

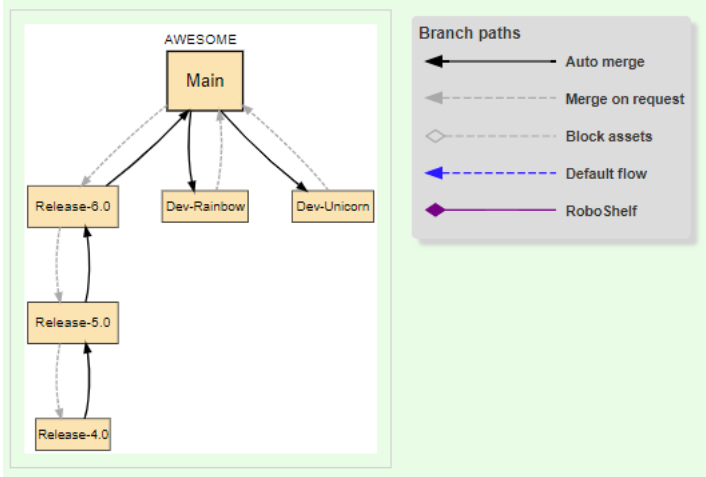
RoboMerge

Created by James Hopkin, last modified on Mar 13, 2019

ROBOMERGE

Usage

RoboMerge monitors commits to Perforce streams. It reads commit messages to find commands telling it to merge between branches, and some branches are set up to be automatically merged. For example:



A common pattern is to configure RoboMerge to merge commits to Main into release and feature branches. In the above example, commits to **Release-5.0** are automatically merged up to **Release-6.0**, and then up to **Main**. The merge paths for each of our development streams can be seen on the [RoboMerge admin page](#).

To request a changelist committed to Main to be merged down to **Release-6.0** and **Release-5.0**, the changelist description would include the line:

```
#robomerge Release-6.0, Release-5.0
```

Multiple branches can be separated by commas and/or spaces. Most branches are given short aliases, which can also be found on the admin page. This command could be simply written

```
#robomerge 5.0
```

using the 5.0 alias and allowing RoboMerge to calculate the route from Main to 5.0. On the admin page, hover over the box for a stream to see what aliases have been set up.

Conflicts

If Perforce encounters a conflict when RoboMerge is merging a changelist:

- An email is sent to the author, warning of the conflict. Robomerge will also reach out via Slack with some common options to address the conflict
- RoboMerge will perform no further merges on the stream until the author unshelves the changes (or uses the Create a Shelf feature), resolves all conflicts and commits the generated changelist.

The admin page shows which streams RoboMerge is currently blocked on, including the author responsible for resolving the conflicts.

Example cases

Most of the changes are fine but one file should be deadended:

Two possibilities:

1. Accept target on that file
2. Exclude the file from the shelf, check in, and hit ignore on that one file (this may cause a problem for subsequent check ins that modify this file)

Flags

The following flags can be added to RoboMerge commands.

Symbol	Name	Kind	Example	Description
-	skip	branch	#robomerge -7.40	Do not merge to the specified branch (used for branches that are merged to automatically)
ignore deadend		global	#robomerge ignore #robomerge deadend	Completely ignore this changelist - no automatic merges happen. Note that a target branch of 'ignore' is interpreted as an ignore flag.
#review		global	#robomerge #review 7.40	Add #codereview for owner for each commit in merge chain
#manual		global	#robomerge #manual	Do not commit merge - add #codereview and shelve for owner
!	null	branch	#robomerge !7.40	Perform a null merge to the specified branch, i.e. convince Perforce that a merge has happened, but don't actually make any changes to the content of the stream.

Symbol	Name	Kind	Example	Description
null		global	<code>#robomerge null</code>	Make every automatic merge of this commit a null merge (this is the old behavior of the deadend tag)

Note that

`#robomerge none`

is now an error, due to confusion surrounding its usage.