



# **QGIS Documentation Guidelines**

**QGIS Project**

**avr. 02, 2022**



<b>1</b>	<b>Contribuer étape par étape</b>	<b>3</b>
1.1	Utiliser l'interface web de GitHub	4
1.1.1	Fork QGIS-Documentation	4
1.1.2	Faire des changements	5
1.1.3	Modifier des fichiers	6
1.1.4	Partagez vos modifications via les demandes d'amélioration	6
1.1.5	Supprimez votre branche fusionnée	9
1.2	Utiliser les outils de ligne de commande Git	9
1.2.1	Dépôt local	10
1.2.2	Ajoutez un autre dépôt distant	11
1.2.3	Mettez à jour votre branche de base	11
1.2.4	Contribuez dans votre branche de production	12
1.2.5	Partagez vos modifications	12
1.2.6	Nettoyez votre dépôt local et distant.	12
1.3	Pour aller plus loin	13
<b>2</b>	<b>Recommandations pour rédiger</b>	<b>15</b>
2.1	Rédiger la Documentation	16
2.1.1	Titres de chapitre	16
2.1.2	Listes	16
2.1.3	Balises en ligne	16
2.1.4	Étiquettes / références	17
2.1.5	Figures et images	18
2.1.6	Index	21
2.1.7	Commentaires Spéciaux	21
2.1.8	Extraits de code	22
2.1.9	Notes de pied de page	22
2.2	Les Captures d'écrans	22
2.2.1	Ajouter de nouvelles captures d'écran	22
2.2.2	Captures d'écran traduites	23
2.3	Documenter les algorithmes de traitement	23
<b>3</b>	<b>Écriture de code dans le livre de recettes « PyQGIS Cookbook »</b>	<b>27</b>
3.1	Comment écrire des extraits de code testables	27
3.1.1	Directives sphinx de Doctest	27
3.1.2	Tests groupés	29
3.2	Comment tester des extraits sur votre ordinateur local	29
<b>4</b>	<b>Recommandations pour la traduction</b>	<b>31</b>
4.1	Processus de traduction	31
4.2	Traduire un fichier	32

4.2.1	Traduire avec Transifex . . . . .	33
4.2.2	Traduire avec Qt Linguist . . . . .	34
4.2.3	Traduire un manuel . . . . .	36
4.2.4	Résumé des règles de traduction . . . . .	37
<b>5</b>	<b>Substitutions</b> . . . . .	<b>39</b>
5.1	Usage . . . . .	40
5.2	Substitutions usuelles . . . . .	40
5.2.1	Icônes de plateforme . . . . .	40
5.2.2	Éléments de menu . . . . .	41
5.3	Icônes de la barre d'outils . . . . .	42
5.3.1	Gestion de couche et aperçu . . . . .	42
5.3.2	Fichier . . . . .	43
5.3.3	Éditer . . . . .	43
5.3.4	Identity result . . . . .	43
5.3.5	Numérisation et numérisation avancée . . . . .	44
5.3.6	Navigateur de cartes et Attributs . . . . .	45
5.3.7	Selection and Expressions . . . . .	45
5.3.8	Étiquettes et Diagrammes . . . . .	46
5.3.9	Décorations . . . . .	46
5.3.10	Help . . . . .	46
5.3.11	Couleurs . . . . .	47
5.4	Autres icônes de base . . . . .	47
5.5	Table attributaire . . . . .	48
5.6	Projections et Géoréférenceur . . . . .	48
5.7	Mise en page de cartes . . . . .	48
5.8	Propriétés des couches . . . . .	50
5.9	Extensions . . . . .	51
5.9.1	Traitements . . . . .	51
5.9.2	Diverses extensions principales . . . . .	51
5.9.3	Intégration GRASS . . . . .	52

La documentation QGIS sera construite automatiquement sur le serveur à 0 h, 8 h et 16 h US / Pacifique (heure du Pacifique). L'état actuel est disponible sur <https://docs.qgis.org>.

Les fichiers source de la documentation de QGIS sont disponibles sur <https://github.com/qgis/QGIS-Documentation>. Ils sont principalement écrits en utilisant la syntaxe du format reStructuredText (reST), associée avec quelques scripts de la boîte à outils Sphinx pour le post traitement de la sortie HTML. Pour plus d'informations sur ces outils, voir <http://docutils.sourceforge.net/docs/ref/rst/restructuredtext.html> ou <https://www.sphinx-doc.org/en/master/usage/restructuredtext/basics.html>.

Les chapitres suivants vous guideront tout au long de l'apprentissage:

- comment gérer les fichiers source de documentation en utilisant le système [git](#) et la plateforme [GitHub](#) sur laquelle ils sont stockés
- comment modifier les textes, fournir des captures d'écran ... de manière conforme
- comment partager et vous assurer que vos modifications sont transmises aux documents officiels.

Si vous recherchez des informations générales sur la façon de contribuer au projet QGIS, vous pouvez trouver de l'aide sur [Participez à la communauté QGIS](#).



---

## Contribuer étape par étape

---

- *Utiliser l'interface web de GitHub*
  - *Fork QGIS-Documentation*
  - *Faire des changements*
    - \* *Alternative 1: Utiliser le lien Edit on GitHub*
    - \* *Alternative 2 : Créer une branche ad hoc dans votre dépôt de documentation*
  - *Modifier des fichiers*
  - *Partagez vos modifications via les demandes d'amélioration*
    - \* *Démarrer une nouvelle demande d'amélioration*
    - \* *Comparer les changements*
    - \* *Décrivez votre demande amélioration*
    - \* *Examiner et commenter la demande d'amélioration*
    - \* *Faire des corrections*
  - *Supprimez votre branche fusionnée*
- *Utiliser les outils de ligne de commande Git*
  - *Dépôt local*
  - *Ajoutez un autre dépôt distant*
  - *Mettez à jour votre branche de base*
  - *Contribuez dans votre branche de production*
  - *Partagez vos modifications*
  - *Nettoyez votre dépôt local et distant.*
- *Pour aller plus loin*

---

**Note:** Bien que la documentation de QGIS soit utilisée pour expliquer le processus, toutes les commandes et étapes

---

montrées ci-dessous sont également applicables au site web de QGIS (dans le dépôt QGIS-Website).

---

Si vous lisez ces lignes, c'est sûrement parce que vous souhaitez contribuer à écrire la documentation de QGIS et cherchez un guide pratique. Vous êtes au bon endroit! Ce document vous guidera à travers les différentes façons d'atteindre cet objectif, vous indiquant les étapes principales, les astuces que vous pouvez utiliser et les pièges dont vous devez vous méfier.

Pour toute aide, n'hésitez pas à demander dans un commentaire sur le rapport de problème que vous essayez de corriger ou à écrire dans la [liste QGIS-community-team](#). Plus de détails dans [Participer à la documentation](#).

Plongeons-nous maintenant dans le processus.

Les sources de la documentation sont stockées à l'aide du système de contrôle de version git et sont disponibles sur GitHub à l'adresse <https://github.com/qgis/QGIS-Documentation>. Une liste de problèmes à résoudre et de fonctionnalités à expliquer est disponible à l'adresse <https://github.com/qgis/QGIS-Documentation/issues>.

---

**Astuce:** Si vous êtes un nouveau contributeur et que vous ne savez pas par où commencer, vous pouvez vous attaquer à nos rapports de bienvenue <<https://github.com/qgis/QGIS-Documentation/issues?q=is%3Aissue+is%3Aopen+label%3AEasy>>\_.

---

Il y a deux façons principales, non mutuellement exclusives, de modifier les fichiers :

1. *à l'aide de l'interface web de Github*
2. *avec des outils de ligne de commande Git.*

## 1.1 Utiliser l'interface web de GitHub

L'interface web de GitHub vous permet de faire comme suit:

- éditer des fichiers
- prévisualiser et valider vos changements
- Créer une pull request pour ajouter vos modifications dans le répertoire principal.
- Créer, mettre à jour ou supprimer des branches

Si vous n'êtes pas encore familier avec le vocabulaire git et GitHub, vous pouvez lire le projet [Hello-world](#) de Github pour apprendre un vocabulaire de base et les actions qui seront utilisées ci-dessous.

---

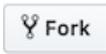
**Note:** Si vous corrigez un problème signalé

Si vous comptez apporter des corrections à un [signalement](#), ajoutez un commentaire au rapport d'erreur afin de vous le faire assigner. Ceci empêchera que plus d'une personne travaille sur la même erreur.

---

### 1.1.1 Fork QGIS-Documentation

Dans l'hypothèse où vous avez déjà un [compte GitHub](#), il vous faudra d'abord cloner les fichiers source de la documentation.

Naviguez vers la page [QGIS-Documentation repository](#) et cliquez sur le bouton  en haut à droite.

Dans votre compte GitHub, vous trouverez un dépôt pour la documentation QGIS (<https://github.com/<YourName>/QGIS-Documentation>). Ce dépôt est une copie du dépôt QGIS-Documentation officiel où vous avez un accès complet en écriture et vous pouvez apporter des modifications sans affecter la documentation officielle.

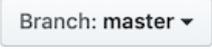
---

## 1.1.2 Faire des changements

Il y a différentes façons de contribuer à la documentation QGIS. Bien que nous les exposions ci-dessous séparément, vous pouvez passer d'un processus à l'autre sans risque.

### Alternative 1: Utiliser le lien `Edit on GitHub`

Chaque page de la documentation QGIS peut être modifiée rapidement et facilement en cliquant sur le lien `Edit on GitHub` en haut à droite de celle-ci.

1. Cela ouvrira le fichier dans la branche `qgis:master` avec un message en haut de la page vous indiquant que vous n'avez pas accès en écriture à ce dépôt et vos modifications seront appliquées à une nouvelle branche de votre dépôt.
2. Faites vos changements. Étant donné que la documentation est écrite à l'aide de la syntaxe `reStructureText`, en fonction de vos modifications, vous devrez peut-être vous baser sur les *Recommandations pour la documentation*.
3. Lorsque vous avez terminé, faites un bref commentaire sur vos modifications et cliquez sur *Propose changes*. Cela va générer une nouvelle *branche* (`patch-xxx`) dans votre dépôt.
4. Après avoir cliqué sur *Propose changes github* accède à la page *Comparing changes* (Comparaison des modifications).
  - Si vous avez terminé d'apporter des modifications, passez à *Partagez vos modifications* dans la section *Partager les changements via une Pull Request* ci-dessous.
  - S'il y a des changements supplémentaires à effectuer avant de les soumettre à QGIS, suivez ces étapes :
    1. Accédez à votre fork de documentation QGIS (<https://github.com/<YourName>/QGIS-Documentation>)
    2. Cliquez sur  et recherchez la branche `patch-xxx`. Sélectionnez cette branche de patch. Le bouton  va maintenant dire *Branch: patch-xxx*
    3. Passez à la section *Modifier des fichiers* ci-dessous.

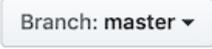
---

**Note:** Le lien `Edit on GitHub` est aussi accessible depuis le menu déroulant au bas de la barre verticale à gauche de chaque page.

---

### Alternative 2 : Créer une branche ad hoc dans votre dépôt de documentation

Vous pouvez modifier des fichiers directement à partir de votre fork de la documentation QGIS.

Cliquez sur  dans le coin supérieur gauche de votre dépôt QGIS-Documentation forké et entrez un nom unique dans le champ de texte pour créer une nouvelle *branche*. Le nom de la nouvelle branche doit être lié au problème que vous souhaitez résoudre. Le bouton  devrait maintenant indiquer *Branch: nom\_branche*

---

**Astuce:** **Faites vos changements dans une branche ad hoc, jamais dans la branche `**`master``**

Par convention, évitez de faire des changements dans votre *branche master* sauf lorsque vous fusionnez les modifications de la *branche master* de `qgis/QGIS-Documentation` dans votre copie du dépôt QGIS-Documentation. Des branches distinctes vous permettent de travailler sur plusieurs problèmes en même temps sans interférer avec d'autres branches. Si vous faites une erreur, vous pouvez toujours supprimer une branche et recommencer en créant une nouvelle à partir de la branche principale.

---

### 1.1.3 Modifier des fichiers

1. Parcourez les fichiers source de votre fork de QGIS-Documentation jusqu'au fichier à modifier
2. Procédez à vos modifications en suivant les *règles d'écriture*.
3. Lorsque vous avez terminé, accédez au cadre **Valider les modifications** au bas de la page, faites un bref commentaire sur vos modifications et cliquez sur : `guilabel:Valider les modifications` pour valider les modifications directement dans votre branche. Assurez-vous que *Valider directement dans la branche nom\_branche* est sélectionné.
4. Répétez les étapes précédentes pour tout autre fichier qui doit être mis à jour pour résoudre le problème

### 1.1.4 Partagez vos modifications via les demandes d'amélioration

Vous devez effectuer une pull request pour ajouter vos modifications dans la documentation officiel.

---

**Note:** Si vous avez utilisé un lien `Edit on GitHub`

Après avoir validé vos modifications, GitHub ouvrira automatiquement une nouvelle page comparant les modifications que vous avez apportées dans votre branche `patch-xxx` à la branche principale `qgis/QGIS-Documentation`.

Passez à l'*Étape 2* ci-dessous.

---

### Démarrer une nouvelle demande d'amélioration

Accédez à la page principale de *QGIS-Documentation* <<https://github.com/qgis/QGIS-Documentation>> \_ repository et cliquez sur *nouvelle demande amélioration*.

### Comparer les changements

Si vous voyez deux boîtes de dialogue, l'une qui dit `base:master` et l'autre `compare:nom_branche` (voir figure), cela ne fera que fusionner vos modifications d'une de vos branches vers votre branche principale. Pour résoudre ce problème, cliquez sur le lien *Comparer à travers les branches*.

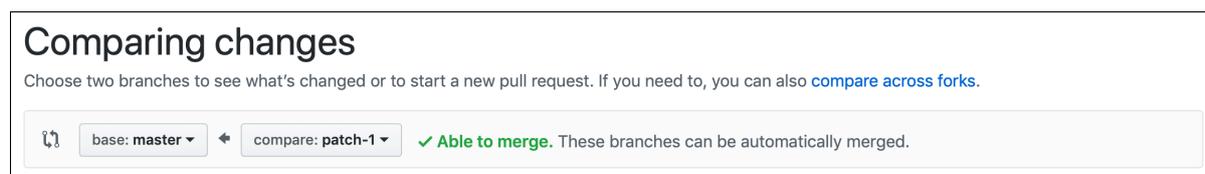


Figure 1.1: Si votre page *Comparaison des modifications* ressemble à ceci, cliquez sur le lien *comparaison entre les branches*.

Vous devriez voir quatre menus déroulants. Ceux-ci vous permettront de comparer les modifications que vous avez apportées dans votre branche avec la branche principale dans laquelle vous souhaitez fusionner. Elles sont:

- **branche base:** la branche dans laquelle vous souhaitez fusionner vos modifications
- **base:** la branche de la fourche de base dans laquelle vous souhaitez fusionner vos modifications
- **branche de tête:** la fourche qui a des changements que vous souhaitez intégrer dans la branche de base
- **comparer:** la branche avec ces changements

Sélectionnez `qgis/QGIS-Documentation` comme **branche de base** avec `master` comme **base**, définissez la fourche de tête sur votre dépôt `<YourName>/QGIS-Documentation`, et comparez à votre branche modifiée.

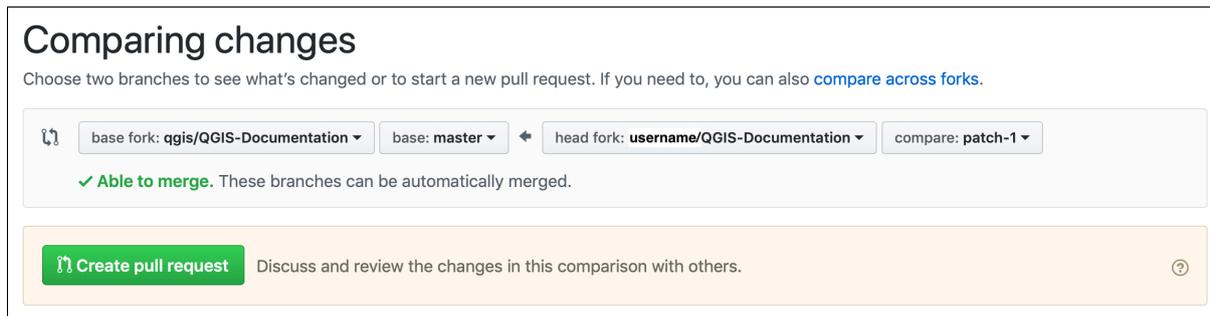


Figure 1.2: Comparaison des modifications entre qgis/QGIS-Documentation et votre dépôt

Une coche verte avec les mots **Capable de fusionner** montre que vos modifications peuvent être fusionnées dans la documentation officielle sans conflits.

Cliquez sur le bouton *Créer une demande amélioration*.

**Avertissement:** Si vous voyez **✗ Can't automatically merge**.

Cela signifie qu'il existe des *conflits* <<https://help.github.com/articles/addressing-merge-conflicts/>> . Les fichiers que vous modifiez ne sont pas à jour avec la branche que vous ciblez, car quelqu'un d'autre a effectué une validation qui entre en conflit avec vos modifications. Vous pouvez toujours créer la demande d'extraction, mais vous devrez corriger tout *conflit* pour terminer la fusion.

**Astuce:** Bien qu'en cours de traduction, la documentation de la **dernière version** de QGIS est toujours maintenu et les problèmes existants sont résolus. Si vous résolvez des problèmes pour une version différente, remplacez **base** de **master** par la branche `release_...` appropriée dans les étapes ci-dessus.

## Décrivez votre demande amélioration

Une zone de texte s'ouvrira: remplissez les commentaires pertinents pour le problème que vous abordez.

Si cela concerne un **problème particulier**, ajoutez le numéro du problème à vos commentaires. Pour ce faire, saisissez # et le numéro du problème (*par exemple* #1234). S'il est précédé de termes tels que `fix` ou `close`, le signalement concerné sera clos dès que la pull request sera fusionnée.

Ajoutez des liens vers toutes les pages de documentation que vous modifiez.

Cliquez sur *Créer une demande d'amélioration*.

## Examiner et commenter la demande d'amélioration

Comme vu ci-dessus, n'importe qui peut soumettre des modifications à la documentation par le biais de pull requests. De même, n'importe qui peut examiner les demandes d'amélioration avec des questions et des **commentaires**. Peut-être que le style d'écriture ne correspond pas aux directives du projet, qu'il manque certains détails majeurs ou captures d'écran au changement, ou peut-être que tout a l'air bien et est en ordre. La révision permet d'améliorer la qualité de la contribution, tant sur le fond que sur la forme.

Pour examiner une demande amelioration :

1. Accédez à la [page des demandes d'amélioration](#) et cliquez sur la demande d'amélioration que vous souhaitez commenter.
2. Au bas de la page, vous trouverez une zone de texte où vous pouvez laisser des commentaires généraux sur la demande amelioration.

3. Pour ajouter des commentaires sur des lignes spécifiques,

1. Cliquez sur  Files changed et recherchez le fichier que vous souhaitez commenter. Vous devrez peut-être cliquer sur *Afficher le diff source* pour voir les changements.
2. Faites défiler jusqu'à la ligne que vous souhaitez commenter et cliquez sur . Cela ouvrira une zone de texte vous permettant de laisser un commentaire.

Des commentaires de ligne spécifiques peuvent être publiés:

- sous forme de commentaires simples, en utilisant le bouton *Ajouter un seul commentaire*. Ils sont publiés au fur et à mesure. N'utilisez ceci que si vous avez peu de commentaires à ajouter ou lorsque vous répondez à un autre commentaire.
- ou dans le cadre d'une révision, en appuyant sur le bouton *Start a review*. Vos commentaires ne sont pas envoyés automatiquement après validation, ce qui vous permet de les modifier ou de les annuler par la suite, d'ajouter un résumé des principaux points de la révision ou des instructions globales concernant la demande d'amélioration et si vous l'approuvez ou non. Ce moyen est le plus pratique car il est plus flexible et vous permet de structurer votre avis, de modifier les commentaires, de publier lorsque vous êtes prêt et d'envoyer une seule notification aux abonnés du dépôt et pas une notification pour chaque commentaire. Obtenez [plus de détails](#).



Figure 1.3: Commenter une ligne avec une suggestion de changement

Les commentaires de ligne peuvent incorporer des suggestions que le rédacteur de demande d'amélioration peut appliquer à la demande d'amélioration. Pour ajouter une suggestion, cliquez sur le bouton  Insérer une suggestion en haut de la zone de texte du commentaire et modifiez le texte dans le bloc de suggestion.

### Astuce: Préférez la validation des suggestions à votre demande amélioration par lots

En tant qu'auteur de demande d'amélioration, lorsque vous intégrez directement les commentaires des réviseurs dans votre demande d'amélioration, évitez d'utiliser le bouton *Commit suggestion* en bas du commentaire lorsque vous avez de nombreuses suggestions à traiter et préférez les ajouter en tant que commit batch, C'est:

1. Basculez vers onglet  Files changed
2. Appuyez sur *Ajouter une suggestion au lot* pour chaque reformulation que vous souhaitez inclure. Vous verrez un compteur augmenter au fur et à mesure.
3. Appuyez sur l'un des boutons *Valider les suggestions* lorsque vous êtes prêt à appliquer les suggestions à votre demande d'amélioration, puis entrez un message décrivant les modifications.

Cela ajoutera toutes les modifications à votre branche en tant que validation unique, résultant en un historique des modifications plus lisible et moins de notifications pour les abonnés du référentiel. Soit dit en passant, procéder

comme cela vous permettra également d'économiser de nombreux clics.

## Faire des corrections

Une nouvelle demande d'amélioration sera automatiquement ajoutée à la [Liste des suggestions d'amélioration](#). D'autres éditeurs et administrateurs examineront votre proposition d'amélioration et pourront faire des suggestions ou demander des corrections.

Une demande d'amélioration déclenchera également des vérifications automatiques de la construction (par exemple, pour le formatage des rst, les syntaxes du code python), et les rapports sont affichés en bas de la page. Si une erreur est trouvée, une croix rouge apparaîtra à côté de votre commit. Cliquez sur la croix rouge ou sur [Détails](#) dans la section résumé en bas de la page de la demande amélioration pour voir les détails de l'erreur. Vous devrez corriger toutes les erreurs ou avertissements signalés avant que vos modifications ne soient validées dans le référentiel `qgis/QGIS-Documentation`.

Vous pouvez apporter des modifications à votre demande d'extraction jusqu'à ce qu'elle soit fusionnée avec le dépôt principal, soit pour améliorer votre demande, pour répondre aux modifications demandées, soit pour corriger une erreur de génération.

Pour apporter des modifications, cliquez sur l'onglet  Files changed dans votre page de demande d'amélioration et cliquez sur le bouton crayon  à côté du nom de fichier que vous souhaitez modifier.

Toute modification supplémentaire sera automatiquement ajoutée à votre demande d'amélioration si vous apportez ces modifications à la même branche que celle que vous avez soumise dans votre demande d'amélioration. Pour cette raison, vous ne devez apporter des modifications supplémentaires que si ces modifications sont liées au problème que vous avez l'intention de résoudre avec cette demande d'amélioration

Si vous souhaitez résoudre un autre problème, créez une nouvelle branche pour ces modifications et répétez les étapes ci-dessus.

Un administrateur fusionnera votre contribution une fois que toutes les erreurs de génération auront été corrigées et que vous et les administrateurs serez satisfaits de vos modifications.

### 1.1.5 Supprimez votre branche fusionnée

Vous pouvez supprimer la branche une fois vos modifications fusionnées. La suppression d'anciennes branches vous évite d'avoir des branches inutilisées et obsolètes dans votre dépôt.

1. Accédez à votre fork du dépôt de la documentation de QGIS (<https://github.com/<YourName>/QGIS-Documentation>).
2. Cliquez sur l'onglet *Branches*. Dans *Your branches*, vous verrez une liste de vos branches.
3. Cliquez sur le bouton  Delete this branch pour supprimer les branches non désirées.

## 1.2 Utiliser les outils de ligne de commande Git

L'interface Web GitHub est un moyen facile de mettre à jour le dépôt de documentation QGIS avec vos contributions, mais elle ne propose pas d'outils pour :

- regrouper vos commits et nettoyer votre historique des changements
- résoudre les conflits possibles avec le dépôt principal
- construire la documentation pour tester vos modifications

Vous devez [installer git](#) sur votre disque dur afin d'avoir accès à des outils plus avancés et plus puissants et avoir une copie locale du dépôt. Certaines notions de base dont vous aurez souvent besoin sont exposées ci-dessous. Vous trouverez également des règles à respecter même si vous optez pour l'interface Web.

Dans les exemples de code ci dessous, les lignes commençant par \$ représente les commandes que vous aurez à taper, alors que les # sont des commentaires.

### 1.2.1 Dépôt local

Vous êtes maintenant prêt à obtenir un clone local de **votre** dépôt de documentation QGIS.

Vous pouvez cloner votre dépôt QGIS à l'aide de l'URL Web comme suit :

```
# move to the folder in which you intend to store the local repository
$ cd ~/Documents/Development/QGIS/
$ git clone https://github.com/<YourName>/QGIS-Documentation.git
```

La première ligne de commande est simplement un exemple. Vous devez adapter à la fois le chemin et l'URL du dépôt, en remplaçant <YourName> par votre nom d'utilisateur github.

Vérifiez les points suivants :

```
# Enter the local repository
$ cd ./QGIS-Documentation
$ git remote -v
origin https://github.com/<YourName>/QGIS-Documentation.git (fetch)
origin https://github.com/<YourName>/QGIS-Documentation.git (push)
$ git branch
* master
```

- *origin* est le nom du dépôt distant de votre dépôt QGIS-Documentation.
- *master* est la branche principale par défaut. Vous ne devriez jamais l'utiliser pour vos contributions ! **Jamais!**

Vous pouvez également cloner votre dépôt QGIS à l'aide du protocole SSH :

```
# move to the folder in which you intend to store the local repository
$ cd ~/Documents/Development/QGIS/
$ git clone git@github.com:<YourName>/QGIS-Documentation.git
```

---

#### Astuce: Permission denied (publickey) error?

Si vous obtenez une erreur Autorisation refusée (publickey) avec l'ancienne commande, il peut y avoir un problème avec votre clé SSH. Voir [Aide GitHub](#) pour plus de détails.

---

Vérifiez les points suivants si vous avez utilisé le protocole SSH:

```
# Enter the local repository
$ cd ./QGIS-Documentation
$ git remote -v
origin git@github.com:<YourName>/QGIS-Documentation.git (fetch)
origin git@github.com:<YourName>/QGIS-Documentation.git (push)
$ git branch
* master
```

Vous pouvez commencer à travailler ici, mais à long terme, vous obtiendrez beaucoup de problèmes lorsque vous pousserez votre contribution (appelée Pull Request dans le processus github) car la branche principale du dépôt qgis/QGIS-Documentation divergera de votre dépôt local / distant. Vous devez ensuite suivre le dépôt distant principal et travailler avec les branches.

## 1.2.2 Ajoutez un autre dépôt distant

Pour pouvoir suivre l'avancement du travail réalisé sur le projet principal, ajoutez un nouveau dépôt distant dans votre dépôt local. Ce nouveau dépôt distant sera le dépôt QGIS-Documentation du projet QGIS :

```
$ git remote add upstream https://github.com/qgis/QGIS-Documentation.git
$ git remote -v
origin https://github.com/<YourName>/QGIS-Documentation.git (fetch)
origin https://github.com/<YourName>/QGIS-Documentation.git (push)
upstream https://github.com/qgis/QGIS-Documentation.git (fetch)
upstream https://github.com/qgis/QGIS-Documentation.git (push)
```

De même, vous pouvez utiliser le protocole SSH pour ajouter un dépôt distant dans votre dépôt local :

```
$ git remote add upstream git@github.com:qgis/QGIS-Documentation.git
$ git remote -v
origin git@github.com:<YourName>/QGIS-Documentation.git (fetch)
origin git@github.com:<YourName>/QGIS-Documentation.git (push)
upstream git@github.com:qgis/QGIS-Documentation.git (fetch)
upstream git@github.com:qgis/QGIS-Documentation.git (push)
```

Désormais, vous avez le choix entre deux dépôts distants :

- *origin* pour « pousser » votre branche locale dans  **votre** dépôt distant
- *upstream* pour fusionner (si vous avez les droits pour le faire) votre contribution avec le dépôt officiel OU pour mettre à jour votre branche « master » sur le dépôt local à partir de la branche « master » du dépôt officiel.

---

**Note:** *upstream* est juste un intitulé, une sorte de nom standard, mais vous pouvez l'appeler comme vous voulez.

---

## 1.2.3 Mettez à jour votre branche de base

Avant de travailler sur une nouvelle contribution, vous devez toujours mettre à jour votre branche principale dans votre dépôt local. En supposant que vous souhaitiez apporter des modifications à la documentation de test, exécutez les lignes de commande suivantes :

```
# switch to master branch (it is easy to forget this step!)
$ git checkout master
# get "information" from the master branch in the upstream repository
# (aka qgis/QGIS-Documentation's repository)
$ git fetch upstream master
# merge update from upstream/master to the current local branch
# (which should be master, see step 1)
$ git merge upstream/master
# update your remote repository (aka <YourName>/QGIS-Documentation)
$ git push origin master
```

Vous avez maintenant vos dépôt locaux et distants qui ont tous deux leur branche `master` à jour avec la branche `master` officielle de la documentation QGIS. Vous pouvez commencer à travailler sur votre contribution.

---

**Note:** Changez de branche si vous souhaitez contribuer au document publié

En parallèle de la documentation de test, nous continuons à résoudre les problèmes de la [dernière version](#), ce qui signifie que vous pouvez également y contribuer. Suivez l'exemple de code de la section précédente, en remplaçant `master` par la branche correspondante de la dernière documentation.

---

### 1.2.4 Contribuez dans votre branche de production

Maintenant que la branche de base est mise à jour, il vous faut créer une branche spéciale pour accueillir vos modifications. Ayez le réflexe de toujours travailler sur une branche autre que celle de base - souvent la *master*! Toujours!

```
# Create a new branch
$ git checkout -b myNewBranch
# checkout means go to the branch
# and -b flag creates a new branch if needed, based on current branch
# Let's check the list of existing branches (* indicates the current branch)
$ git branch
master
release_2.18
...
* myNewBranch
# You can now add your contribution, by editing the concerned file(s)
# with any application (in this case, vim is used)
$ vim myFile
# once done
$ git add myFile
$ git commit
```

Quelques remarques à propos des commandes de commit/push :

- essayez de ne « commiter » qu'une seule contribution (changement atomique). En d'autres termes, n'adressez qu'une seule erreur à la fois.
- essayez d'expliquer avec soin ce que vous avez modifié dans le titre de votre commit et dans la description. La première ligne est un titre, doit commencer par une lettre majuscule, devra contenir 80 caractères au maximum et ne devra pas se terminer par un .. Soyez concis. Votre description peut être plus longue et se termine par un .. Vous pouvez y donner plus de détails.
- utilisez un # avec un nombre pour faire référence à un problème. Préfixez avec `Fix` si vous fixez le ticket: votre commit fermera le ticket.

Maintenant que vos modifications sont sauvegardées et intégrées dans votre branche locale, il va falloir les envoyer sur le dépôt en ligne, afin de pouvoir créer des pull-requests:

```
$ git push origin myNewBranch
```

### 1.2.5 Partagez vos modifications

Vous pouvez maintenant aller dans votre dépôt github et *créer une Pull Request* comme expliqué dans une section précédente. Assurez-vous de créer une PR depuis votre branche vers la branche distante cible dans le dépôt officiel de QGIS-Documentation.

### 1.2.6 Nettoyez votre dépôt local et distant.

Une fois que votre PR a été fusionnée dans le dépôt officiel QGIS-Documentation, vous pouvez supprimer votre branche. En effet, si vous contribuez souvent, vous vous retrouverez d'ici quelques semaines avec un nombre considérable de branches inutilisées. Du coup, gardez votre dépôt propre de cette façon :

```
# delete local branch
$ git branch -d myNewBranch
# Remove your remote myNewBranch by pushing nothing to it
$ git push origin :myNewBranch
```

Et n'oubliez pas de mettre à jour la branche `master` dans votre dépôt local!

## 1.3 Pour aller plus loin

- Outre l'interface Web Github et les outils de ligne de commande git exposés ci-dessus, il existe également des applications *GUI* que vous pouvez utiliser pour créer et gérer vos contributions à la documentation.
- Lorsque les modifications de la demande d'extraction sont en conflit avec des modifications récentes transmises à la branche cible, les conflits doivent être résolus avant qu'une fusion ne soit possible:
  - si le conflit concerne quelques lignes concurrentes, un bouton *Résoudre les conflits* est disponible dans la page de demande d'extraction Github. Appuyez sur le bouton et résolvez le problème comme expliqué sur <https://help.github.com/articles/resolving-a-merge-conflict-on-github/>
  - si le conflit implique le renommage ou la suppression de fichiers, vous devrez résoudre le conflit à l'aide des lignes de commande git. En règle générale, vous devez d'abord rebaser votre branche sur la branche cible à l'aide de l'appel `git rebase targetBranch` et corriger les conflits signalés. En savoir plus sur <https://help.github.com/articles/resolving-a-merge-conflict-using-the-command-line/>
- Parfois, à la fin du processus de relecture, vous pouvez vous retrouver avec des modifications divisées en plusieurs validations qui n'en valent pas nécessairement la peine. Les lignes de commande Git vous aident à écraser ces validations en un nombre plus petit et en des messages de validation plus significatifs. Quelques détails sur <https://help.github.com/articles/using-git-rebase-on-the-command-line/>



---

### Recommandations pour rédiger

---

- *Rédiger la Documentation*
  - *Titres de chapitre*
  - *Listes*
  - *Balises en ligne*
  - *Étiquettes / références*
  - *Figures et images*
    - \* *Illustrations*
    - \* *Remplacement*
    - \* *Figure*
    - \* *Tables*
  - *Index*
  - *Commentaires Spéciaux*
  - *Extraits de code*
  - *Notes de pied de page*
- *Les Captures d'écrans*
  - *Ajouter de nouvelles captures d'écran*
  - *Captures d'écran traduites*
- *Documenter les algorithmes de traitement*

En général, lors de la création de la documentation reST pour le projet QGIS, veuillez suivre les [Consignes de style de documentation Python](#). Pour plus de commodité, nous fournissons ci-dessous un ensemble de règles générales sur lesquelles nous nous appuyons pour rédiger la documentation QGIS.

## 2.1 Rédiger la Documentation

### 2.1.1 Titres de chapitre

À chaque page web de la documentation correspond un fichier `.rst`.

Les différentes parties qui structurent le texte sont identifiées par leur titre qui est souligné (et surligné pour le premier niveau). Les titres de même niveau doivent utiliser le même caractère de soulignement. Dans la documentation QGIS, voici les styles à utiliser pour le chapitre, la section, la sous-section et la mini section.

```

*****
Chapter
*****

Section
=====

Subsection
-----

Minisec
.....

Subminisec
^^^^^^^^^^
    
```

### 2.1.2 Listes

Les listes sont utiles pour structurer le texte. Voici quelques règles simples communes à toutes les listes:

- Démarrer tous les éléments de la liste avec une majuscule
- N'utilisez pas de ponctuation après les éléments de liste qui ne contiennent qu'une seule phrase simple
- Utilisez le point (.) Comme signe de ponctuation pour les éléments de liste qui se composent de plusieurs phrases ou d'une seule phrase composée

### 2.1.3 Balises en ligne

Vous pouvez utiliser des balises pour souligner les éléments.

- **Interface de menu** : pour indiquer une séquence de sélections dans un menu, y compris la sélection de sous-menus et le choix d'une opération particulière, ou n'importe quelle sous-séquence d'une telle séquence.

```
:menuselection:`menu --> submenu`
```

- **Boîtes de dialogue et titres d'onglet**: étiquettes présentées dans le cadre d'une interface utilisateur interactive, y compris les titres de fenêtre, les titres d'onglet, les étiquettes de bouton et d'option.

```
:guilabel:`title`
```

- **Noms de fichiers et répertoires**

```
:file:`README.rst`
```

- **Icônes avec texte contextuel**

```
|icon| :sup:`popup_text`
```

(voir *image* ci-dessous).

• **Raccourcis clavier**

```
:kbd:`Ctrl+B`
```

affichera `Ctrl+B`

Lors de la description des raccourcis clavier, les conventions suivantes doivent être utilisées:

- Les lettres sont affichées en majuscules: `S`
- Les touches spéciales sont affichées avec une première lettre majuscule: `Esc`
- Les combinaisons de touches sont affichées avec un signe + entre les touches, sans espaces: `Shift+R`

• **Texte utilisateur**

```
`label`
```

### 2.1.4 Étiquettes / références

Les ancres à l'intérieur du texte peuvent être utilisées pour créer des hyperliens vers des sections ou des pages.

L'exemple ci-dessous crée la référence d'une section (e.g., Libellé/titre de la référence)

```
.. _my_anchor:  
  
Label/reference  
-----
```

Pour appeler la référence dans la **même page**, utilisez

```
see my_anchor_ for more information.
```

qui renverra :

voir *my\_anchor* pour plus d'informations.

Notez qu'il sautera à la ligne / chose qui suit «l'ancre». Vous n'avez pas besoin d'utiliser des apostrophes, mais vous devez avoir des lignes vides après l'ancre.

Une autre façon de créer un lien accessible **depuis n'importe où dans la documentation** est d'utiliser `:ref:`.

```
see :ref:`my_anchor` for more information.
```

ce qui créera un lien avec la légende à la place (dans ce cas le titre de cette section!):

voir *Étiquettes / références* pour plus d'informations.

Donc, référence 1 (*my\_anchor*) et référence 2 (*Étiquettes / références*). Étant donné que la référence affiche souvent une légende complète, il n'est pas vraiment nécessaire d'utiliser le mot *section*. Notez que vous pouvez également utiliser une légende personnalisée pour décrire la référence:

```
see :ref:`Label and reference <my_anchor>` for more information.
```

qui renvoie:

voir *Étiquette et référence* pour davantage de détails.

### 2.1.5 Figures et images

#### Illustrations

Pour insérer une image, utilisez

```
.. figure:: /static/common/logo.png
   :width: 10 em
```

qui renvoie



#### Remplacement

Vous pouvez mettre une image dans du texte ou ajouter un alias à utiliser partout. Pour utiliser une image à l'intérieur d'un paragraphe, créez d'abord un alias dans le fichier `source/substitutions.txt`:

```
.. |nice_logo| image:: /static/common/logo.png
   :width: 1 em
```

puis appelez-le dans votre paragraphe:

```
My paragraph begins here with a nice logo |nice_logo|.
```

Voici comment l'exemple sera affiché:

Mon paragraphe commence ici par un joli logo .

Pour permettre un aperçu de rendu dans GitHub aussi proche que possible du rendu HTML, vous devrez également ajouter l'appel de remplacement d'image à la fin du fichier que vous avez modifié. Cela peut être fait en le copiant-collant depuis `substitutions.txt` ou en exécutant le script `scripts/find_set_subst.py`.

---

**Note:** Actuellement, pour assurer la cohérence et aider à l'utilisation des icônes QGIS, une liste d'alias est construite et disponible dans le chapitre [Substitutions](#).

---

#### Figure

```
.. _figure_logo:
.. figure:: /static/common/logo.png
   :width: 20 em
   :align: center

   A caption: A logo I like
```

Le résultat ressemble à ceci :



Figure2.1: Un titre : Un logo que j'aime

Pour éviter les conflits avec d'autres références, commencez toujours les ancres de figures par `_figure_` et utilisez des termes qui se connectent facilement à la légende de la figure. Bien que seul l'alignement centré soit obligatoire pour l'image, n'hésitez pas à utiliser d'autres options pour les figures (telles que `width`, `height`, `scale` ...) si nécessaire.

Les scripts inséreront un numéro généré automatiquement avant la légende du chiffre dans les versions HTML et PDF de la documentation générée.

Pour utiliser un titre (voir Mon titre) insérez juste un texte indenté après une ligne blanche dans le bloc de la figure.

Une figure peut être référencée à l'aide de l'étiquette de référence comme ceci:

```
see :numref:`figure_logo`
```

rend comme ceci :

voir [Fig. 2.1](#)

C'est la façon préférée de référencer les nombres.

---

**Note:** Pour que `:numref:` fonctionne, le chiffre **doit avoir une légende**.

---

Évitez d'utiliser `:ref:` au lieu de `:numref:` pour la référence, car cela renvoie la légende complète de l'image.

```
see :ref:`figure_logo`
```

rend comme ceci :

voir *Un titre : Un logo que j'aime*

## Tables

Un simple tableau peut être codé comme suit

```

=====  =====  =====
x          y          z
=====  =====  =====
1          2          3
4          5
=====  =====  =====
    
```

Il sera rendu ainsi :

x	y	z
1	2	3
4		5

Utilisez une barre oblique inversée (backslash) suivie d'un espace vide pour laisser un espace vide.

Vous pouvez également faire des tableaux plus compliqués et les référencer :

```

.. my_drawn_table:

+-----+-----+
| Windows      | macOS      |
+-----+-----+
| |win|        | |osx|      |
+-----+-----+
| and of course not to forget |nix| |
+-----+-----+

My drawn table, mind you this is unfortunately not regarded as a caption

You can reference it like this: my_drawn_table_.
    
```

Le résultat :

Windows	macOS
	
et bien sûr, ne pas oublier 	

Ma table dessinée, remarquez que ce n'est malheureusement pas considéré comme une légende

Vous pouvez le référencer comme ceci *my\_drawn\_table*.

Pour des tables encore plus complexes, il est plus simple d'utiliser `list-table`:

```

.. list-table::
:header-rows: 1
:widths: 20 20 20 40

* - What
  - Purpose
  - Key word
  - Description
* - Test
  - ``Useful test``
  - complexity
  - Geometry. One of:
    
```

(suite sur la page suivante)

(suite de la page précédente)

```
* Point
* Line
```

Le résultat :

Quoi	Fonction	Mot-clé	Description
<b>Test</b>	Test utile	complexité	Géométrie. Un des: <ul style="list-style-type: none"> <li>• Point</li> <li>• Ligne</li> </ul>

## 2.1.6 Index

Un index est un moyen pratique pour aider le lecteur à trouver des informations dans un document. La documentation QGIS fournit quelques indices essentiels. Il existe quelques règles qui nous aident à fournir un ensemble d'indices vraiment utiles (cohérents, consistant et vraiment connectés les uns aux autres):

- Un index doit être lisible par l'homme, compréhensible et traduisible; un index peut être créé à partir de nombreux mots, mais vous devez éviter tout caractère inutile `_`, ``` - ... pour les lier, c'est-à-dire `Chargement des couches` au lieu de `chargement_couches` ou `chargementCouches`.
- Ne mettez en majuscule que la première lettre de l'index, sauf si le mot a une orthographe particulière. Par exemple, `Chargement des couches`, `Génération Atlas`, `WMS`, `pgsql2shp`.
- Gardez un œil sur la [Liste d'index](#) existante afin de réutiliser l'expression la plus pratique avec l'orthographe correcte et d'éviter les doublons inutiles.

Plusieurs balises d'index existent dans RST. Vous pouvez utiliser la balise inline `:index:` dans le texte normal:

```
QGIS can load several :index:`Vector formats` supported by GDAL/OGR ...
```

Ou vous pouvez utiliser le balisage de niveau bloc `.. index::` qui renvoie au début du paragraphe suivant. En raison des règles mentionnées ci-dessus, il est recommandé d'utiliser la balise de niveau bloc:

```
.. index:: WMS, WFS, Loading layers
```

Il est également recommandé d'utiliser des paramètres d'index tels que `single`, `pair` et `see`, afin de construire une table d'index plus structurée et interconnectée. Voir [Génération d'index](#) pour plus d'informations sur la création d'index.

## 2.1.7 Commentaires Spéciaux

Parfois, vous souhaitez peut-être mettre l'accent sur certains points de la description, soit pour avertir, rappeler ou donner des conseils à l'utilisateur. Dans la documentation QGIS, nous utilisons des directives spéciales reST telles que `.. warning::`, `.. seealso::`, `.. note::` et `.. tip::`. Ces directives génèrent des cadres qui mettent en valeur vos commentaires. Voir [Balisage de niveau de paragraphe](#) pour plus d'informations. Un titre clair et approprié est requis pour les avertissements et les conseils.

```
.. tip:: **Always use a meaningful title for tips**
```

```
Begin tips with a title that summarizes what it is about. This helps
users to quickly overview the message you want to give them, and
decide on its relevance.
```

### 2.1.8 Extraits de code

Vous pouvez également donner des exemples et insérer des extraits de code. Dans ce cas, écrivez le commentaire sous une ligne avec la directive `::` insérée. Pour un meilleur rendu, en particulier pour appliquer une surbrillance des couleurs au code en fonction de son langage, utilisez la directive de bloc de code, par ex. `.. code-block:: xml`. Plus de détails sur [Affichage du code](#).

---

**Note:** Bien que les textes des cadres de notes, de conseils et d'avertissement soient traduisibles, sachez que les cadres de blocs de code ne permettent pas la traduction. Évitez donc les commentaires non liés au code et gardez les commentaires aussi courts que possible.

---

### 2.1.9 Notes de pied de page

Remarque: Les notes de bas de page ne sont reconnues par aucun logiciel de traduction et elles ne sont pas non plus correctement converties au format PDF. Donc, si possible, n'utilisez pas de notes de bas de page dans la documentation.

Pour créer une note de bas de page (montrant comme exemple<sup>1</sup>)

```
blabla [1]_
```

qui pointera vers :

## 2.2 Les Captures d'écrans

### 2.2.1 Ajouter de nouvelles captures d'écran

Voici quelques conseils pour créer de nouvelles captures d'écran attrayantes. Les images doivent être placées dans un dossier image (`img/`) qui se trouve dans le même dossier que le fichier de référence `.rst`.

- Vous pouvez trouver des projets QGIS préparés qui sont utilisés pour créer des captures d'écran dans le dossier `./Qgis-projects` de ce référentiel. Cela facilite la reproduction de captures d'écran pour la prochaine version de QGIS. Ces projets utilisent le QGIS [Sample Data](#) (alias Alaska Dataset), qui doit être placé dans le même dossier que le référentiel de documentation QGIS.
- Réduisez la fenêtre à l'espace minimal nécessaire pour afficher la fonctionnalité (prendre tout l'écran pour une petite fenêtre modale est exagéré)
- Moins d'encombrement, mieux c'est (pas besoin d'activer toutes les barres d'outils)
- Ne les redimensionnez pas dans un éditeur d'images; la taille sera définie dans les fichiers `.rst` si nécessaire (réduction des dimensions sans augmenter correctement la résolution)
- Coupez l'arrière-plan
- Rendre les coins supérieurs transparents si l'arrière-plan n'est pas blanc
- Réglez la résolution de la taille d'impression sur `135 dpi` (par exemple, dans Gimp, définissez la résolution d'impression *Image ► Taille d'impression* et enregistrez). De cette façon, les images seront à leur taille d'origine en html et à une bonne résolution d'impression dans le PDF. Vous pouvez également utiliser la commande ImageMagick `convert` pour faire un lot d'images:

```
convert -units PixelsPerInch input.png -density 135 output.png
```

- Enregistrez-les sous `.png` (pour éviter les artefacts `.jpeg`)
- La capture d'écran doit montrer le contenu selon ce qui est décrit dans le texte

---

<sup>1</sup> Mise à jour des extensions principales

**Astuce:** Si vous utilisez Ubuntu, vous pouvez utiliser la commande suivante pour supprimer la fonction de menu global et créer des écrans d'application plus petits avec des menus:

```
sudo apt autoremove appmenu-gtk appmenu-gtk3 appmenu-qt
```

## 2.2.2 Captures d'écran traduites

Voici quelques conseils supplémentaires pour ceux qui souhaitent créer des captures d'écran pour un guide d'utilisateur traduit:

Les images traduites doivent être placées dans un dossier `img/<your_language>/`. Utilisez le même nom de fichier que la capture d'écran «originale» en anglais.

## 2.3 Documenter les algorithmes de traitement

Si vous souhaitez écrire de la documentation sur les algorithmes de traitement, tenez compte des recommandations suivantes:

- Les fichiers d'aide de l'algorithme de traitement font partie du Guide de l'utilisateur de QGIS, utilisez donc le même formatage que le Guide de l'utilisateur et toute autre documentation.
- Chaque documentation d'algorithme doit être placée dans le dossier **provider** et le fichier **group** correspondants, par ex. l'algorithme *Voronoi polygon* appartient au fournisseur *QGIS* et au groupe *vectorgeometry*. Le fichier correct pour ajouter la description est donc: `source/docs/user_manual/processing_algs/qgis/vectorgeometry.rst`.

**Note:** Avant de commencer à rédiger le guide, vérifiez si l'algorithme est déjà décrit. Dans ce cas, vous pouvez améliorer la description existante.

- Il est **extrêmement** important que chaque algorithme ait une *ancree* qui correspond au nom du fournisseur + le nom unique de l'algorithme lui-même. Cela permet au bouton Aide d'ouvrir la page d'aide de la bonne section. L'ancree doit être placée **au-dessus** du titre, par ex. (voir aussi la section *Étiquettes / références*)

```
.. _qgisvoronoipolygons:
Voronoi polygons
-----
```

Pour connaître le nom de l'algorithme, vous pouvez simplement passer la souris sur l'algorithme dans la boîte à outils Traitement.

- Évitez d'utiliser « Cet algorithme fait ceci et cela ... » comme première phrase de la description de l'algorithme. Essayez d'utiliser des expressions plus générales comme

```
Takes a point layer and generates a polygon layer containing the...
```

- Évitez de décrire ce que fait l'algorithme en répliquant son nom et veuillez ne pas répliquer le nom du paramètre dans la description du paramètre lui-même. Par exemple, si l'algorithme est un `polygone de Voronoï`, envisagez de décrire la couche d'entrée comme une couche à partir de laquelle calculer le polygone.
- Indiquez dans la description si l'algorithme a un raccourci par défaut dans QGIS ou prend en charge l'édition sur place.

- Ajouter des images ! Une image vaut mieux que mille mots ! Utilisez le format `.png` et suivez les instructions générales pour la documentation (voir la section *Figures et images* pour plus d'informations). Placez le fichier image dans le dossier correct, c'est-à-dire le dossier `img` à côté du fichier `.rst` que vous modifiez.
- Si nécessaire, ajoutez des liens dans la section « Voir aussi » qui fournissent des informations supplémentaires sur l'algorithme (par exemple, des publications ou des pages Web). N'ajoutez la section « Voir aussi » que s'il y a vraiment quelque chose à voir. Comme bonne pratique, la section « Voir aussi » peut être remplie de liens vers des algorithmes similaires.
- Expliquez clairement les paramètres et les sorties des algorithmes: inspirez-vous des algorithmes existants.
- Évitez de dupliquer la description détaillée des options de l'algorithme. Ajoutez ces informations dans la description du paramètre.
- Évitez d'ajouter des informations sur le type de géométrie vectorielle dans l'algorithme ou la description des paramètres, car ces informations sont déjà disponibles dans les descriptions des paramètres.
- Ajoutez la valeur par défaut du paramètre, par exemple:

```
* - Number of points
- ``NUMBER_OF_POINTS``
- [number]

Default: 1
- Number of points to create
```

- Décrivez le *type* d'entrée pris en charge par les paramètres. Il existe plusieurs types disponibles, vous pouvez en choisir un:

Paramètre/type de sortie	Description	Indicateur visuel
Couche vecteur de points	vecteur : point	
Couche vecteur de lignes	vecteur : ligne	
Couche vecteur de polygones	vecteur : polygone	
Couche vectorielle générique	vector: any	
Champ vecteur numérique	tablefield: numeric	<b>1.2</b>
Champ vecteur texte	tablefield: string	<b>abc</b>
Champ générique de vecteur	tablefield: any	
Couche raster	raster	
Bande raster	raster band	
Fichier HTML	html	
Couche de table	table	
Expression	expression	
Géométrie de point	coordinates	
Etendue	extent	
SCR	crs	
Enumeration	enumeration	
Liste	liste	
Nombre	number	<input type="text" value="1,00"/>
Caractère	string	Display name <input type="text" value="lakes.shp"/>
Booléen	boolean	<input checked="" type="checkbox"/>
Chemin d'accès au dossier	folder	
Fichier	file	
Table	matrix	
Couche	layer	

suite sur la page suivante

Table 2.1 – suite de la page précédente

Paramètre/type de sortie	Description	Indicateur visuel
Même type de sortie que le type d'entrée	same as input	
Definition	definition	
Point	point	
MultipleLayers	multipleLayers	
Plage	range	
AuthConfig	authconfig	
Mesh	mesh	
Mise en pagef	layout	
Element mise en page	layoutitem	
Couleur	color	
Échelle	scale	

- Étudiez un algorithme existant et bien documenté, et copiez toutes les dispositions utiles.
- Lorsque vous avez terminé, suivez simplement les instructions décrites dans *Contribuer étape par étape* pour valider vos modifications et effectuer une requete d'amélioration

Voici l'exemple d'un algorithme pour vous aider dans la mise en forme et la description

```
.. _qgiscountpointsinpolygon:

Count points in polygon
-----

Takes a point and a polygon layer and counts the number of points from the
point layer in each of the polygons of the polygon layer.
A new polygon layer is generated, with the exact same content as the input
polygon layer, but containing an additional field with the points count
corresponding to each polygon.

.. figure:: img/count_points_polygon.png
   :align: center

   The labels in the polygons show the point count

An optional weight field can be used to assign weights to each point.
Alternatively, a unique class field can be specified. If both options
are used, the weight field will take precedence and the unique class field
will be ignored.

``Default menu``: :menuselection:`Vector --> Analysis Tools`

Parameters
.....

.. list-table::
   :header-rows: 1
   :widths: 20 20 20 40

   * - Label
     - Name
     - Type
     - Description
   * - Polygons
     - ``POLYGONS``
     - [vector: polygon]
     - Polygon layer whose features are associated with the count of
       points they contain
   * - Points
     - ``POINTS``
```

(suite sur la page suivante)

```

- [vector: point]
- Point layer with features to count
* - Weight field

  Optional
  - ``WEIGHT``
  - [tablefield: numeric]
  - A field from the point layer.
    The count generated will be the sum of the weight field of the
    points contained by the polygon.
* - Class field

  Optional
  - ``CLASSFIELD``
  - [tablefield: any]
  - Points are classified based on the selected attribute and if
    several points with the same attribute value are within the
    polygon, only one of them is counted.
    The final count of the points in a polygon is, therefore, the
    count of different classes that are found in it.
* - Count field name
  - ``FIELD``
  - [string]

  Default: 'NUMPOINTS'
  - The name of the field to store the count of points
* - Count
  - ``OUTPUT``
  - [vector: polygon]

  Default: [Create temporary layer]
  - Specification of the output layer type (temporary, file,
  GeoPackage or PostGIS table).
  Encoding can also be specified.

Outputs
.....

.. list-table::
   :header-rows: 1
   :widths: 20 20 20 40

   * - Label
     - Name
     - Type
     - Description
   * - Count
     - ``OUTPUT``
     - [vector: polygon]
     - Resulting layer with the attribute table containing the
       new column with the points count

```

---

## Écriture de code dans le livre de recettes « PyQGIS Cookbook »

---

- *Comment écrire des extraits de code testables*
  - *Directives sphinx de Doctest*
  - *Tests groupés*
- *Comment tester des extraits sur votre ordinateur local*

Si vous envisagez d'ajouter ou de mettre à jour certains chapitres du PyQGIS-Developer-Cookbook, vous devez respecter certaines règles pour activer le test automatique des extraits de code.

Les tests sont vraiment importants car ils permettent un contrôle automatique du code. Les extraits de code avec des erreurs ou un code utilisant des méthodes obsolètes échoueront et la notification vous aidera à résoudre les problèmes.

Pour tester, nous utilisons l'extension [Sphinx doctest](#). Reportez-vous à la documentation de l'extension pour plus d'informations.

### 3.1 Comment écrire des extraits de code testables

L'écriture d'extraits de code testables n'est pas si différente de *l'ancienne* méthode. En gros, vous devez utiliser une directive Sphinx différente.

#### 3.1.1 Directives sphinx de Doctest

Au lieu d'intégrer le code dans une directive `.. code-block:: python` (qui mettrait automatiquement en évidence la syntaxe du code), vous devez maintenant l'intégrer dans un `.. testcode::`. C'est à la place de ceci

```
.. code-block:: python

    crs = QgsCoordinateReferenceSystem("EPSG:4326")
    assert crs.isValid()
```

Vous utilisez maintenant ceci

```
.. testcode::

    crs = QgsCoordinateReferenceSystem("EPSG:4326")
    assert crs.isValid()
```

Après avoir écrit l'exemple de code, vous devez ajouter une *assertion* qui évaluera le code et sera exécutée automatiquement.

Dans l'exemple ci-dessus, vous créez un fichier crs et avec `assert crs.isValid()`, vous **testez** s'il est valide. Si le code a une mauvaise syntaxe python ou si `crs.isValid()` renvoie `False`, cet extrait de code échouera pendant le test.

Pour réussir les tests sur les extraits, vous devez importer toutes les classes et déclarer toutes les variables utilisées dans les extraits de code. Vous pouvez les inclure dans l'extrait de code lui-même (visible dans les pages HTML) ou les ajouter à une directive `.. testsetup::` (masquée dans les pages HTML). Le `.. testsetup::` doit être placé avant le `.. testcode::`:

```
.. testsetup::

    from qgis.core import QgsCoordinateReferenceSystem

.. testcode::

    crs = QgsCoordinateReferenceSystem("EPSG:4326")
    assert crs.isValid()
```

Si l'extrait de code ne crée pas d'objets (et que vous ne pouvez donc pas utiliser quelque chose comme `assert object.isValid()`), vous pouvez tester le code à l'aide de la méthode `print()`, puis ajouter les résultats attendus dans une directive `.. testoutput::` pour comparer la sortie attendue

```
.. testcode::

    print("QGIS CRS ID:", crs.srsid())
    print("PostGIS SRID:", crs.postgisSrid())

.. testoutput::

    QGIS CRS ID: 3452
    PostGIS SRID: 4326
```

Par défaut, le contenu de `.. testoutput::` est affiché dans la sortie HTML. Pour le cacher du HTML, utilisez l'option `:hide:`

```
.. testoutput::
    :hide:

    QGIS CRS ID: 3452
    PostGIS SRID: 4326
```

---

**Note:** Si l'extrait de code contient des instructions d'impression, vous **DEVEZ** ajouter un `testoutput` avec les résultats attendus; sinon le test échouera.

---

### 3.1.2 Tests groupés

Pour chaque document « rst », les extraits de code sont testés séquentiellement, ce qui signifie que vous pouvez utiliser un `.. testsetup::` pour tous les extraits de code suivants et que les extraits suivants auront accès aux variables déclarées dans les précédents.

Vous pouvez également utiliser des groupes pour décomposer les exemples de la même page dans différents tests.

Vous ajoutez l'extrait de code aux groupes en ajoutant un ou plusieurs noms de groupe (séparés par des virgules) dans la directive respective

```
.. testcode:: crs_crsfromID [, morenames]

    crs = QgsCoordinateReferenceSystem("EPSG:4326")
    assert crs.isValid()
```

doctest va choisir chaque groupe de codes et les exécuter indépendamment.

**Note:** Utilisez des noms de groupe qui ont un sens avec le contenu associé. Utilisez quelque chose du type `<chapter>_<subchapter>`, par exemple: `crs_intro`, `crs_fromwkt`. En cas d'échec, cela aidera à identifier où les échecs se produisent.

Si vous ne déclarez aucun groupe, l'extrait de code sera ajouté à un groupe nommé `default`. Si au lieu de cela, vous utilisez `*` comme nom de groupe, l'extrait de code sera utilisé dans tous les groupes de test, ce qui est normalement utile dans la configuration de test

```
.. testsetup:: *

    from qgis.core import QgsCoordinateReferenceSystem
```

## 3.2 Comment tester des extraits sur votre ordinateur local

**Note:** Les instructions sont valables pour le système Linux.

Pour tester les extraits de code Python, vous avez besoin d'une installation *QGIS*. Pour cela, il existe de nombreuses options. Vous pouvez :

- Utilisez votre installation *QGIS* avec *Sphinx* à partir d'un environnement virtuel Python:

```
make -f venv.mk doctest
```

- Utilisez une installation manuelle de *QGIS*. Vous aurez besoin de:

1. Créez une extension Makefile personnalisée en haut du fichier `venv.mk`, par exemple un fichier `user.mk` avec le contenu suivant:

```
# Root installation folder
QGIS_PREFIX_PATH = /home/user/apps/qgis-master

include venv.mk
```

Ou

```
# build output folder
QGIS_PREFIX_PATH = /home/user/dev/QGIS-build-master/output

include venv.mk
```

2. Ensuite, utilisez-le pour lancer la cible `doctest`

```
make -f user.mk doctest
```

- Exécutez la cible `doctest` à l'intérieur de l'image officielle du docker *QGIS*:

```
make -f docker.mk doctest
```

Vous devez d'abord installer [Docker](#) parce que cela utilise une image docker avec QGIS dedans.

---

## Recommandations pour la traduction

---

- *Processus de traduction*
- *Traduire un fichier*
  - *Traduire avec Transifex*
  - *Traduire avec Qt Linguist*
  - *Traduire un manuel*
  - *Résumé des règles de traduction*

Ce manuel a pour but d'aider le traducteur. D'abord il présente de façon technique le processus général de réalisation d'une traduction. Ensuite, il explique la mise en traduction d'un document rst de l'anglais vers le néerlandais et enfin, un résumé des *Règles de traduction* est proposé.

---

**Note:** Bien que ces recommandations se focalisent sur la documentation de QGIS, les méthodes et règles décrites ci-dessous s'appliquent également à la traduction de l'interface et du site web de QGIS.

---

### 4.1 Processus de traduction

La documentation de QGIS est écrite en anglais dans des fichiers `.rst`. Pour fournir des traductions :

1. Un script pré-préparé génère les fichiers de traduction nommés fichiers `.po` pour la langue anglaise, dans le répertoire `/QGIS-Documentation/locale/en`.
2. Ces fichiers "originaux" sont ensuite copiés par le script dans les répertoires `locale` des autres langues.
3. Les phrases dans les fichiers `.po` sont envoyées sur la plateforme web Transifex et sont disponibles pour les traducteurs qui peuvent commencer à traduire de l'anglais vers leur langue avec l'éditeur.
4. A la fin de la journée, un script récupère les traductions validées.
5. A la prochaine compilation de la documentation (qui a lieu au moins une fois par jour), un script réutilise les phrases pour créer la documentation traduite.

6. Quand, ensuite, un document `.rst` est mis à jour, un nouveau fichier `.po` est créé dans la partie anglaise. Le contenu de ce fichier sera comparé avec chaque fichier `.po` existant, pour chaque langage. Cela signifie que lorsqu'une ligne est ajoutée à un document `.rst` déjà traduit, seules les phrases ajoutées/modifiées seront intégrées dans le fichier `.po` et auront besoin d'être traduites. La quantité de travail nécessaire pour mettre à jour la traduction pour les prochaines versions de QGIS en sera relativement réduite.

---

**Note:** Le processus décrit ci-dessus est le même pour la traduction du site web de QGIS, du logiciel QGIS et de QGIS Server. La différence avec les applications est qu'au lieu de fichiers `.po`, toutes les chaînes à traduire dans les fichiers `.py`, `.cpp`, `.yaml` et d'autres sont chargées et récupérées depuis Transifex dans un unique fichier `.ts`.

---

Deux outils peuvent être utilisés pour effectuer des traductions :

- la [plateforme web Transifex](#), recommandée, il s'agit de la manière la plus simple de traduire QGIS car les étapes décrites ci-dessus sont réalisées de manière transparente et toutes les chaînes à traduire sont rassemblées en un endroit unique pour le traducteur. Ils n'a plus qu'à choisir un fichier et le traduire. Les fichiers traduits sont stockés sur la plateforme en attendant qu'une nouvelle version ne les mette à jour.
- [Qt Linguist](#), un outil de développement Qt qui demande au traducteur de charger localement les fichiers `.po` (ou `.ts`) depuis le code source, les traduire et les renvoyer.

Quel que soit l'outil de traduction choisi, les règles de traduction sont les mêmes.

## 4.2 Traduire un fichier

Pour expliquer le fonctionnement de la traduction, nous allons prendre l'extension « Carte de chaleur » pour exemple. Dans cet exemple, nous traduirons de l'anglais vers le néerlandais mais c'est pratiquement la même démarche pour d'autres documents, dans toutes les langues.

La source du document est disponible ici:

```
QGIS-Documentation/source/docs/user_manual/plugins/plugins_heatmap.rst
```

Pourquoi ai-je donc choisi ce document?

1. Il contient des images, des captures d'écran, des en-têtes, des références et des substitutions.
2. Je l'ai rédigé donc il m'est plus facile de le traduire ;-)

Le processus de compilation a créé un fichier `.po` en anglais disponible ici :

```
QGIS-Documentation/locale/en/LC_MESSAGES/docs/user_manual/plugins/plugins_heatmap.  
↔po
```

L'équivalent néerlandais du fichier `.po` (en fait, une copie) est disponible ici :

```
QGIS-Documentation/locale/nl/LC_MESSAGES/docs/user_manual/plugins/plugins_heatmap.  
↔po
```

A côté de ce fichier se trouve fichier `.mo` très léger, indiquant qu'il ne contient encore aucune traduction.

## 4.2.1 Traduire avec Transifex

Pour traduire en utilisant Transifex, vous devez :

1. créer un compte sur Transifex et rejoindre le projet QGIS.
2. Dès que vous faites partie d'une équipe de traduction dans une langue, cliquez sur le projet correspondant (dans ce cas QGIS Documentation). La liste de toutes les langues et leur pourcentage de traduction est affichée.

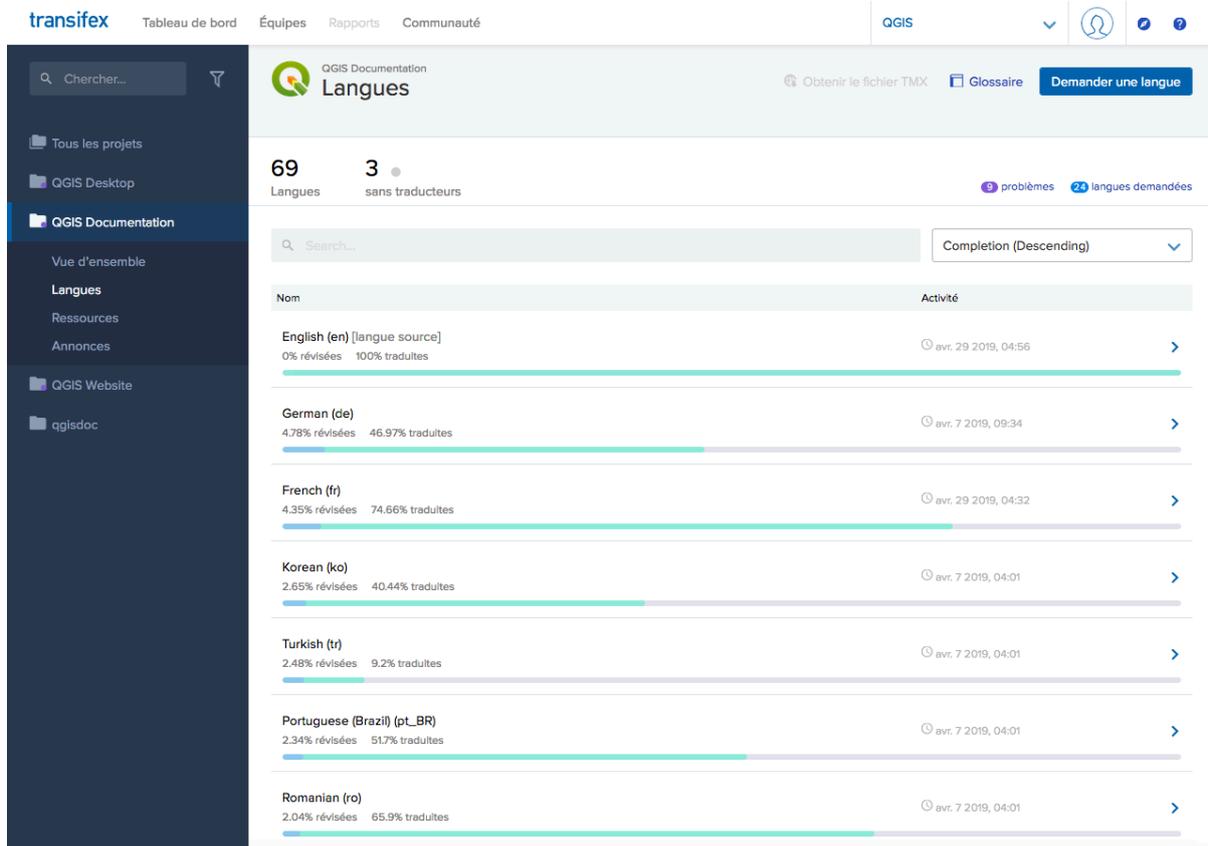


Figure4.1: Clic sur le menu Langues dans Transifex pour accéder aux traductions

3. Passez la souris sur votre langue de traduction et cliquez sur :
  - *Voir les ressources* : les fichiers de traduction .po et leur pourcentage de traduction, le nombre de chaînes et quelques autres métadonnées sont affichées.
  - ou sur *Traduire* : ouvre l'interface de traduction avec tous les fichiers .po disponibles.
4. Identifiez le fichier que vous souhaitez traduire (dans notre cas, nous cherchons le fichier d'extension carte de chaleur : docs\_user-manual\_plugins\_plugins-heatmap) ou sur n'importe quel fichier non terminé et cliquez dessus : les chaînes de ce fichier sont chargées et vous pouvez utiliser l'interface pour filtrer, traduire, suggérer des traductions...

---

**Astuce:** Pour la documentation ou le site web, cliquez sur le lien *Corrigez moi* dans le pied de page vous amène directement sur la page de traduction correspondante dans Transifex.

---

5. Tout ce que vous avez à faire est de sélectionner chaque texte et le traduire en suivant les *recommandations*.

Pour plus d'informations sur l'éditeur web Transifex, veuillez consulter <https://docs.transifex.com/translation/translating-with-the-web-editor>.

## 4.2.2 Traduire avec Qt Linguist

Avec Qt Linguist, vous devez :

1. récupérer manuellement les fichiers `.po` ou `.ts`. Vous pouvez le faire en téléchargeant les fichiers soit depuis la plateforme Transifex soit depuis le répertoire `locale/$language` du dépôt source (dans GitHub).
2. effectuer la traduction localement
3. charger les fichiers modifiés vers leur source (Transifex ou GitHub).

Le téléchargement et le chargement de fichiers de traduction peuvent être faits via Transifex mais ce procédé n'est pas recommandé. Comme il n'y a pas de système de versionnement dans Transifex, le fichier que vous allez charger remplacera simplement celui pré-existant et effacera potentiellement les modifications que d'autres utilisateurs auront pu faire sur la plateforme dans le même temps.

Quand vous ouvrez un fichier dans Qt Linguist pour la première fois, vous verrez la fenêtre suivante :



Figure4.2: Sélection de la langue de traduction dans le menu de Qt Linguist

La langue cible devrait être renseignée correctement. La langue d'origine peut être laissée sur POSIX de même que l'option Pays/Région peut être laissée sur Tout Pays.

Lorsque vous appuyez sur le bouton *OK*, Qt Linguist est rempli de phrases et vous pouvez commencer à traduire, voir Fig. 4.3.

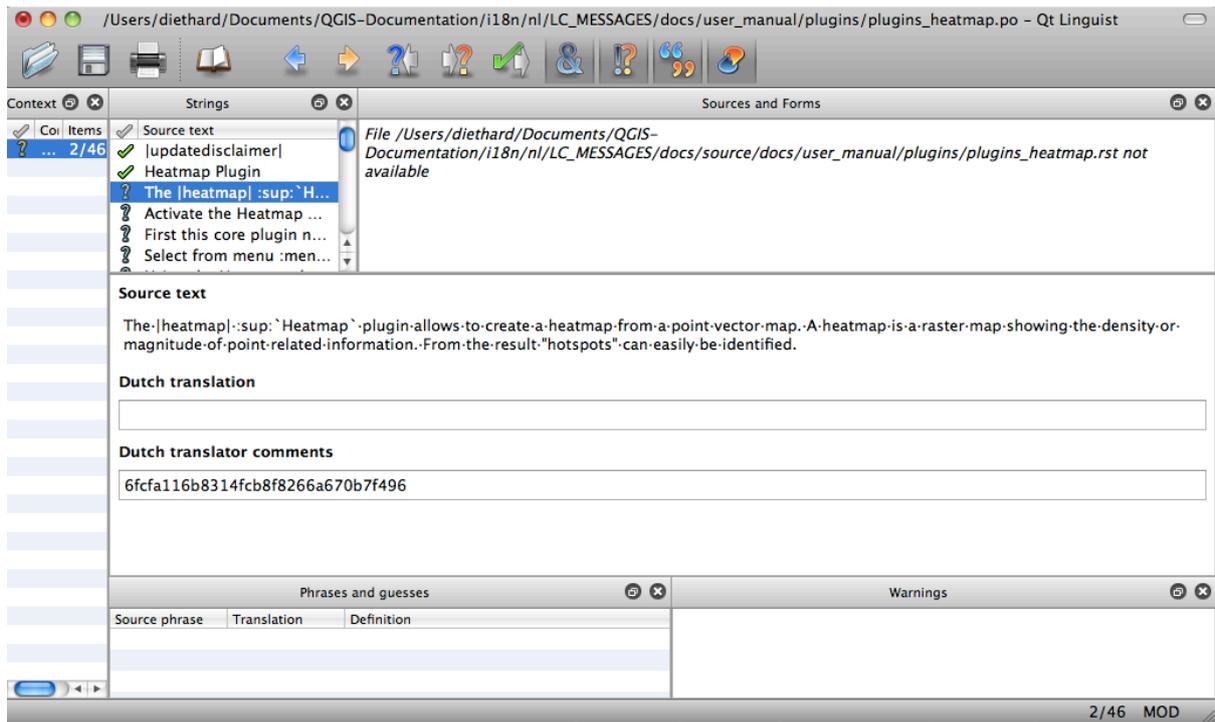


Figure4.3: Traduire avec le menu linguist

Dans le menu figurent les boutons suivants, pratiques à utiliser.

-  Le bouton Traduction Faîte, Au Suivant est le bouton le plus important. Si l'élément nécessite une traduction, on saisit la traduction dans le champ texte, puis on appuie sur ce bouton. Si l'élément ne nécessite aucune traduction, il suffit de laisser le champ texte vide, puis de cliquer sur ce bouton qui indiquera que l'élément a été traité et qui passera au suivant.
-  Le bouton Aller au précédent peut être utilisé pour aller à la traduction précédente.
-  Le bouton Aller au suivant peut être utilisé pour aller à la traduction suivante.
-  Le bouton Suivant A Faire saute sur le premier texte non encore traduit. Pratique lorsque le document original a subi des modifications et que seules quelques phrases nouvelles ou modifiées nécessitent d'être traduites.
-  Le bouton Précédent A Faire effectue une recherche vers l'arrière et saute sur le premier texte non encore traduit.

Pour plus d'informations sur l'utilisation de Qt Linguist, voir <https://doc-snapshots.qt.io/qt5-5.12/linguist-translators.html>

**Avertissement:** Si vous voulez télécharger du contenu pour le traduire depuis le dépôt source, ne le faites jamais dans la branche `master`. Il y a toujours une branche de traduction disponible, une fois que le document est complètement mis à jour en anglais pour une certaine version. Par exemple, pour traduire le manuel de QGIS 2.8, vous devez utiliser la branche `manual_en_v2.8`.

### 4.2.3 Traduire un manuel

Commençons maintenant la traduction du manuel de l'extension Carte de chaleur !

La traduction de la plupart des phrases devrait être aisée. Au cours de cette session de traduction, je veillerai à préciser quelles parties (instructions rst) nécessitent d'être traduites.

Ci-dessous, voyez une phrase intéressante à traduire :

```
The |heatmap| :sup:`Heatmap` plugin allows to create a heatmap from a point vector map. A heatmap is a raster map showing the density or magnitude of point related information. From the result "hotspots" can easily be identified.
```

Cette phrase contient deux instructions rst :

1. |heatmap| Les mots entre | sont des textes de substitution et ne devraient jamais être traduits. Celui-ci sera remplacé par l'icône de l'extension Carte de chaleur !
2. :sup:`Heatmap` : L'instruction :sup: est une instruction de superposition et affiche le texte suivant un peu en hauteur. Cela sert à afficher les textes de popup qui apparaissent lorsque vous déplacez le pointeur au-dessus de l'élément de la barre d'outils, et ceci peut être différent de la traduction faite dans l'application QGIS. Dans ce cas-ci, il ne l'est pas !

Tous les autres textes bruts dans cette phrase peuvent être traduits !

L'élément suivant à traduire contient l'instruction :ref: qui est communément utilisée pour faire référence à une autre section du manuel ! Le texte qui suit une instruction :ref: ne devrait jamais être modifié car il s'agit d'un identifiant unique !

```
First this core plugin needs to be activated using the Plugin Manager (see Section :ref:`load_core_plugin`). After activation the heatmap icon |heatmap| can be found in the Raster Toolbar.
```

Dans le cas présent, load\_core\_plugin est un code d'identification unique placé avant un élément rst ayant une légende. L'instruction ref sera remplacée par le texte d'en-tête et convertie en lien hypertexte. Lorsque l'en-tête auquel renvoie la référence est traduit, toutes les occurrences de la référence le sont automatiquement.

L'élément suivant contient la balise rst :menuselection: suivi du texte réellement affiché dans un menu de l'application QGIS ; ce texte peut être traduit dans l'application et par conséquent doit être modifié si tel est le cas.

```
Select from menu :menuselection:`View --> Toolbars --> Raster` to activate the Raster Toolbar when it is not yet activated.
```

Ci-dessus, l'élément « View -> » est en réalité traduit en « Beeld -> » parce que c'est la traduction utilisée dans la version en néerlandais de QGIS.

Un peu plus loin, nous voyons un point délicat de traduction :

```
The |heatmap| :sup:`Heatmap` tool button starts the Dialog of the Heatmap plugin (see :numref:`figure_heatmap_settings`).
```

Il contient une référence à une figure figure\_heatmap\_settings\_, et comme pour une référence de section, cette référence ne devrait pas être modifiée ! La définition de la référence depuis le document rst n'est pas incluse dans le fichier .po et ne peut par conséquent pas être modifiée. Cela signifie que la référence aux figures ne peut être traduite. Lorsqu'un la page HTML est créé, vous verrez figure\_heatmap\_settings. S'il s'agit d'un document PDF, figure\_heatmap\_settings sera remplacé par un numéro de figure.

L'élément suivant à traduire qui contient des attributs rst est :

```
**Input Point dialog**: Provides a selection of loaded point vector maps.
```

Ne pas enlever les étoiles dans la ligne. ci-dessus Il affiche le texte qu'elle contient en caractères gras. Le texte lui-même est souvent inclus dans la boîte de dialogue elle-même et peut bien être traduite dans l'application.

L'élément de traduction suivant contient la balise rst : `guilabel:`.

```
When the |checkbox| :guilabel:`Advanced` checkbox is checked it will
give access to additional advanced options.
```

Le texte `Advanced` de la balise `guilabel` pourrait bien être traduit dans l'application QGIS et devrait probablement être modifié !

L'élément de traduction suivant contient ```airports```. Les apostrophes sont utilisées afin de donner au texte un autre style de texte. Dans ce cas-ci, il s'agit d'une valeur littérale et cela ne requiert pas de traduction.

```
For the following example, we will use the ``airports`` vector point
layer from the QGIS sample dataset (see :ref:`label_sampledata`).
Another excellent QGIS tutorial on making heatmaps can be found on
`https://www.qgistutorials.com
<https://www.qgistutorials.com/en/docs/creating_heatmaps.html>`_.
```

Cet élément comprend également un lien hypertexte avec une url et un texte de présentation. L'url doit bien sûr être laissée intacte, mais vous êtes autorisé à modifier le texte externe `https://www.qgistutorials.com` qui est visible par le lecteur. Ne jamais supprimer le trait de soulignement à la fin du lien hypertexte qui forme une partie essentielle de celui-ci !

## 4.2.4 Résumé des règles de traduction

1. Ne modifiez pas le texte entre deux caractères « | » tels que `|bronze|`, `|checkbox|`, `|labels|`, `|selectString|`, `|addLayer|` ... Ce sont des tags spéciaux en remplacement d'images
2. Ne changez pas les références qui commencent par `:ref:`, `:file:`, `:numref:` sauf si elles incluent un titre. Dans ce cas, vous pouvez traduire le titre mais gardez intact le lien (c'est-à-dire le texte placé entre `<` et `>`)

**Astuce:** Lorsqu'un titre est fourni pour une référence, Transifex peut afficher un nombre dans le texte source en anglais à la place du lien. Cliquez sur le nombre dans la source pour ajouter la référence au lien à la suite du titre traduit.

3. Ne modifiez pas les textes finissant par un caractère underscore tels que `figure_labels_1_`
4. Ne changez pas l'url dans les liens hypertextes, mais vous pouvez changer la description externe. Laissez le trait de soulignement à la fin du lien hypertexte, sans espace supplémentaire (`>`_``).
5. Changez le contenu des instructions `:index:`, `:sup:`, `:guilabel:` et `:menuselection:`, vérifiez si et comment il est traduit dans l'application QGIS. Ne traduisez pas le tag en lui-même.
6. Les textes entre double étoiles et double apostrophes indiquent souvent des valeurs ou des noms de champs, et nécessitent parfois d'être traduits.
7. Veillez à utiliser les mêmes (nombres de) caractères spéciaux que le fichier source, notamment ```, ````, `*`, `**`, `::`. Ceux-ci contribuent à la mise en forme de l'information fournie.
8. Ne débutez ni ne terminez un texte associé à des caractères spéciaux ou à des tags par un espace
9. N'ajoutez pas un saut de ligne à la fin de la traduction, sinon le text ne sera pas traduit lors de la génération du fichier html.

Conformez-vous aux règles présentées ci-dessus et la traduction du document sera réussie !

Pour toute question, veuillez contacter l'Equipe [QGIS Community](#) ou l'équipe [QGIS Traduction](#).



- *Usage*
- *Substitutions usuelles*
  - *Icônes de plateforme*
  - *Éléments de menu*
- *Icônes de la barre d'outils*
  - *Gestion de couche et aperçu*
  - *Fichier*
  - *Éditer*
  - *Identity result*
  - *Numérisation et numérisation avancée*
  - *Navigateur de cartes et Attributs*
  - *Selection and Expressions*
  - *Étiquettes et Diagrammes*
  - *Décorations*
  - *Help*
  - *Couleurs*
- *Autres icônes de base*
- *Table attributaire*
- *Projections et Géoréférencement*
- *Mise en page de cartes*
- *Propriétés des couches*
- *Extensions*
  - *Traitements*

- *Diverses extensions principales*
- *Intégration GRASS*

## 5.1 Usage

Pour faciliter l'utilisation d'icônes dans les manuels de QGIS, le fichier `/source/substitutions.txt` dans le dépôt [QGIS-Documentation](#) contient un texte de substitution pour chaque fichier d'icône. Certaines de ces substitutions sont listées ci-après. De fait, lorsque vous voulez utiliser une icône de QGIS dans la documentation, il y a de grandes chances qu'une solution de substitution existe déjà et puisse/doive être utilisée.

Si aucun remplacement n'existe :

1. vérifiez dans le dépôt de documentation si l'icône est disponible dans le dossier `/static/common`. S'il n'y a aucune image, alors il faudra trouver et copier l'image de l'icône depuis le [Dépôt QGIS](#) (souvent dans le répertoire [https://github.com/qgis/QGIS/blob/release-3\\_16/images/themes/default](https://github.com/qgis/QGIS/blob/release-3_16/images/themes/default)) et le coller (au format `.png`) dans le dossier `/static/common`. Par praticité et pour les mises à jour, il est recommandé de conserver les noms de fichiers autant que possible.
2. créez la référence à la substitution dans le fichier `/source/substitutions.txt` en suivant l'exemple ci dessous. Le texte de remplacement devra être en casse Camel :

```
.. |splitLayer| image:: /static/common/split_layer.png  
:width: 1.5em
```

3. lancez le script `scripts/find_set_subst.py` pour mettre à jour les définitions des substitutions dans les fichiers `rst` et inclure la ou les nouvelles substitution(s).
4. (facultatif mais souhaitable) ajoutez l'icône et sa référence dans la liste ci-dessous.

## 5.2 Substitutions usuelles

Ci-dessous quelques icônes et leur substitution quand on écrit de la documentation. Vous les trouverez dans de nombreux endroits dans les manuels.

### 5.2.1 Icônes de plateforme

Icône	Substitution	Icône	Substitution
	logo		
	kde		nix
	osx		win

## 5.2.2 Éléments de menu

Icône	Substitution	Icône	Substitution
	checkbox		unchecked
	radioButtonOn		radioButtonOff
	selectNumber		selectString
	selectColor		selectColorRamp
	tab		degrees
Display name <input type="text" value="lakes.shp"/>	inputText		slider

## 5.3 Icônes de la barre d'outils

### 5.3.1 Gestion de couche et aperçu

Icône	Substitution	Icône	Substitution
	dataSourceManager		
	addOgrLayer		addDb2Layer
	addRasterLayer		addMssqlLayer
	addDelimitedTextLayer		addSpatiaLiteLayer
	addPostgisLayer		addOracleLayer
	addAfsLayer		addAmsLayer
	addMeshLayer		
	addVectorTileLayer		addXyzLayer
	addVirtualLayer		addWmsLayer
	addWcsLayer		addWfsLayer
	newVectorLayer		newSpatiaLiteLayer
	newGeoPackageLayer		createMemory
	newVirtualLayer		
	dbManager		gdal
	geoPackage		spatialite
	virtualLayer		wms
	wcs		wfs
	dbSchema		
	inOverview		addAllToOverview
	removeAllOverview		removeLayer
	showAllLayers		hideAllLayers
	showMapTheme		showSelectedLayers
	hideSelectedLayers		hideDeselectedLayers
	toggleAllLayers		toggleSelectedLayers
	addLayer		
	indicatorTemporal		indicatorNonRemovable
	indicatorEmbedded		indicatorFilter
	indicatorMemory		indicatorNoCRS
	indicatorBadLayer		favourites

### 5.3.2 Fichier

Icône	Substitution	Icône	Substitution
	fileNew		fileOpen
	fileSave		fileSaveAs
	fileExit		

### 5.3.3 Éditer

Icône	Substitution	Icône	Substitution
	undo		redo
	editCopy		editPaste
	editCut		saveEdits
	editableEdits		
	circle2Points		circle2TangentsPoint
	circle3Points		circle3Tangents
	circleCenterPoint		ellipseCenter2Points
	ellipseCenterPoint		ellipseExtent
	ellipseFoci		rectangle3PointsDistance
	rectangle3PointsProjected		rectangleCenter
	rectangleExtent		regularPolygon2Points
	regularPolygonCenterCorner		regularPolygonCenterPoint

### 5.3.4 Identity result

Icône	Substitution	Icône	Substitution
	expandTree		collapseTree
	expandNewTree		formView
	deselectAll		editCopy
	filePrint		
	identifyByRectangle		identifyByPolygon
	identifyByFreehand		identifyByRadius

### 5.3.5 Numérisation et numérisation avancée

Icône	Substitution	Icône	Substitution
	cad		cadConstruction
	cadParallele		cadPerpendicular
	toggleEditing		allEdits
	tracing		snapping
	capturePoint		capturePolygon
	captureLine		deleteSelectedFeatures
	circularStringCurvePoint		circularStringRadius
	vertexTool		vertexToolActiveLayer
	moveFeature		moveFeatureCopy
	moveFeatureLine		moveFeatureCopyLine
	moveFeaturePoint		moveFeatureCopyPoint
	rotateFeature		rotatePointSymbols
	offsetCurve		offsetPointSymbols
	simplifyFeatures		reshape
	addRing		addPart
	fillRing		
	deleteRing		deletePart
	mergeFeatures		mergeFeatAttributes
	splitFeatures		splitParts
	reverseLine		

### 5.3.6 Navigateur de cartes et Attributs

Icône	Substitution	Icône	Substitution
	pan		panToSelected
	zoomIn		zoomOut
	zoomActual		zoomFullExtent
	zoomToLayer		zoomToSelected
	zoomLast		zoomNext
	refresh		
	identify		mapTips
	showBookmarks		newBookmark
	measure		measureArea
	measureAngle		
	newMap		new3DMap
	tiltUp		tiltDown
	3dNavigation		play

### 5.3.7 Selection and Expressions

Icône	Substitution	Icône	Substitution
	selectRectangle		selectPolygon
	selectFreehand		selectRadius
	selectAll		deselectAll
	invertSelection		expressionSelect
	deselectActiveLayer		algorithmSelectLocation
	selectAllTree		select
	formSelect		dataDefined
	expression		dataDefineOn
	dataDefineExpressionOn		dataDefineError
	dataDefineExpressionError		
	addExpression		
	expressionFilter		filterMap

### 5.3.8 Étiquettes et Diagrammes

Icône	Substitution	Icône	Substitution
	labeling		labelingNone
	labelingRuleBased		labelingObstacle
	piechart		diagramNone
abc	text		histogram
	stackedBar		
	annotation		textAnnotation
	formAnnotation		htmlAnnotation
	svgAnnotation		autoPlacement
	labelbackground	abc	labelbuffer
+abc <c	labelformatting		labelplacement
	labelshadow		render
abc /	labelcallout		
	labelAnchorCenter		labelAnchorCustom
	labelAnchorEnd		labelAnchorStart
abc ●	pinLabels	abc ●	showHideLabels
abc □	moveLabel	abc □	rotateLabel
abc ●	showPinnedLabels	abc □	showUnplacedLabel
abc □	changeLabelProperties		

### 5.3.9 Décorations

Icône	Substitution	Icône	Substitution
	copyrightLabel		addGrid
	titleLabel		northArrow
	scaleBar		addMap
	addImage		

### 5.3.10 Help

Icône	Substitution	Icône	Substitution
	helpContents		qgisHomePage
	success		
	helpSponsors		contextHelp

### 5.3.11 Couleurs

Icône	Substitution	Icône	Substitution
	colorBox		colorPicker
	colorSwatches		colorWheel

### 5.4 Autres icônes de base

Icône	Substitution	Icône	Substitution
	arrowDown		arrowUp
	signPlus		signMinus
	projectProperties		options
	interfaceCustomization		keyboardShortcuts
	copyrightLabel		northArrow
	scaleBar		tracking
	gpsImporter		gpsTrackBarChart
	folder		extents
	settings		start
	3dconfigure		deleteSelected
	browserExpand		browserCollapse
	codeEditor		

## 5.5 Table attributaire

Icône	Substitution	Icône	Substitution
	openTable		selectedToTop
	selectAll		invertSelection
	panToSelected		zoomToSelected
	copySelected		editPaste
	expressionSelect		deleteSelectedFeatures
	newAttribute		deleteAttribute
	newTableRow		calculateField
	refresh		formView
	conditionalFormatting		multiEdit
	dock		actionRun
	duplicateFeature		
	panTo		highlightFeature

## 5.6 Projections et Géoréférenceur

Icône	Substitution	Icône	Substitution
	geographic		crs
	customProjection		setProjection
	projectionDisabled		projectionEnabled
	transformation		
	georefRun		pencil
	linkQgisToGeoref		linkGeorefToQgis
	fullHistogramStretch		

## 5.7 Mise en page de cartes

Icône	Substitution	Icône	Substitution
	newLayout		layoutManager
	duplicateLayout		
	newReport		newPage
	atlasSettings		atlas
	filePrint		saveMapAsImage

suite sur la page suivante

Table 5.1 – suite de la page précédente

Icône	Substitution	Icône	Substitution
	saveAsSVG		saveAsPDF
	addBasicShape		addBasicCircle
	addBasicTriangle		addBasicRectangle
	addNodesShape		editNodesShape
	addPolygon		addPolyline
	addArrow		northArrow
	add3DMap		addMap
	addLegend		addHtml
	addManualTable		addTable
	addImage		addMarker
	label		scaleBar
	select		moveItemContent
	setToCanvasScale		setToCanvasExtent
	viewScaleInCanvas		viewExtentInCanvas
	raiseItems		lowerItems
	moveItemsToTop		moveItemsToBottom
	alignLeft		alignRight
	alignHCenter		alignVCenter
	alignTop		alignBottom
	resizeShortest		resizeTallest
	resizeNarrowest		resizeWidest
	resizeSquare		
	lockItems		unlockAll
	locked		unlocked
	lockedRepeat		lockedGray
	groupItems		

## 5.8 Propriétés des couches

Icône	Substitution	Icône	Substitution
	symbology		labeling
	sourceFields		general
	metadata		action
	display		rendering
	join		diagram
	labelmask		temporal
	legend		dependencies
	3d		system
	editMetadata		overlay
	digitizing		auxiliaryStorage
	history		stylePreset
	search		pyramids
	transparency		rasterHistogram
	singleSymbol		nullSymbol
	graduatedSymbol		categorizedSymbol
	25dSymbol		ruleBasedSymbol
	invertedSymbol		heatmapSymbol
	pointDisplacementSymbol		pointClusterSymbol
	meshcontours		meshcontoursoff
	meshvectors		meshvectorsoff
	meshframe		
	sum		sort
	paintEffects		mapIdentification
	styleManager		iconView
	joinNotEditable		joinedLayerNotEditable
	joinHasNotUpsertOnEdit		filterTableFields
	symbologyEdit		
	sharingImport		sharingExport

## 5.9 Extensions

### 5.9.1 Traitements

Icône	Substitution	Icône	Substitution
	processing		processingModel
	processingHistory		processingResult
	processSelected		processingHelp
	saveAsPython		processingOutput
	qgsProjectFile		addToProject
	integer		
	mean		layerExtent
	randomSelection		vectorGrid
	convexHull		buffer
	intersect		union
	symDifference		clip
	difference		dissolve
	checkGeometry		exportGeometry
	delaunay		centroids
	toLines		extractNodes
	splitLayer		heatmap
	showRasterCalculator		showMeshCalculator

### 5.9.2 Diverses extensions principales

Fournies avec l'installation de base, mais non chargées lors de l'installation initiale

Icône	Substitution	Icône	Substitution
	showPluginManager		installPluginFromZip
	pythonFile		runConsole
	showEditorConsole		clearConsole
	offlineEditingCopy		offlineEditingSync
	plugin		interpolation
	gdalScript		metasearch
	geometryChecker		topologyChecker
	fromSelectedFeature		sqlQueryBuilder

### 5.9.3 Intégration GRASS

Icône	Substitution	Icône	Substitution
	grass		grassRegion
	grassTools		grassNewMapset
	grassOpenMapset		grassCloseMapset