortext     = 255, 0, 0
statusnone    = 120, 120, 120
statussaved   = 2, 133, 37
statusunsaved = 200, 15, 39
```

Dans la section `Main`, vous devez au moins définir les paramètres `name` et `icontheme`. Le paramètre `icontheme` doit correspondre à l'un des thèmes d'icônes internes, soit `typicons_light` ou `typicons_dark`, soit à un thème d'icônes dans votre répertoire d'icônes personnalisé. Le paramètre doit correspondre au nom du dossier du thème d'icônes.

Les valeurs de la palette correspondent aux valeurs de l'enum Qt pour `QPalette::ColorRole`, voir la documentation Qt <<https://doc.qt.io/qt-6/qpalette.html#ColorRole-enum>> pour plus de détails. Les valeurs de couleur sont des nombres RVB au format `r, g, b` où chacun est un entier de 0 à 255. Les valeurs omises ne sont pas chargées et utilisent les valeurs par défaut. Si la couleur `helptext` n'est pas définie, elle est calculée comme une couleur comprise entre les couleurs `window` et `windowtext`.

Ajouté dans la version 2.5: Les entrées de couleur des thèmes `fadedtext` et `errortext` ont été ajoutées.

17.2.2 Syntaxe des thèmes personnalisés

Un fichier `.conf` de thème est constitué des paramètres suivants :

```
[Main]
name      = My Syntax Theme
author    = Jane Doe
credit    = John Doe
url       = https://example.com
license   = CC BY-SA 4.0
licenseurl = https://creativecommons.org/licenses/by-sa/4.0/

[Syntax]
background    = 255, 255, 255
text          = 0, 0, 0
link          = 0, 0, 0
headertext    = 0, 0, 0
headertag     = 0, 0, 0
emphasis      = 0, 0, 0
dialog        = 0, 0, 0
altdialog     = 0, 0, 0
note          = 0, 0, 0
hidden        = 0, 0, 0
shortcode     = 0, 0, 0
keyword       = 0, 0, 0
tag           = 0, 0, 0
value         = 0, 0, 0
optional      = 0, 0, 0
spellcheckline = 0, 0, 0
errorline     = 0, 0, 0
replacetag    = 0, 0, 0
modifier      = 0, 0, 0
texthighlight = 255, 255, 255, 128
```

Dans la section `Main`, vous devez définir au moins le paramètre `name`. Les valeurs de couleur de la syntaxe sont des nombres RVB(A) du format `r, g, b, a` où chacun est un entier de 0 à 255. La quatrième valeur est le canal alpha, qui peut être omis.

Les couleurs omises dans la syntaxe prennent par défaut la couleur noire, à l'exception de `background` qui prend par défaut la couleur blanche, et de `texthighlight` qui prend par défaut la couleur blanche avec une demi-transparence.

Ajouté dans la version 2.2: Le paramètre de couleur `shortcode` a été ajouté.

Ajouté dans la version 2.3: Le paramètre de couleur `optional` a été ajouté.

Ajouté dans la version 2.4: Le paramètre de couleur `texthighlight` a été ajouté.

Ajouté dans la version 2.5: Les paramètres de couleur `dialog`, `altdialog`, `note` et `tag` ont été ajoutés. Les paramètres `guillemets droits`, `guillemets doubles` et `guillemets simples` ont été supprimés.

CHAPTER 18

Modifications du format des projet

La plupart des modifications apportées aux formats de fichiers au cours de l'histoire de novelWriter n'ont pas d'impact sur l'utilisateur. Les fichiers de projet sont généralement mis à jour automatiquement. Cependant, certains changements nécessitent des actions mineures de la part de l'utilisateur.

Les principales modifications apportées aux formats sont énumérées dans ce chapitre, ainsi que les actions requises de la part de l'utilisateur, le cas échéant.

A full project file format specification is available under « More Documents ».

Prudence

Lorsque vous mettez à jour un projet d'une version de format à la suivante, le projet ne peut plus être ouvert par une version de novelWriter antérieure à la version où le nouveau format de fichier a été introduit. Vous recevrez une notification concernant toute mise à jour du format de fichier de votre projet et vous aurez la possibilité de refuser la mise à jour.

18.1 Format 1.5 Modifications

Ce format de projet a été introduit dans la version 2.0 RC 2 de novelWriter.

Il s'agit d'une modification du format 1.4. Il rend le XML plus cohérent en ce sens que les métadonnées ont été déplacées vers leurs nœuds de section respectifs en tant qu'attributs, et que les paramètres clé/valeur ont désormais un format cohérent. Les indicateurs logiques sont sauvegardés sous forme de oui/non au lieu de True/False de Python, et le titre principal du document est désormais sauvegardé dans l'article plutôt que dans l'index. La conversion est effectuée automatiquement lors du premier chargement d'un projet. Aucune action de l'utilisateur n'est nécessaire.

18.2 Format 1.4 Modifications

Ce format de projet a été introduit dans la version 2.0 RC 1 de novelWriter. Comme il s'agissait d'une version candidate, il est peu probable que votre projet l'utilise, mais c'est peut-être le cas si vous avez installé une préversion.

Ce format modifie la manière dont les éléments du projet (dossiers, documents et notes) sont stockés. Il s'agit d'un format plus compact, plus simple et plus rapide à analyser, et plus facile à étendre. La conversion se fait automatiquement lors du premier chargement d'un projet. Aucune action de l'utilisateur n'est nécessaire.

18.3 Format 1.3 Modifications

Ce format de projet a été introduit dans la version 1.5 de novelWriter.

Avec ce format, le nombre de présentations de documents a été réduit de huit à deux. La conversion des présentations de documents s'effectue automatiquement lors de l'ouverture du projet.

En raison de la réduction des présentations, certaines caractéristiques qui étaient auparavant contrôlées par ces présentations seront perdues. Ces fonctionnalités sont désormais contrôlées par des codes syntaxiques. Pour les récupérer, l'utilisateur doit donc apporter quelques modifications mineures aux documents sélectionnés.

Les modifications manuelles que l'utilisateur doit apporter sont très peu nombreuses car elles s'appliquent à des mises en page de documents qui ne doivent être utilisées qu'à quelques endroits dans un projet donné. Ces modifications sont les suivantes :

Pages de titres

- Le formatage du titre de niveau 1 sur la page de titre doit être modifié de `# Texte du titre` à `# ! Texte du titre` afin de conserver la fonctionnalité précédente. Voir *En-têtes*.
- Tout texte qui était précédemment centré sur la page doit être centré manuellement en utilisant la fonction d'alignement du texte. Voir *Alignement et indentation des paragraphes*.

Chapitres non numérotés

- La mise en page spécifique pour les chapitres non numérotés ayant été abandonnée, ces chapitres doivent tous utiliser le code de formatage `# ! Nom de chapitre` au lieu de `## Nom de chapitre`. Cela inclut également les chapitres marqués d'un astérisque : `## *Nom de chapitre`, car cette fonctionnalité a également été abandonnée. Voir *En-têtes*.

Pages vierges

- La mise en page appelée « Page vierge » a également été supprimée. La seule caractéristique de cette mise en page était qu'elle garantissait que le contenu commençait toujours sur une nouvelle page. Dans le nouveau format, les pages fraîches peuvent être placées n'importe où dans le texte avec le code `[NEW PAGE]`. Voir *Espace vertical et sauts de page*.

18.4 Format 1.2 Modifications

Ce format de projet a été introduit dans la version 0.10 de novelWriter.

Avec ce format, la façon dont les entrées de remplacement automatique étaient stockées dans le fichier XML du projet principal a changé.

18.5 Format 1.1 Modifications

Ce format de projet a été introduit dans la version 0.7 de novelWriter.

Avec ce format, le dossier `content` a été introduit dans le stockage du projet. Auparavant, tous les documents de novelWriter étaient sauvegardés dans une série de dossiers numérotés de `data_0` à `data_f`.

Il réduit également le nombre de métadonnées et de fichiers de cache. Ces fichiers sont automatiquement supprimés lorsqu'un ancien projet est ouvert. C'est également à cette époque que le fichier de table des matières a été introduit.

18.6 Format 1.0 Modifications

Il s'agit du format de fichier et de la structure de projet d'origine. Il a été utilisé jusqu'à la version 0.6.3.

Décompte des mots et des textes

Il s'agit d'une vue d'ensemble de la manière dont les mots et autres décomptes de votre texte sont effectués. Les règles de comptage devraient être relativement standard, et sont comparables aux règles de Libre Office Writer.

Les décomptes fournis dans l'application sur le texte brut sont approximatifs. Pour des décomptes plus précis, vous devez compiler votre manuscrit avec **Outil manuscrit** et vérifier les décomptes sur l'aperçu généré.

19.1 Décompte des mots du texte et statistiques

Il s'agit des règles relatives aux principaux décomptes disponibles pour chaque document d'un projet.

Pour tous les décomptes, les règles suivantes s'appliquent.

1. Les tirets courts (-) et longs (—) sont considérés comme des séparateurs de mots.
2. Toute ligne commençant par % ou @ est ignorée.
3. Les espaces blancs de fin de ligne sont ignorés, y compris les sauts de ligne.
4. Les > de début et les < de fin sont ignorés, de même que les espaces qui les accompagnent.
5. Les shortcodes valides et les autres commandes entre crochets [] sont ignorés.
6. La syntaxe Markdown en ligne dans les paragraphes de texte est traitée comme faisant partie du texte.

Après la préparation du texte, les chiffres suivants sont disponibles.

Décompte des caractères

Le nombre de caractères est la somme des caractères par ligne, y compris les caractères d'en-tête et les caractères d'espacement dans le texte, mais pas les caractères d'espacement de fin. Les Shortcodes dans le texte ne sont pas inclus, mais les codes Markdown le sont. Seuls les en-têtes et le texte sont pris en compte.

Décompte des mots

Le nombre de mots est la somme des blocs de caractères continus par ligne, séparés par un nombre quelconque de caractères d'espacement ou de tirets. Seuls les en-têtes et le texte sont comptés.

Décompte des paragraphes

Le nombre de paragraphes est le nombre de blocs de texte séparés par une ou plusieurs lignes vides. Une ligne constituée uniquement d'espaces blancs est considérée comme vide.

19.2 Décompte du manuscrit

Voici les règles pour les décomptes disponibles pour un manuscrit dans l'outil **Outil Manuscrit**. Les règles ont été ajustées pour correspondre à LibreOffice Writer, mais varieront légèrement en fonction du contenu de votre texte. LibreOffice Writer compte également le texte dans l'en-tête de la page, ce que ne fait pas l'outil **Outil Manuscrit**.

Le contenu de chaque ligne est compté après le traitement de toutes les mises en forme, de sorte que le résultat sera plus précis que les décomptes pour les documents textuels dans d'autres parties de l'application. Les règles suivantes s'appliquent :

1. Les tirets courts (-) et longs (—) sont considérés comme des séparateurs de mots.
2. Les espaces blancs de début et de fin sont généralement inclus, mais les sauts de paragraphe ne le sont pas.
3. Les retours à la ligne dans un paragraphe sont considérés comme des caractères d'espacement.
4. Tous les codes de formatage sont ignorés, y compris les shortcodes, les commandes et Markdown.
5. Les séparateurs de scène et de section sont pris en compte.
6. Les lignes de commentaires et de métadonnées sont comptées après leur mise en forme.
7. Les en-têtes sont comptés après avoir été formatés avec des formats personnalisés.

Les décomptes suivants sont disponibles :

En-têtes

Le nombre d'en-têtes dans le manuscrit.

Paragraphes

Le nombre de paragraphes dans le manuscrit.

Mots

Le nombre de mots du manuscrit, y compris les commentaires et les métadonnées.

Mots dans le texte

Nombre de mots dans les paragraphes, à l'exclusion de tout autre texte.

Mots dans les en-têtes

Le nombre de mots dans les en-têtes, y compris le formatage inséré comme les numéros de chapitre, etc.

Caractères

Le nombre de caractères de toutes les lignes, y compris les commentaires et le texte des métadonnées. Les sauts de paragraphe ne sont pas comptés, mais les sauts de ligne dans les paragraphes le sont.

Caractères dans le texte

Nombre de caractères dans les paragraphes. Les sauts de paragraphe ne sont pas comptés, mais les sauts de ligne dans les paragraphes le sont.

Caractères dans les en-têtes

Le nombre de caractères dans les en-têtes.

Caractères dans le texte, sans les espaces

Le nombre de caractères dans les paragraphes considérés comme faisant partie d'un mot ou d'une ponctuation. En d'autres termes, les caractères d'espacement ne sont pas pris en compte.

Caractères dans les en-têtes, sans les espaces

Le nombre de caractères dans les en-têtes considérés comme faisant partie d'un mot ou de la ponctuation. En d'autres termes, les caractères d'espacement ne sont pas pris en compte.

Emplacement des fichiers

novelWriter créera quelques fichiers sur votre système en dehors du dossier de l'application elle-même. L'emplacement de ces fichiers est décrit dans ce chapitre.

20.1 Configuration

La configuration générale de novelWriter, y compris tout ce qui se trouve dans les **Préférences**, est enregistrée dans un fichier de configuration central. L'emplacement de ce fichier dépend de votre système d'exploitation. Les chemins d'accès au système sont fournis par la classe Qt `QStandardPaths` et sa valeur `ConfigLocation`.

Les chemins standard sont les suivants :

- Linux : `~/.config/novelwriter/novelwriter.conf`
- MacOS: `~/Library/Preferences/novelwriter/novelwriter.conf`
- Windows: `C:\Users\<USER>\AppData\Local\novelwriter\novelwriter.conf`

Ici, `~` correspond au répertoire personnel de l'utilisateur sous Linux et MacOS, et `<USER>` est le nom d'utilisateur de l'utilisateur sous Windows.

Note

Il s'agit des emplacements standard définis par le système d'exploitation. Si votre système a été configuré différemment, ces emplacements peuvent également être différents.

20.2 Données d'application

novelWriter stocke également un certain nombre de données générées par les actions de l'utilisateur. Cela inclut la liste des projets récents de la boîte de dialogue **Bienvenue**. Les thèmes personnalisés devraient également être sauvegardés ici. Les chemins d'accès au système sont fournis par la classe Qt `QStandardPaths` et sa valeur `AppDataLocation`.

Les chemins standard sont les suivants :

- Linux: ~/.local/share/novelwriter/
- MacOS: ~/Library/Application Support/novelwriter/
- Windows: C:\Users\<USER>\AppData\Roaming\novelwriter\

Ici, ~ correspond au répertoire personnel de l'utilisateur sous Linux et MacOS, et <USER> est le nom d'utilisateur de l'utilisateur sous Windows.

Note

Il s'agit des emplacements standard définis par le système d'exploitation. Si votre système a été configuré différemment, ces emplacements peuvent également être différents.

L'emplacement des données d'application contient également plusieurs dossiers :

cache

Ce dossier est utilisé pour enregistrer les données de prévisualisation pour l'outil **Compilation de manuscrit**.

icons, syntax and themes

Ces dossiers sont vides par défaut, mais c'est là que l'utilisateur peut stocker des fichiers de thèmes personnalisés. Voir *Personnalisations* pour plus de détails.

Comment les données sont stockées

Ce chapitre contient des détails sur la manière dont novelWriter stocke et gère les données du projet.

21.1 Structure du projet

Tous les fichiers novelWriter sont écrits avec l'encodage utf-8. Comme Python convertit automatiquement les fins de ligne Unix en fins de ligne Windows sur les systèmes Windows, novelWriter ne fait aucune adaptation au formatage sur les systèmes Windows. Ceci est entièrement géré par la bibliothèque standard de Python. Python s'en occupe également lorsqu'il travaille sur les mêmes fichiers sur les systèmes d'exploitation Windows et Unix.

21.1.1 Fichier principal du projet

Le projet lui-même nécessite un dossier dédié pour stocker ses fichiers, où novelWriter créera son propre « système de fichiers » où la hiérarchie des dossiers et des fichiers du projet est décrite dans un fichier XML du projet. C'est le fichier principal du projet dans le dossier racine du projet avec le nom `nwProject.nwx`. Ce fichier contient également toutes les métadonnées requises pour le projet (à l'exception des données d'index), ainsi qu'un certain nombre de paramètres liés au projet.

Si ce fichier est perdu ou corrompu, la structure du projet est perdue, mais pas le texte lui-même. Il est important de sauvegarder ce fichier, soit à l'aide de l'outil de sauvegarde intégré, soit à l'aide de votre propre solution de sauvegarde.

Le fichier XML du projet est formaté en indentation et convient aux outils de comparaison et de contrôle de version, car la majeure partie du fichier reste statique, bien qu'un horodatage soit défini dans la section méta à la ligne 2 et que diverses entrées de données méta soient incrémentées à chaque enregistrement.

A full project file format specification is available under « More Documents ».

21.2 Documents de projet

Tous les documents du projet sont enregistrés dans un sous-dossier du dossier principal du projet nommé **content**. Chaque document possède un identifiant de fichier basé sur un nombre aléatoire de 52 bits, représenté sous la forme d'une chaîne hexadécimale. Les documents sont enregistrés avec un nom de fichier assemblé à partir de cet identifiant et de l'extension `.nwd`.

Si vous souhaitez trouver l'emplacement du système de fichiers d'un document dans le projet, vous pouvez soit le rechercher dans le fichier XML du projet, soit sélectionner **Afficher les détails du fichier** dans le menu **Document** lorsque le document est ouvert dans l'éditeur, soit rechercher dans le fichier `ToC.txt` à la racine du dossier du projet. Le fichier `ToC.txt` contient une liste de tous les documents du projet, référencés par leur étiquette, et l'endroit où ils sont sauvegardés.

La raison de ce nom de fichier cryptique est d'éviter les problèmes liés aux conventions de dénomination des fichiers et aux restrictions sur les différents systèmes d'exploitation, et aussi d'avoir un nom de fichier qui ne dépend pas du nom que vous donnez au document dans le projet, ou des modifications que vous lui apportez. Ceci est particulièrement utile lors de l'utilisation d'un système de gestion des versions.

Chaque fichier de document contient une version en texte brut du texte de l'éditeur. Le fichier peut en principe être édité dans n'importe quel éditeur de texte, et convient pour le diffing et le contrôle de version si on le souhaite. Assurez-vous simplement que le fichier reste en encodage utf-8, sinon les caractères unicode risquent d'être altérés lorsque le fichier est ouvert dans novelWriter.

Il est généralement déconseillé de modifier ces fichiers. La raison en est que l'index ne sera pas automatiquement mis à jour, ce qui signifie que novelWriter ne sait pas que vous avez modifié le fichier. Si vous modifiez un fichier de cette manière, vous devez reconstruire l'index lorsque vous ouvrez le projet dans novelWriter.

Les premières lignes du fichier peuvent contenir des métadonnées commençant par les caractères `%%~`. Ces lignes sont principalement là pour restaurer certaines informations si le fichier est perdu dans le fichier principal du projet, et ces informations peuvent être utiles si vous ouvrez le fichier dans un éditeur externe car elles contiennent l'étiquette du document ainsi que la classe et la mise en page du document. Les lignes peuvent être supprimées sans conséquences pour le reste du contenu du fichier, et seront ajoutées la prochaine fois que le document sera sauvegardé dans novelWriter.

21.2.1 Le processus d'enregistrement des fichiers

Lors de l'enregistrement du fichier de projet ou de l'un des documents, les données sont d'abord enregistrées dans un fichier temporaire. En cas de succès, l'ancien fichier de données est alors supprimé et remplacé par le fichier temporaire. Cela garantit que les données précédemment enregistrées ne sont remplacées que lorsque les nouvelles données ont été enregistrées avec succès sur le support de stockage.

21.3 Métadonnées du projet

Le dossier du projet contient un sous-dossier nommé **meta**, contenant un certain nombre de fichiers. Le dossier meta contient des fichiers semi-importants. C'est-à-dire qu'ils peuvent être perdus avec un impact mineur sur le projet. Tous les fichiers de ce dossier sont des fichiers JSON ou JSON Lines, bien que certains autres fichiers puissent provenir de versions antérieures de novelWriter, car ils n'ont pas tous été des fichiers JSON dans le passé.

Si vous utilisez un logiciel de contrôle de version pour votre projet, vous pouvez exclure ce dossier, même si vous souhaitez suivre le fichier journal de la session et la liste des mots personnalisés.

21.3.1 L'index des projets

Entre les sessions d'écriture, l'index du projet est sauvegardé dans un fichier JSON dans `meta/index.json`. Ce fichier n'est pas critique. S'il est perdu, il peut être complètement reconstruit à partir de novelWriter dans le menu **Outils**.

L'index est maintenu et mis à jour chaque fois qu'un document ou une note est enregistré dans l'éditeur. Il contient toutes les références et les balises des documents et des notes, ainsi que l'emplacement de tous les en-têtes dans le projet et le nombre de mots dans chaque section de l'en-tête.

L'intégrité de l'index est vérifiée lorsque le fichier est chargé. Il est possible de corrompre l'index si le fichier est édité et manipulé manuellement, c'est pourquoi cette vérification est importante pour éviter les plantages soudains de novelWriter. Si le fichier contient des erreurs, novelWriter le reconstruira automatiquement. Si la vérification échoue d'une manière ou d'une autre et que novelWriter continue à se bloquer, vous pouvez supprimer le fichier manuellement et reconstruire l'index. Si cela échoue également, vous avez probablement rencontré un bogue.

21.3.2 Définitions des compilations

Les définitions de construction de l'outil **Compilation du manuscrit** sont conservées dans le fichier `meta/builds.json`. Si ce fichier est perdu, toutes les définitions de construction personnalisées sont également perdues.

21.3.3 Options de l'interface graphique mises en cache

Un fichier nommé `meta/options.json` contient le dernier état des différents boutons de l'interface graphique, des commutateurs, de la taille des fenêtres de dialogue, de la taille des colonnes, etc, de l'interface graphique. Ce sont les paramètres de l'interface graphique qui sont spécifiques au projet. Les paramètres globaux de l'interface graphique sont stockés dans le fichier de configuration principal.

Le fichier n'est pas critique, mais s'il est perdu, toutes les options de l'interface graphique reviendront à leurs paramètres par défaut.

21.3.4 Liste des mots personnalisés

Un fichier nommé `meta/userdict.json` contient tous les mots personnalisés que vous avez ajoutés au projet à des fins de vérification orthographique. Le contenu de ce fichier peut être édité à partir du menu **Outils**. Si vous perdez ce fichier, tous vos mots personnalisés pour la vérification orthographique seront également perdus.

21.3.5 Statistiques de la session

La progression de l'écriture est sauvegardée dans le fichier `meta/sessions.jsonl`. Ce fichier enregistre la durée et le nombre de mots de chaque session d'écriture sur le projet donné. Ce fichier est utilisé par l'outil **Statistiques d'écriture**. Si ce fichier est perdu, l'historique qu'il contient est également perdu, mais cela n'a par ailleurs aucun impact sur le projet.

Chaque session est enregistrée sous la forme d'un objet JSON sur une seule ligne du fichier. Chaque enregistrement de session est ajouté au fichier.

Executer depuis la source

Ce chapitre décrit les différentes façons d'exécuter novelWriter directement à partir du code source, et comment construire les différents composants tels que les fichiers de traduction et la documentation.

Note

Le texte ci-dessous suppose que la commande `python` correspond à un exécutable Python 3. Python 2 est maintenant déprécié, mais sur de nombreux systèmes, la commande `python3` peut être utilisée à la place. De même, `pip` peut être remplacé par `pip3`.

La plupart des commandes personnalisées pour construire des paquets de novelWriter, ou pour construire des actifs, sont contenues dans le script `pkgutils.py` à la racine du code source. Vous pouvez lister les commandes disponibles en lançant :

```
python pkgutils.py help
```

22.1 Dépendances

novelWriter a été conçu pour s'appuyer sur le moins de dépendances possible. Seul le wrapper Python pour les bibliothèques Qt GUI est nécessaire. Le packaging pour la vérification orthographique est optionnel, mais recommandé. Tout le reste est géré par les bibliothèques Python standard.

Les paquets Python suivants sont nécessaires pour exécuter toutes les fonctionnalités de novelWriter :

- PyQt6 - nécessaire pour se connecter aux bibliothèques Qt6.
- PyEnchant - nécessaire pour la vérification orthographique (optionnel).

Si vous voulez un correcteur orthographique, vous devez installer le paquet PyEnchant. La bibliothèque de vérification orthographique doit être au moins 3.0 pour fonctionner sous Windows. Sous Linux, la version 2.0 fonctionne également très bien.

Si vous installez à partir de PyPi, ces dépendances devraient être installées automatiquement. Si vous installez à partir des sources, les dépendances peuvent toujours être installées à partir de PyPi avec :

```
pip install -r requirements.txt
```

Note

Sur les distributions Linux, la bibliothèque Qt est généralement divisée en plusieurs paquets. Dans certains cas, des dépendances secondaires peuvent ne pas être installées automatiquement. Pour novelWriter, les fichiers de la bibliothèque pour le rendu des icônes SVG peuvent être omis et doivent être installés manuellement. C'est le cas, par exemple, sous Arch Linux.

22.2 Compiler et installer à partir des sources

Si vous souhaitez installer novelWriter directement à partir des sources disponibles sur [GitHub](#), vous devez d'abord construire le paquet à l'aide de l'outil de construction de la Python Packaging Authority. Il peut être installé avec :

```
pip install build
```

Sur les systèmes basés sur Debian, l'outil peut également être installé avec :

```
sudo apt install python3-build
```

Une fois l'outil installé, exécutez la commande suivante à partir de la racine du code source de novelWriter :

```
python -m build --wheel
```

Cela devrait générer un fichier `.whl` dans le dossier `dist/` de votre emplacement actuel. Le fichier wheel peut alors être installé sur votre système. Ici avec l'exemple de la version numéro 2.0.7, mais la vôtre peut être différente :

```
pip install --user dist/novelWriter-2.0.7-py3-none-any.whl
```

22.3 Création des fichiers de traduction

Si vous avez installé novelWriter à partir d'un paquetage, les fichiers de traduction devraient être préconstruits et inclus. Si vous exécutez novelWriter à partir du code source, vous devrez générer les fichiers vous-même. Les fichiers dont vous avez besoin seront écrits dans le dossier `novelwriter/assets/i18n`, et auront l'extension `.qm`.

Vous pouvez construire les fichiers `.qm` avec :

```
python pkgutils.py qtlrelease
```

Cela nécessite que l'outil Qt Linguist soit installé sur votre système. Sur Ubuntu et Debian, le paquet nécessaire s'appelle `qttools5-dev-tools`.

Note

Si vous voulez améliorer novelWriter avec des fichiers de traduction pour une autre langue, ou mettre à jour une traduction existante, les instructions pour contribuer peuvent être trouvées dans le fichier `README.md` dans le dossier `i18n` du code source.

22.4 Construction du projet d'exemple

Afin de pouvoir créer de nouveaux projets à partir des fichiers d'exemple, vous avez besoin d'un fichier `sample.zip` dans le dossier `assets` de la source. Ce fichier peut être construit à partir du script `pkgutils.py` en exécutant :

```
python pkgutils.py sample
```

22.5 Élaboration de la documentation

Une copie locale de cette documentation peut être générée au format HTML. Cela nécessite l'installation de certains paquets Python à partir de PyPi :

```
pip install -r docs/requirements.txt
```

La documentation peut ensuite être créée à partir du dossier racine du code source en lançant le programme :

```
make -C docs html
```

En cas de succès, la documentation devrait être disponible dans le dossier `docs/build/html` et vous pouvez ouvrir le fichier `index.html` dans votre navigateur.

Vous pouvez également créer un manuel PDF à partir de la documentation en utilisant le script `pkgutils.py` :

```
python pkgutils.py docs-pdf en
```

This will build the English documentation as a PDF using LaTeX. The file will then be copied into the assets folder and made available in the **Help** menu in novelWriter. Replace `en` with `all` to build for all languages. The Sphinx build system has a few extra dependencies when building the PDF. Please check the [Sphinx Docs](#) for more details.

Exécution des tests

Le code source de novelWriter est bien couvert par des tests. Le cadre de test utilisé pour le développement est `pytest` avec l'utilisation d'une extension pour Qt.

23.1 Dépendances

Les dépendances nécessaires à l'exécution des tests peuvent être installées avec :

```
pip install -r tests/requirements.txt
```

Cela installera quelques paquets supplémentaires pour la couverture et la gestion des tests. Le minimum requis est `pytest` et `pytest-qt`.

23.2 Test simple

Pour exécuter les tests, il suffit d'exécuter ce qui suit à partir de la racine du dossier source :

```
pytest
```

Puisque plusieurs des tests impliquent l'ouverture de l'interface graphique de novelWriter, vous pouvez désactiver l'interface graphique pour la durée de l'exécution du test. Si vous déplacez votre souris pendant que les tests sont en cours d'exécution, vous risquez d'interférer avec l'exécution de certains tests.

Vous pouvez désactiver le rendu de l'interface graphique en définissant le drapeau `QT_QPA_PLATFORM=offscreen` :

```
export QT_QPA_PLATFORM=offscreen pytest
```

23.3 Options avancées

L'ajout du drapeau `-v` à la commande `pytest` augmente la verbosité de l'exécution du test.

Vous pouvez également ajouter la génération de rapports de couverture. Par exemple en HTML :

```
export QT_QPA_PLATFORM=offscreen pytest -v --cov=novelwriter --cov-report=html
```

D'autres formats de rapport utiles sont `xml`, et `term` pour la sortie terminal.

Vous pouvez aussi lancer des tests par sous-paquet de `novelWriter` avec la commande `-m`. Les groupes de sous-paquetages disponibles sont `base`, `core`, et `gui`. Prenons un exemple :

```
export QT_QPA_PLATFORM=offscreen pytest -v --cov=novelwriter --cov-report=html -m core
```

Ceci n'exécutera que les tests du paquet « `core` », c'est-à-dire toutes les classes qui traitent les données d'un projet `novelWriter`. Les tests « `gui` », de même, exécuteront les tests des composants de l'interface graphique, et les tests « `base` » couvriront les éléments intermédiaires.

Vous pouvez également filtrer les tests avec le commutateur `-k`. Ce qui suit fera la même chose que `-m core` :

```
export QT_QPA_PLATFORM=offscreen pytest -v --cov=novelwriter --cov-report=html -k testCore
```

Tous les tests sont nommés de manière à ce que vous puissiez les filtrer en ajoutant d'autres éléments aux noms des tests. Ils commencent tous par le mot « `test` ». Vient ensuite le groupe : « `Core` », « `Base` », « `Dlg` », « `Tool` » ou « `Gui` ». Enfin, vient le nom de la classe ou du module, qui correspond généralement à un seul fichier de code source. Par exemple, l'exécution de ce qui suit lancera tous les tests pour l'éditeur de documents :

```
export QT_QPA_PLATFORM=offscreen pytest -v --cov=novelwriter --cov-report=html -k testGuiEditor
```

Pour lancer un seul test, il suffit d'ajouter le nom complet du test au commutateur `-k`.

D

Documents du roman, [19](#)

Dossier racine, [19](#)

E

En-têtes, [19](#)

I

Index du projet, [19](#)

M

Menu contextuel, [19](#)

Mot-clé, [19](#)

N

Notes de projet, [20](#)

R

Référence, [20](#)

T

Tag, [20](#)