

Using branches

Learn how to work with branches in Phrase localization projects.

About branching

Branching is a concept typically used in software version control. In branching copies of project files called *branches* allow teams to work on parallel versions of the project at the same time while retaining an unedited copy. This eliminates the risk of accidentally overwriting others' changes to project files.

Turn on branching in Phrase

You can turn on branching in the Advanced section of the Project Settings for any project by selecting "Enable branching".

The screenshot shows a 'Project Settings' dialog box with a sidebar on the left and a main content area on the right. The sidebar contains the following menu items: General, Placeholders, Project image, **Advanced** (highlighted in blue), Review Workflow (with a 'Beta' badge), Glossaries, In-Context-Editor, GitHub Sync, API, and Autopilot. The main content area is titled 'Advanced' and contains several settings:

- Enable branching**
You can enable the branching feature for this project.
 - Protect master branch**
Make sure that changes to the project can only be done in branches
- Enable machine translation**
Enable machine translation for this project. You can view and apply machine translations from the sidebar in the Translation Editor. To be able to use SmartSuggest and Autopilot machine translation must be enabled.
- Enable SmartSuggest**
SmartSuggest provides real-time suggestions based on machine translation and your glossary to speed-up translations and make them more consistent
 - Include machine translation** ⓘ
If machine translation is enabled for this project, you can use it with SmartSuggest.
 - Include glossary terms**
Enable to let SmartSuggest also consider terms in your glossary and include these entries in the suggestions
- Enable ICU Message format support**
We can validate and highlight your ICU messages [Learn more](#)
- Enable translators to edit translations of all types**
Otherwise, translators are not allowed to edit translations other than strings
- Enable 'ZERO' plural form**
Displays the input fields for the 'ZERO' plural form for every key as well although only some languages require the 'ZERO' explicitly.
- Share Translation Memory**
Include this project in the Translation Memory

A blue 'Save' button is located at the bottom right of the dialog box.

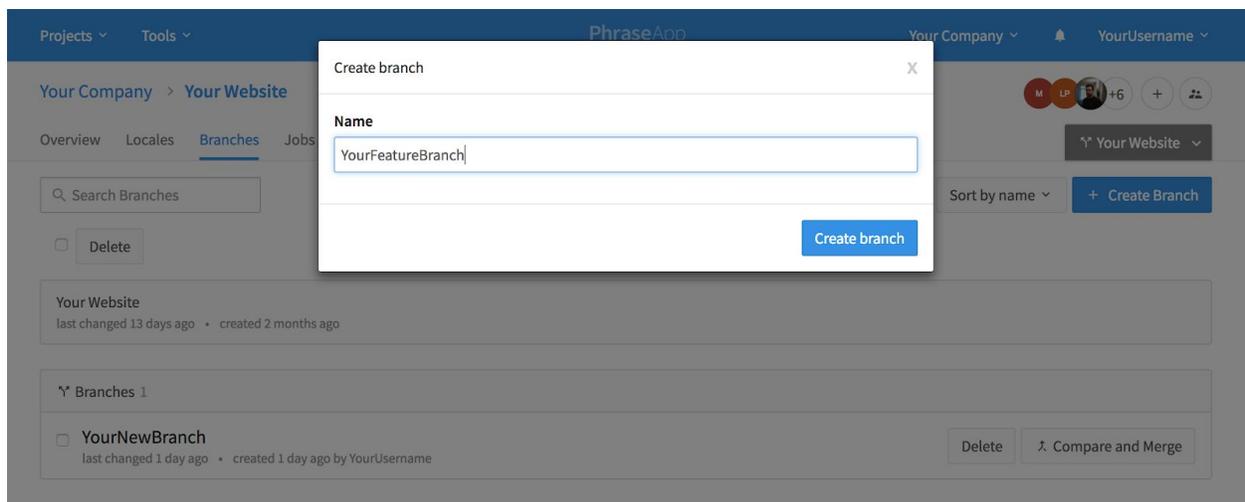
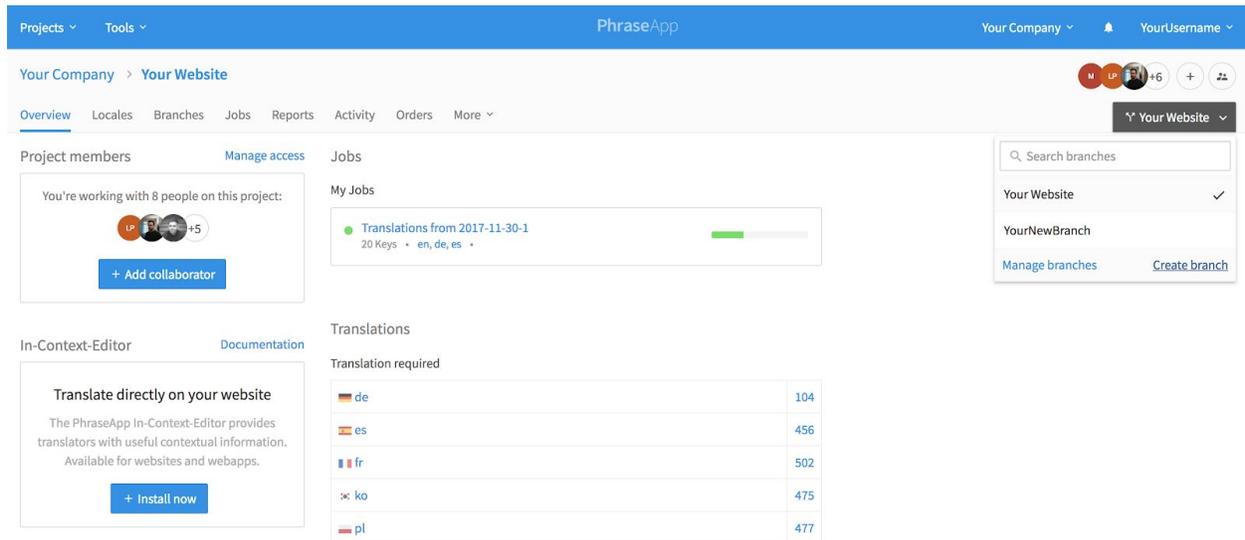
You can also select “Protect master branch.” If you select this option, you won’t be able to make any changes to the master branch directly. Instead, you can work in side branches and merge them into the master branch.

Working with branches

Create a new branch

Users with the Manager and Developer roles can create new branches from the Translation Center, the [API](#) or the [command-line client](#).

To create a new branch from the Translation Center, open the branch menu in the upper-right corner and choose “Create branch,” or switch to the Branches view and click Create Branch. After creating a branch, you can switch to it from the branch menu.



To create a new branch using the [API](#) or the [command-line client](#), run:

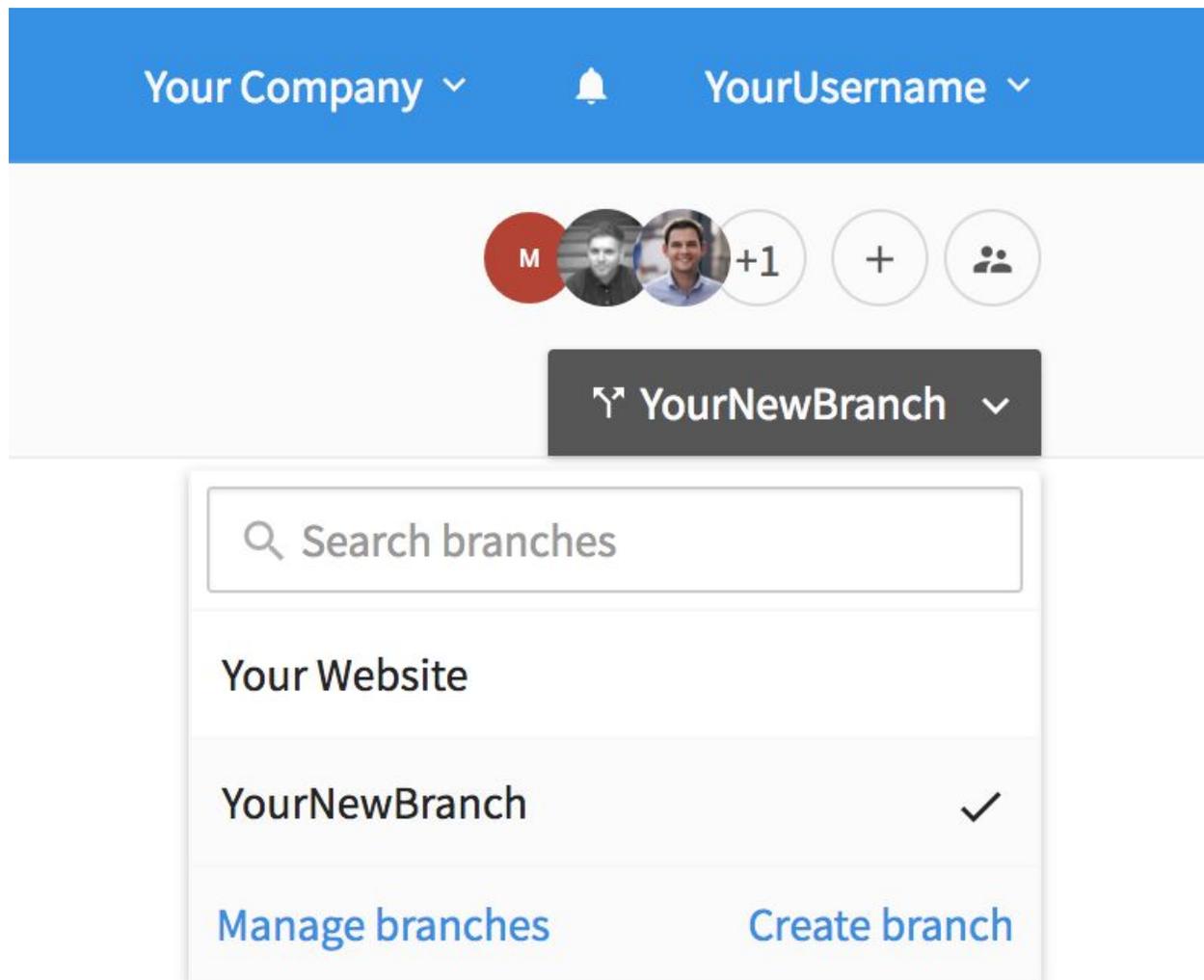
```
$ phraseapp branch create --name <branch_name>
```

or create a branch on the fly when pushing translations:

```
$ phraseapp push --branch <branch_name>
```

Change the active branch

You can easily switch between branches using the branch menu. The current branch is indicated with a check mark.



To specify a branch as a command-line argument, use the `--branch` parameter:

```
--branch <branch_name>
```

Manage branches

You can manage all the branches of your project from the Branches view. You can see when a branch was last changed, when it was created, who created it, and whether it has already been merged into the master branch. You can delete a branch using its Delete button, and you can merge a branch into the master branch using its Compare and Merge button.

To list all branches for a project using the [API](#) or the [command-line client](#), run:

```
$ phraseapp branches list <project_id>
```

To delete a branch using the [API](#) or the [command-line client](#), run:

```
$ phraseapp branch delete <project_id> <branch_name>
```

Merge branches

When you and your team are done making changes to a branch, you can merge it into the master branch to incorporate the new translations into it.

To merge a branch from the Translation Center, go to the Branches view for your project. In the entry for the branch you want to merge, click Compare and Merge. Phrase shows you an overview of the changes you'll make to the main branch, as well as any conflicts. To see details for a conflict, expand it using the arrow control on the right. To resolve conflicts, choose a merge strategy and click Merge Branch. If you want to reject the conflicting changes and preserve existing translations, choose Use Main Project. If you want to replace existing translations with changes from the branch, choose Use (Branch Name)..

Merge Branch YourNewBranch

What will be changed?

	Added	Modified	Deleted	Conflicts
Locales	1	1	0	✔ No Conflicts
Keys	1	0	1	✔ No Conflicts
Translations	3	1	6	❗ 2 Conflicts

Conflicts

❗ We have a problem merging the Branch YourNewBranch into the Master Project.

There are 2 Conflicts.

❗ 2 Translations changed in YourNewBranch and in the Main Project. ▾

	Main Project	YourNewBranch
application.application.navigation.demo en		
Content	Demopage	Demo Page
Unverified	⊘ No	✔ Yes
	last changed 8 minutes ago	last changed 9 minutes ago
application.application.navigation.demo de		
Content	Demo Seite	Demoseite
	last changed 10 minutes ago	last changed 10 minutes ago

Merge Changes into your Project

There are conflicts between YourNewBranch and the Main Project. How would you like to resolve them?

- Use Main Project
- Use YourNewBranch

Merge Branch

Even if there are no conflicts, you can see which translations will be modified during a merge and how by clicking a colored cell in the table.

To merge a branch using the [API](#) or the [command-line client](#), choose a merge strategy using the `--strategy` parameter:

```
$ phraseapp branch merge <project_id> <branch_name> --strategy <use_master/use_branch>
```

use_branch resolves conflicts by applying changes from the branch. use_master resolves conflicts by rejecting changes from the branch. If you don't specify a strategy, Phrase defaults to use_branch.

Note that only changes to the following resources are applied to the master branch during merges:

- Locales
- Keys (except screenshots)
- Translations

Changes to other resources, such as Activities, Jobs and Orders, are not applied.

Push and pull

When using the [command-line client](#), you can specify a branch to push to or pull from with the --branch parameter:

```
$ phraseapp push --branch <branch_name>
```

```
$ phraseapp pull --branch <branch_name>
```

If the specified branch doesn't exist in your project, the client will create it for you.

You can also use --use-local-branch-name to push and pull using the branch name of your active git branch:

```
$ phraseapp push --use-local-branch-name
```

```
$ phraseapp pull --use-local-branch-name
```

API

All API endpoints that take a project argument also support the --branch parameter to perform an action on a specific branch of a project:

```
$ phraseapp upload create <project_id> \
```

```
--branch <branch_name>
```

```
--file /path/to/my/file.json \
```

```
--file-format json \
```

```
--locale-id abcd1234cdef1234abcd1234cdef1234 \
```

```
--tags awesome-feature,needs-proofreading \
```

```
--locale-mapping '{"en": "2"}" \
```

```
--format-options '{"foo": "bar"}"
```

FAQs

To see a demo and learn more about the benefits of branching, you can watch our webinar on [Branching and Versioning](#). After entering your contact information, you'll be able to access the webinar.

What is the difference between the main project and a branch?

The main project is copied each time you create a new branch. Phrase doesn't treat the main project as a separate branch. Moreover, branches are always merged into the main project, not the other way around.

Are changes to a branch also applied to the main project?

Changes to a branch are only applied to the main project when you merge the branch into it.

Can several teams work on several branches at the same time?

Yes. Localization teams can work on several branches within Phrase just as developers might work on several branches within a software version control system.

Are there any limitations when working in a branch?

Yes. The following limitations on branches exist:

- Jobs created in a specific branch are not visible in the main project and vice versa. When you merge a branch into the main project, any open Jobs in it are lost.
- Statistics in the Reports view reflect only the status of the active branch. Merging a branch into the main project updates the main project's statistics.
- *The Activity view contains only activities in the active branch. Activities in branches aren't visible in the main project and vice versa. Activities in a branch are lost when merging it into the main project.*
- GitHub Sync cannot be used in branches. You can turn it on only for the main project.
- Comments made in a branch aren't visible in the main project and vice versa.

- *Orders* created in a branch aren't visible in the main project and vice versa. When you merge a branch into the main project, any open orders in it are lost.

Can a new branch automatically be created when pushing from the command line?

Yes. If you specify the name of a branch that doesn't exist during a push, the client creates a new branch for you:

```
$ phraseapp push --branch <branch_name>
```

How long does a branch exist?

A branch exists until you merge it into the main project and delete it. All merged branches are still visible in the Branches view, where they are marked as merged. You can't open a merged branch to work on it again. Instead, you can create a new branch from the main project.

Can I revert a merge?

No.

Can I undo deleting a branch?

No.

What is a conflict?

A *conflict* occurs when a resource (like a translation, key or locale) has changed in the main project since the time a branch was created. For example, translations and key settings (like pluralization) may have changed in the main project after you created a branch, so the values in your branch no longer match those in the main project.

-

Can a branch be merged into another branch?

No. You can only merge a branch into the main project.

Do I need to change my `.phraseapp.yml` config file when working with branches?

If you reference hashed locale IDs using the `locale_id` parameter, yes. Locales in branches have separate, unique IDs. You can work around this by using locale names instead of locale IDs.

How do translators know which branch to work on?

You can give translators permission to work on particular locales, but you can't restrict them to specific branches. Instead, tell translators explicitly which branch to work on and how to switch to it.

Why do I get an error message that says “Access Denied”?

To access the Branches view and manage branches, you need to have the Manager or Developer role in your project.

Can I use a different branch when working with the In-Context Editor?

Yes. You can initialize the In-Context Editor for a specific branch by adding a branch configuration value. To learn how, read about [how to set up the In-Context Editor](#).