

ITKv4 Migration Guide

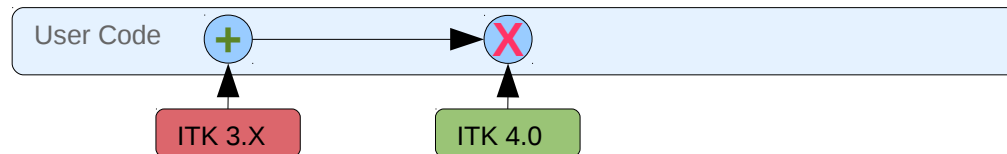
User Viewpoint

User Questions

- Why does ITKv4 Break my code?
- Where can I find fix information?
- How do I actually fix it?
- Any help for the easy ones?

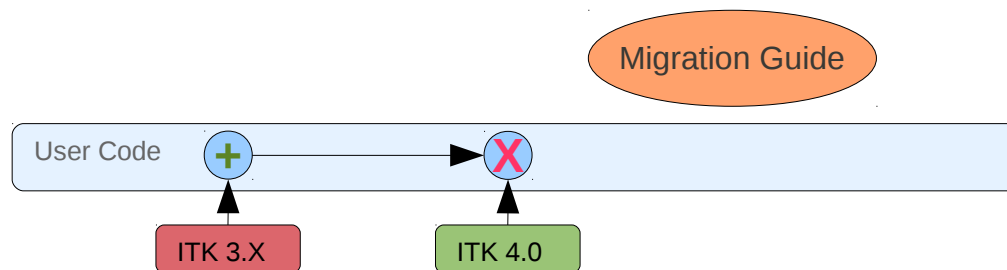
User Questions

- Why does ITKv4 Break my code?
 - Need plain description text of what changed and why
- Where can I find fix information?
- How do I actually fix it?
- Any help for the easy ones?



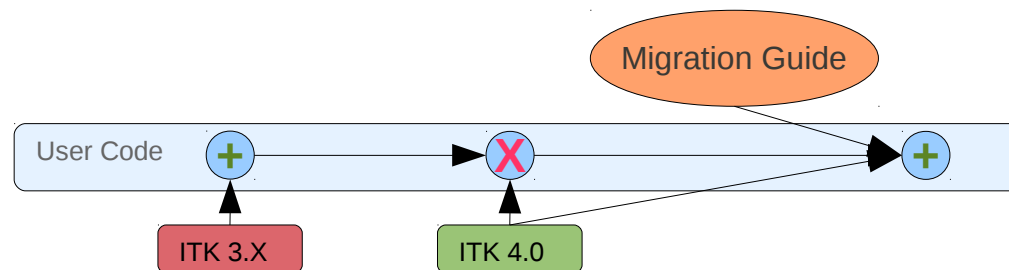
User Questions

- Why does ITKv4 Break my code?
- Where can I find fix information?
 - Google search for file names or compiler errors
 - Want them to find online guide (phpMyFAQ)
- How do I actually fix it?
- Any help for the easy ones?



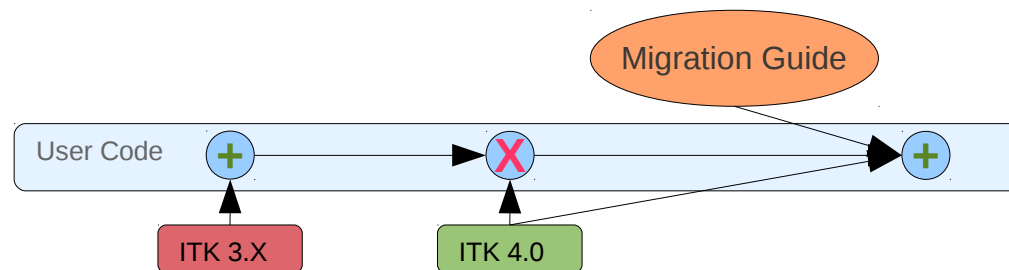
User Questions

- Why does ITKv4 Break my code?
- Where can I find fix information?
- How do I actually fix it?
 - Need simple sample code examples
 - Want detailed examples of real fixes
- Any help for the easy ones?

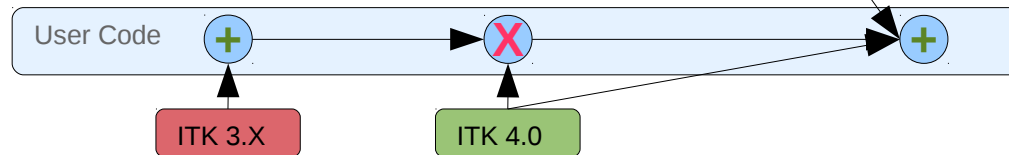
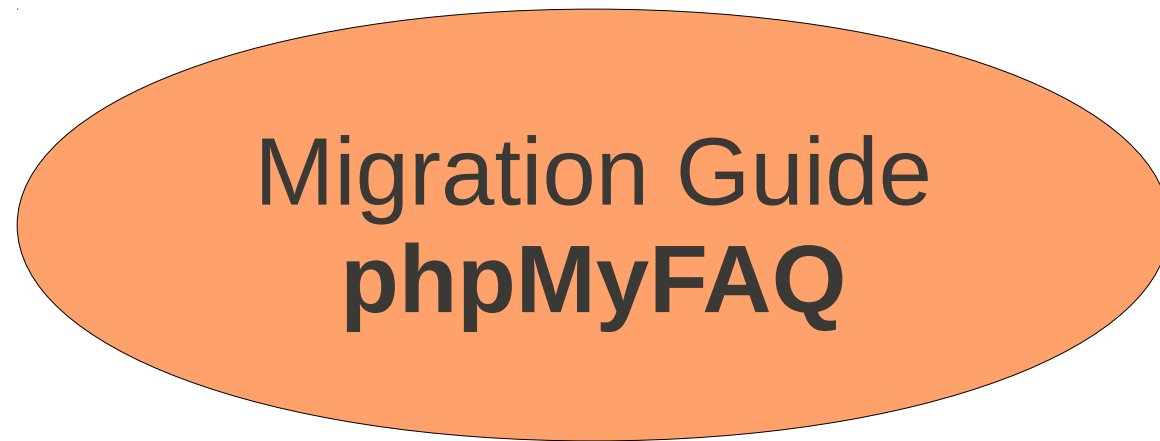


User Questions

- Why does ITKv4 Break my code?
- Where can I find fix information?
- How do I actually fix it?
- Any help for the easy ones?
 - Migration tool to do global find and replace
 - Try to help on harder ones (flag them) but no guarantee



Online Migration Guide



Developer Workflow

Sequestered Applications



ITKv4 Master

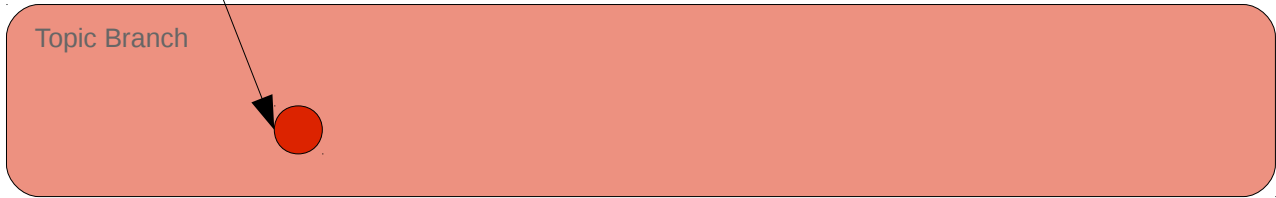


Gerrit Review



Published Migration Guide





Sequestered Applications

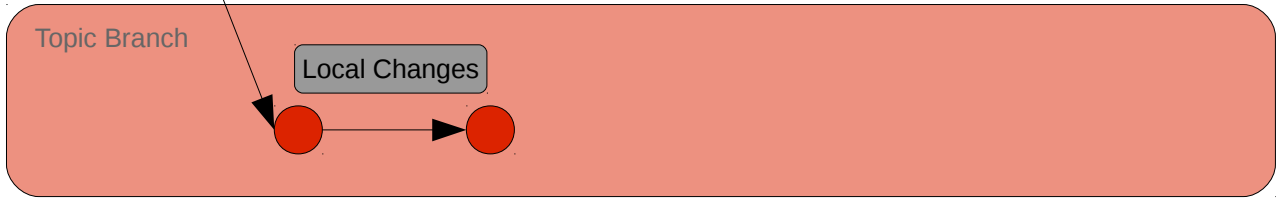
ITKv4 Master

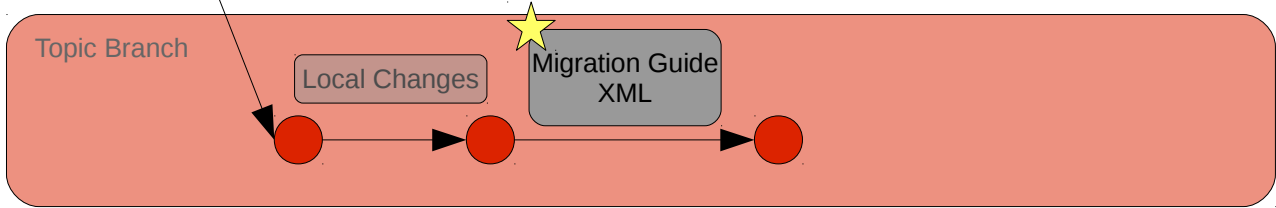
Gerrit Review

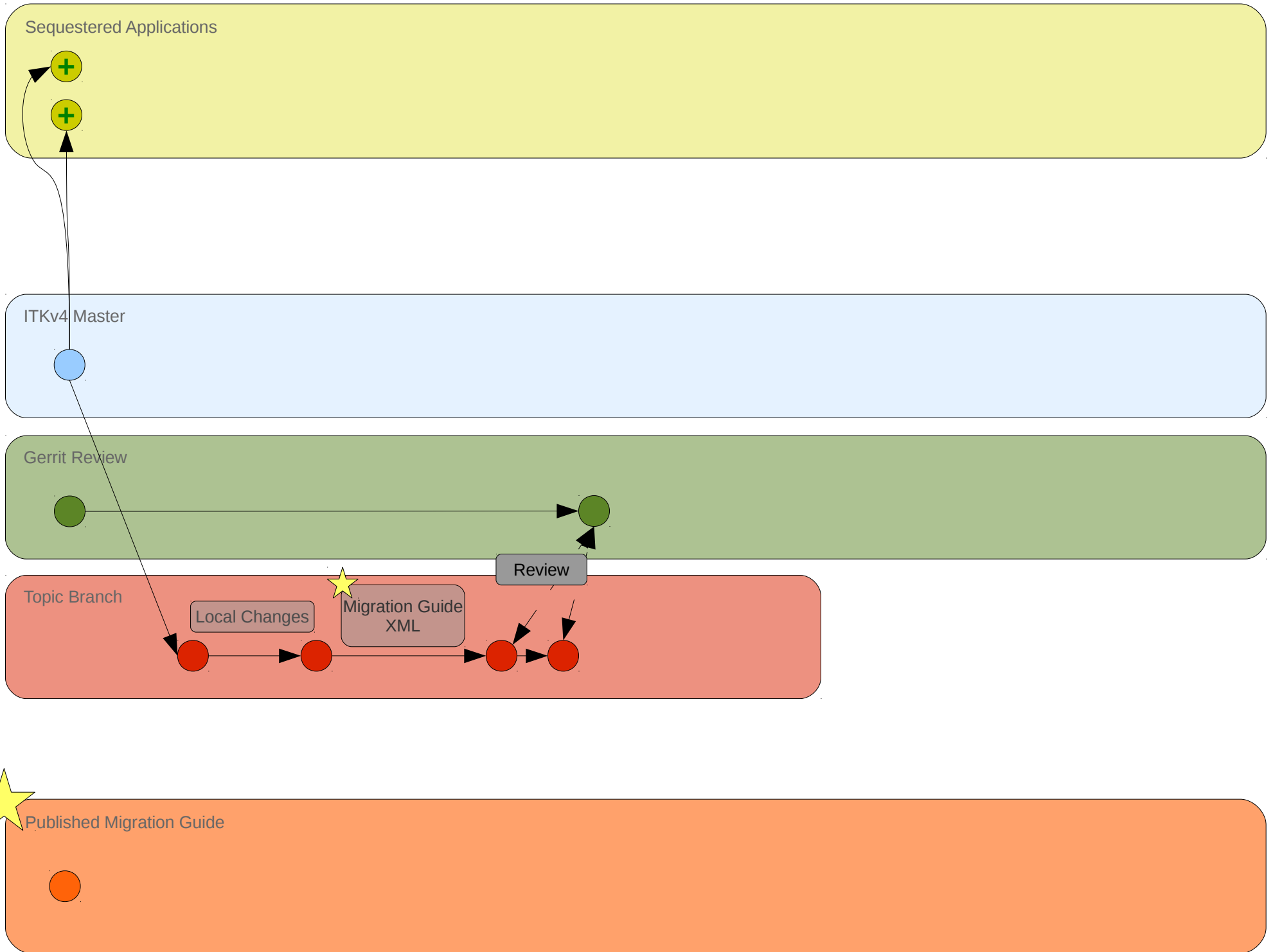
Topic Branch

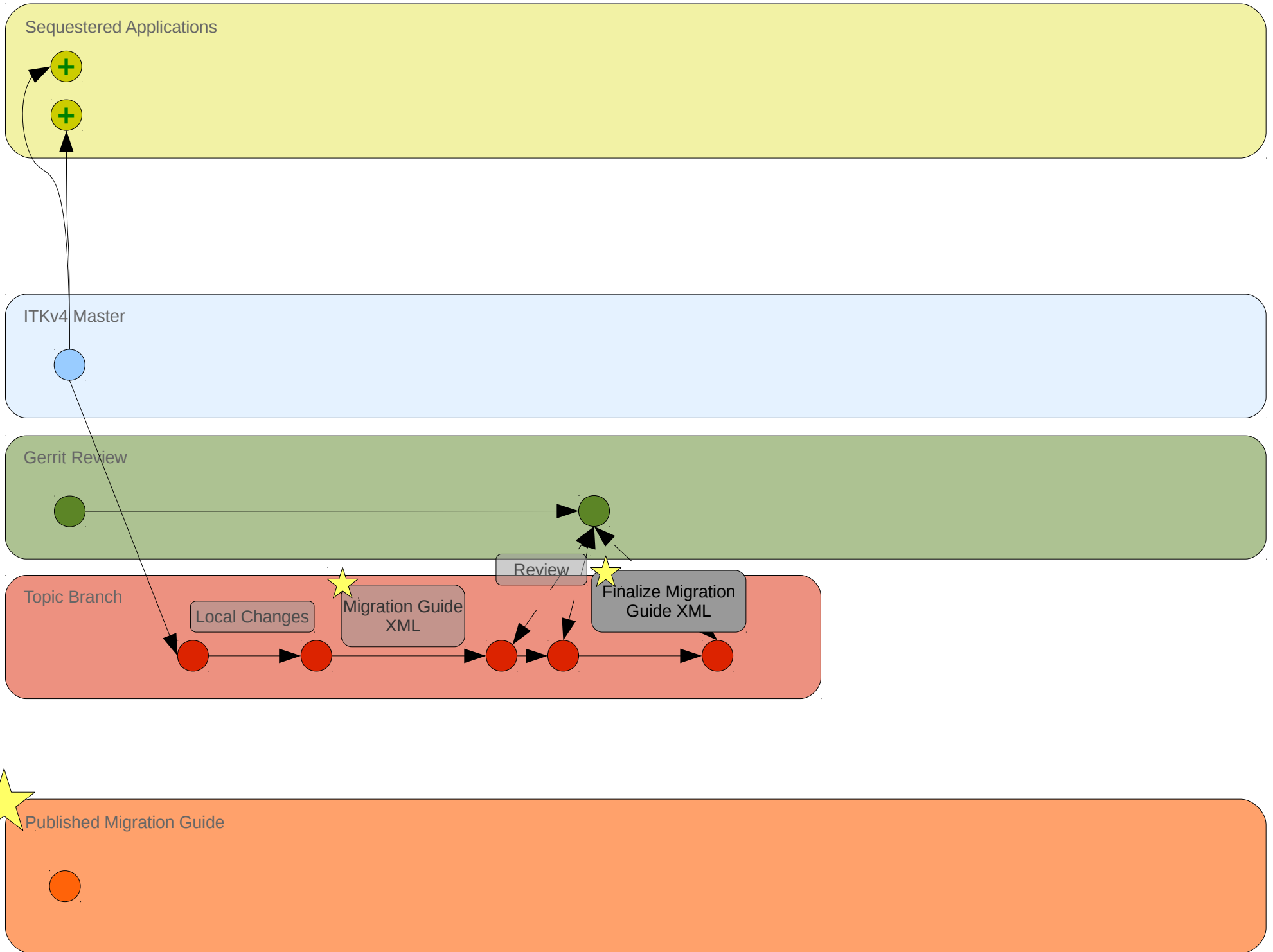
★ Published Migration Guide



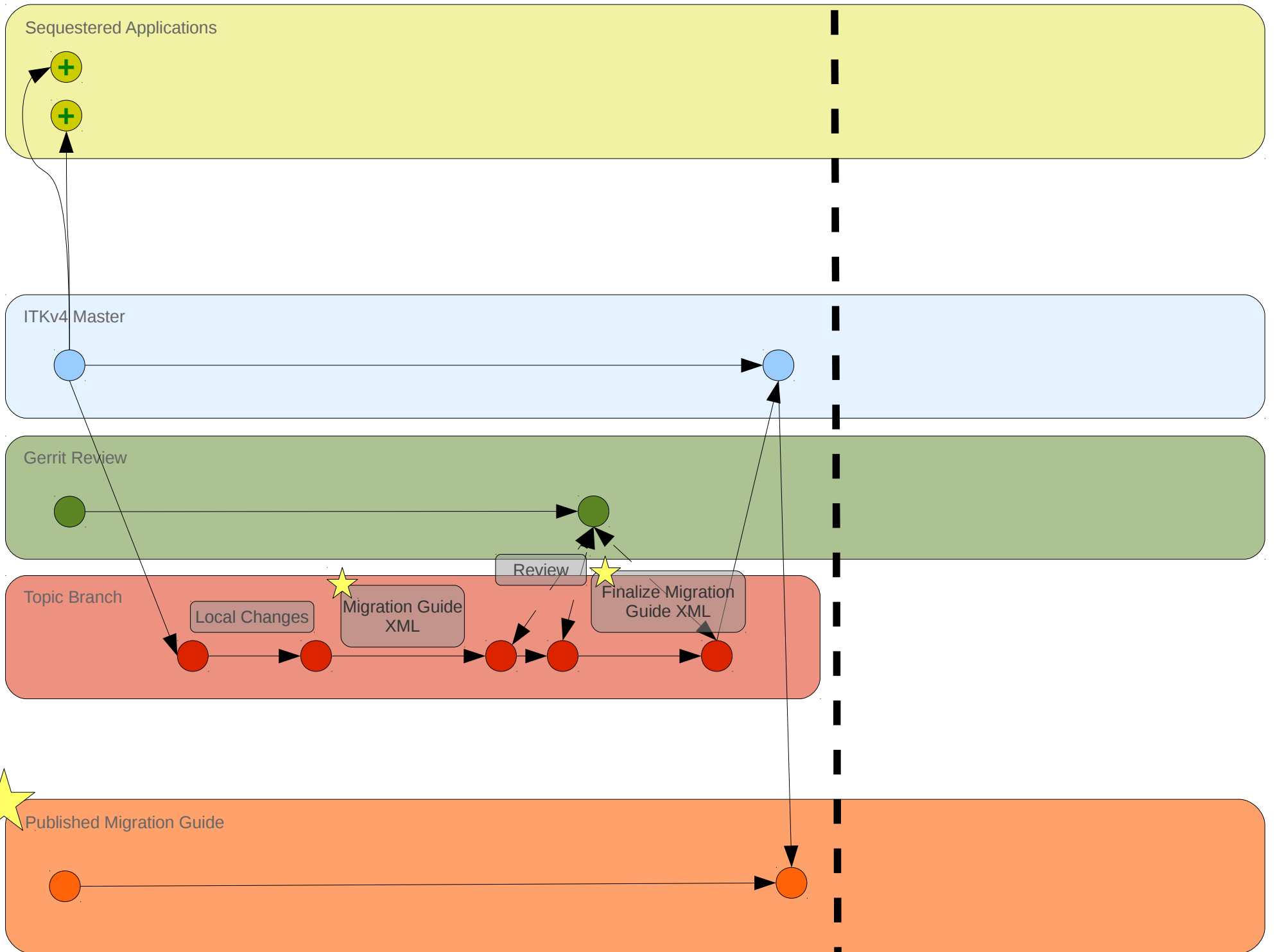






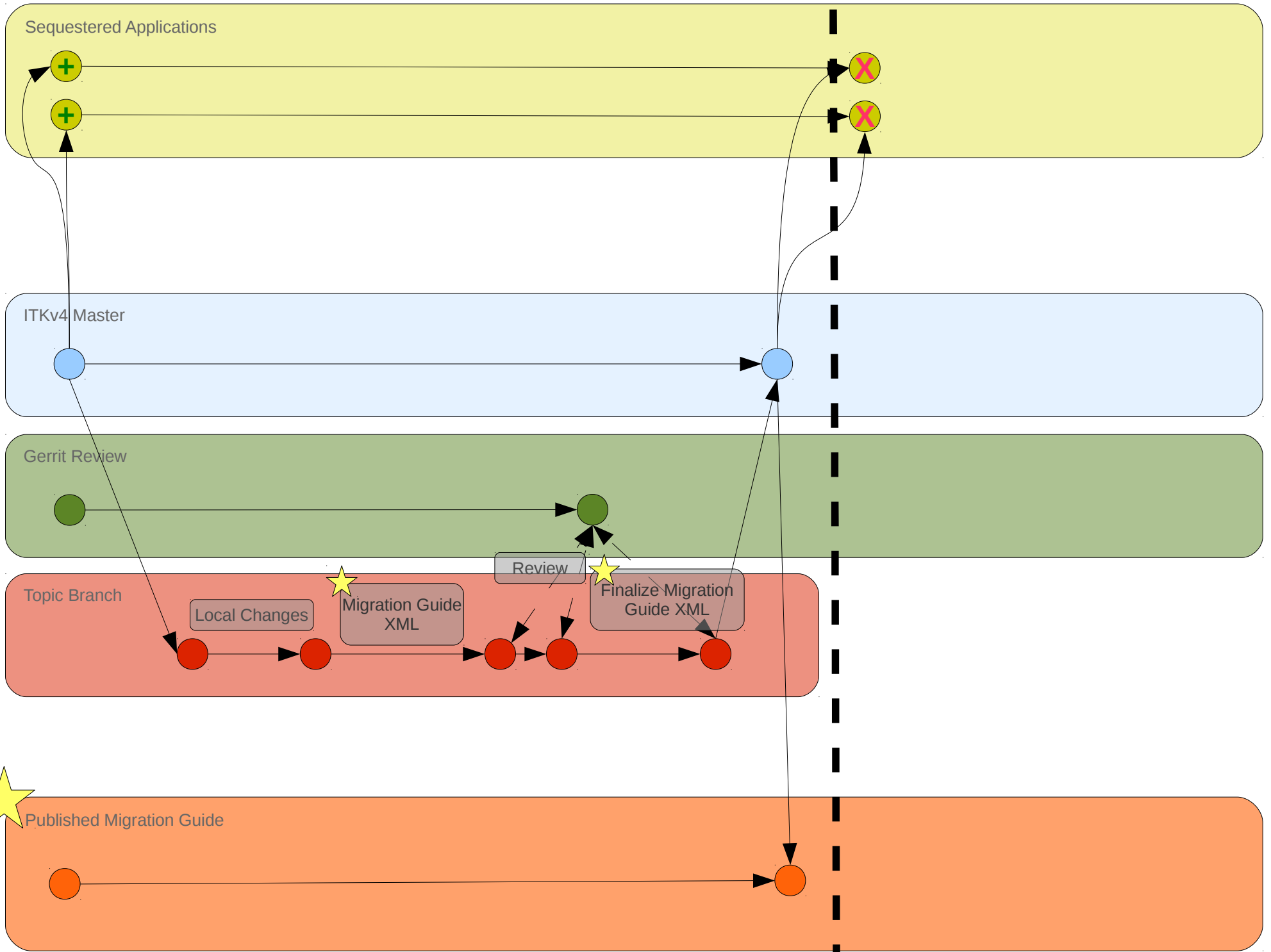


DAY 1



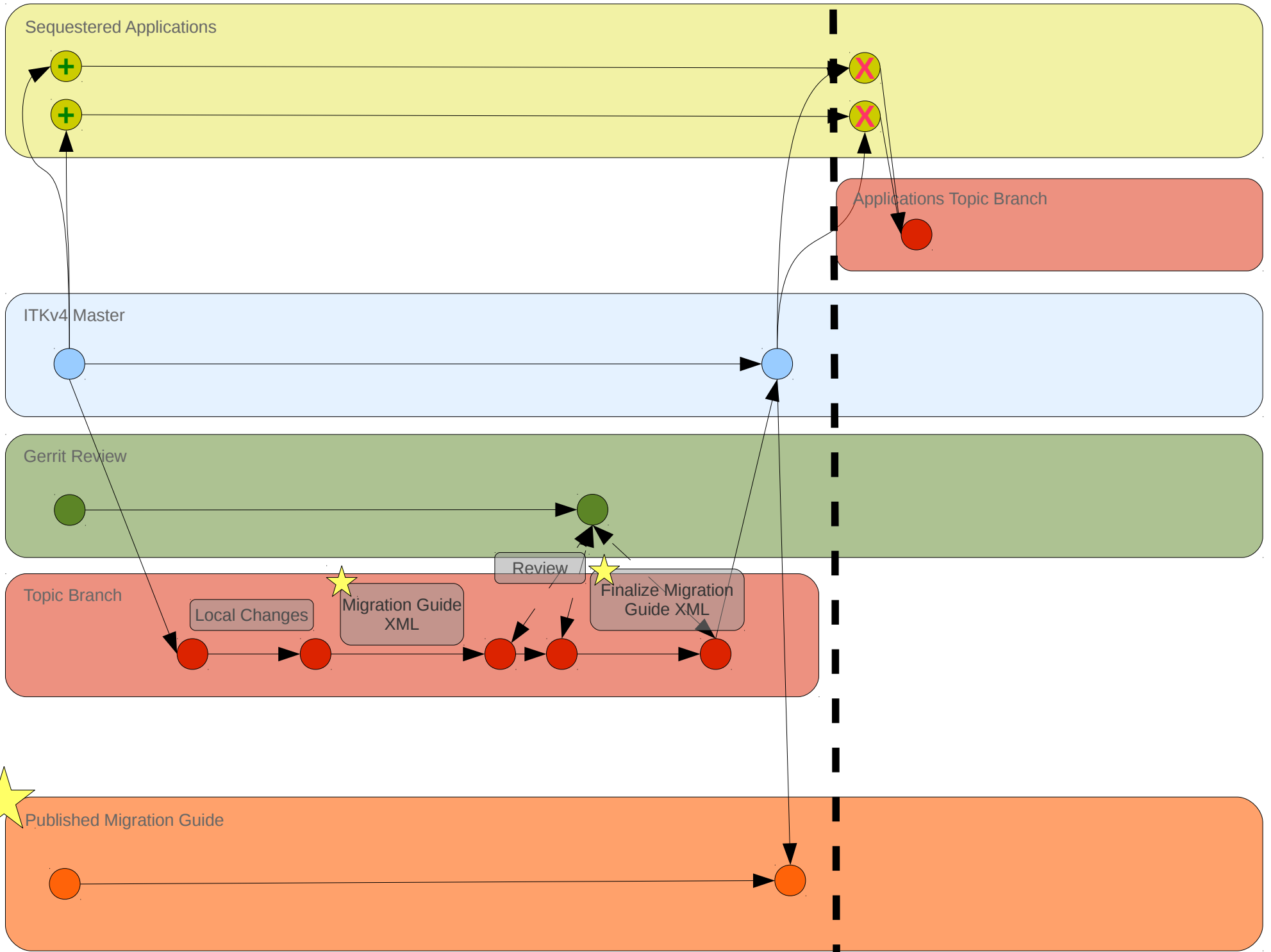
DAY 1

DAY 2



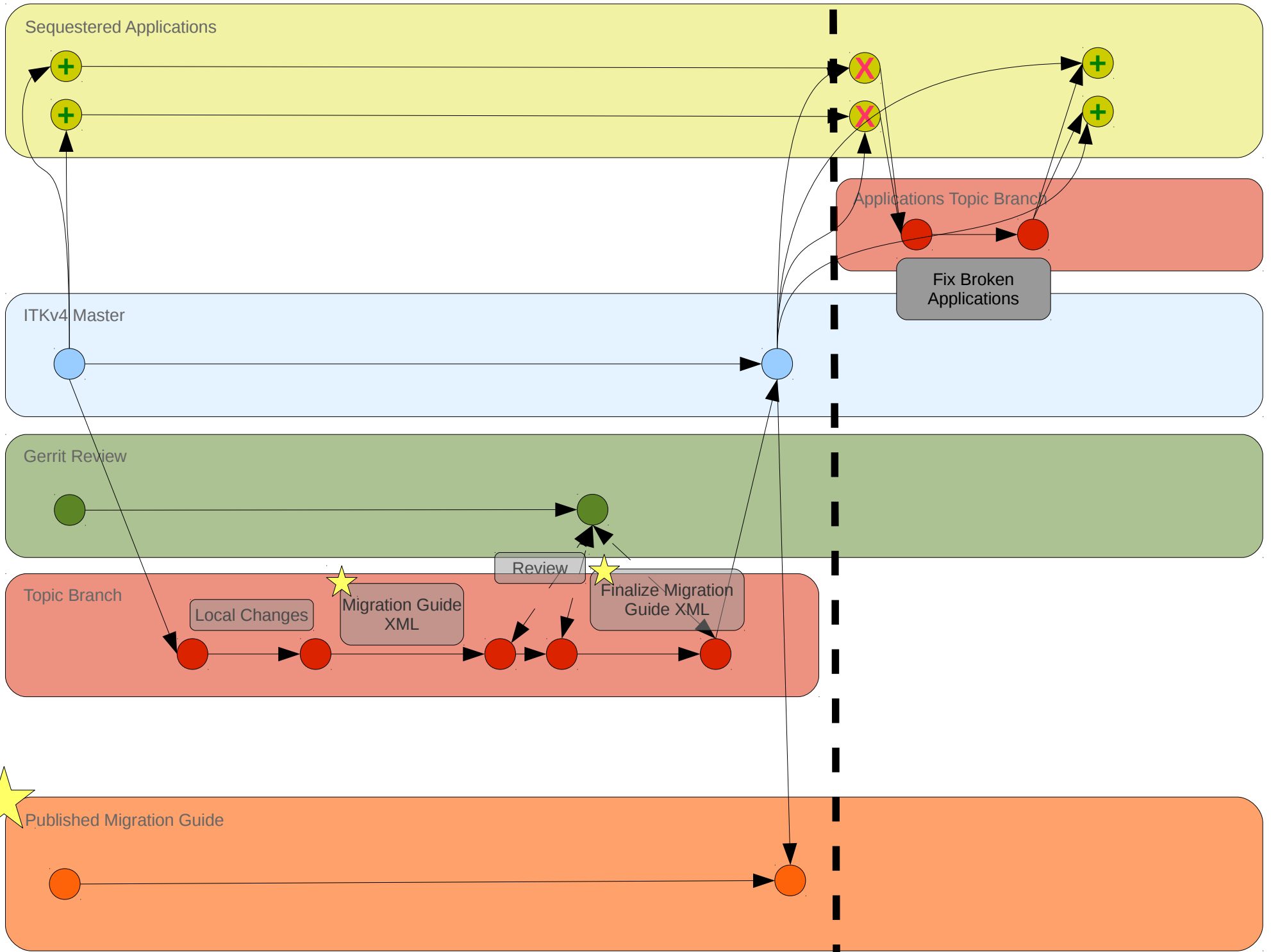
DAY 1

DAY 2



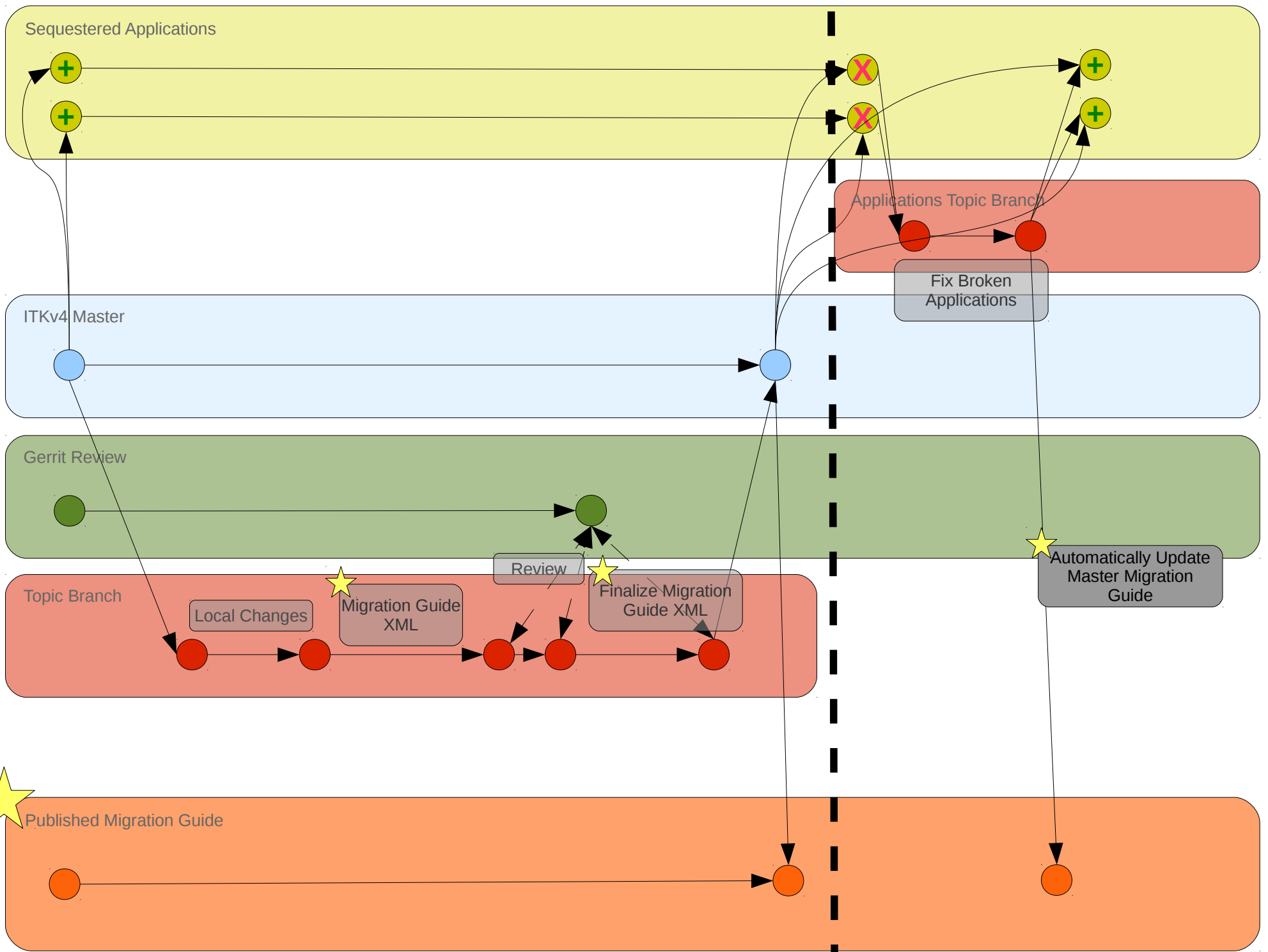
DAY 1

DAY 2



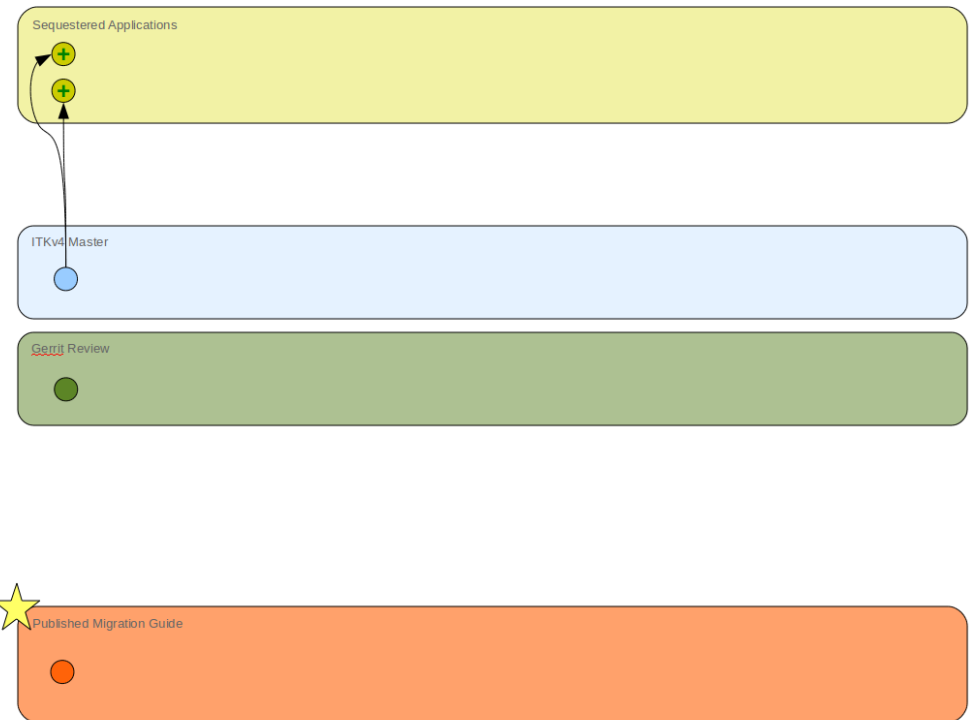
DAY 1

DAY 2



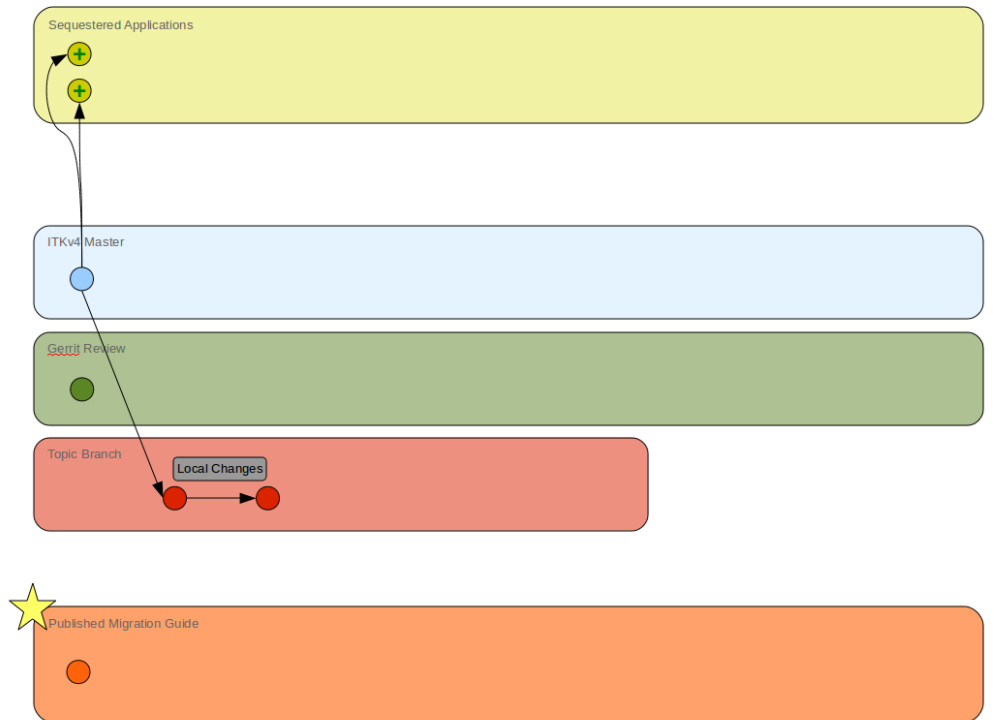
Initial Setup

- The starting state assumes stable states for:
 - Master
 - Gerrit
 - Migration Guide
 - Sequestered Applications
- Same as current workflow with the addition of Migration Guide and Sequestered Applications



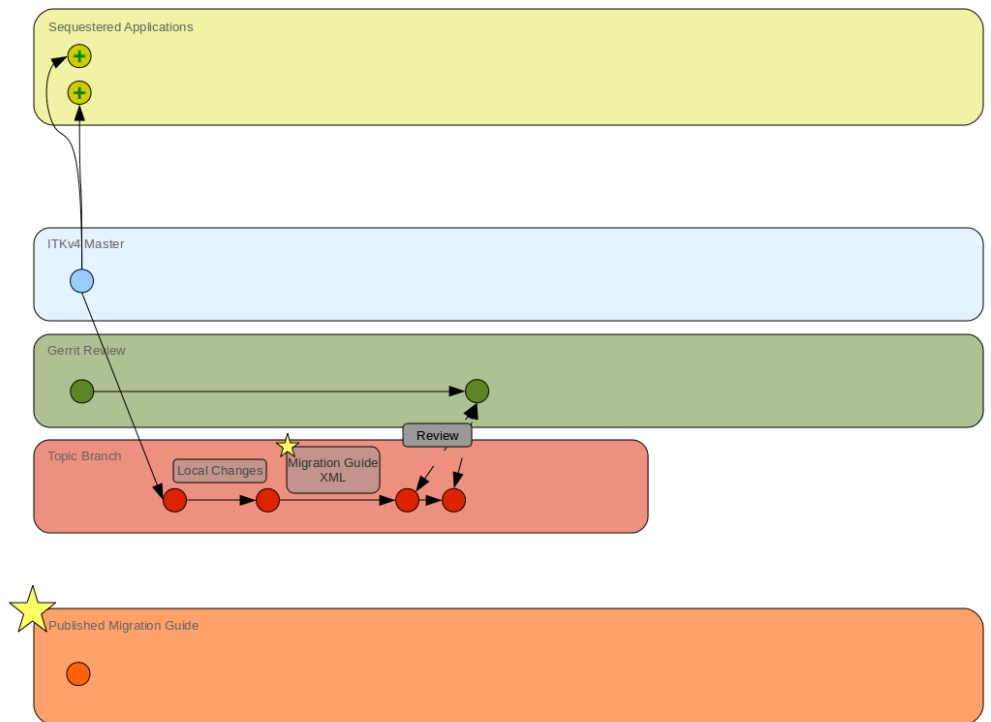
Local Work

- Same as the standard Git workflow
- Edit, commit, edit commit, etc...



Write Migration Guide XML

- Before publishing to Gerrit, create a Migration Guide XML document
- Document lives in **\$ITK_SOURCE_DIR/Migration**
- Document must have a unique name
- The goal is to create as much of this file automatically as possible using information from Git commit messages
- For detailed specs, see the [wiki](#)



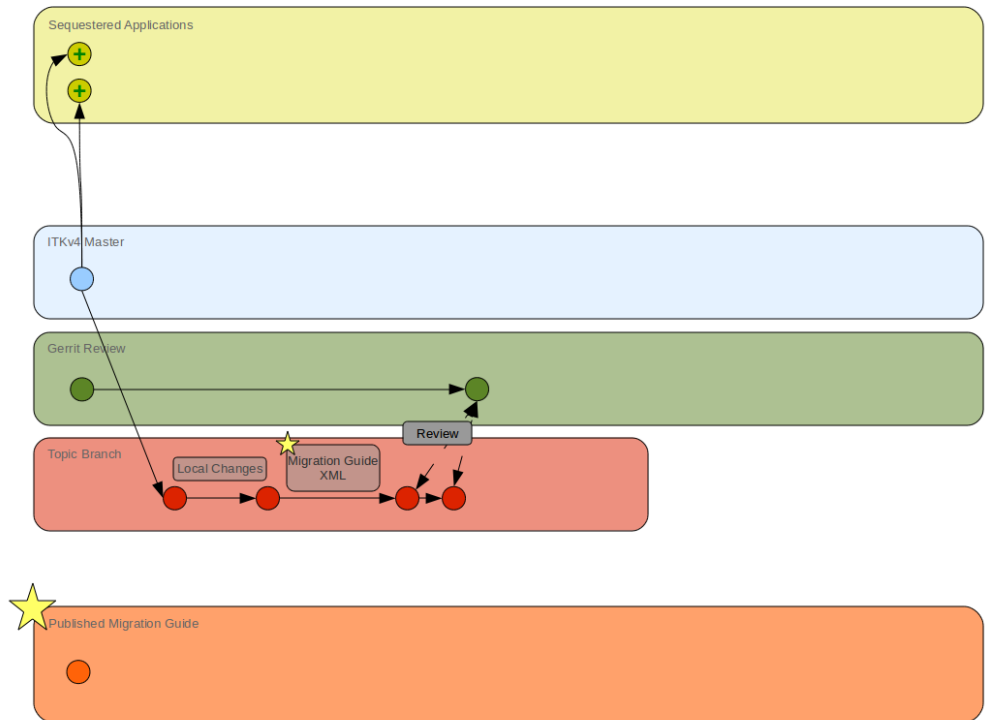
XML Document Format

- **Required Tags:**

- <Title>: Migration Guide page title
- <Description>: English description of change
- <SampleCode>: Simple example of fixing after change
 - _ <Old>: Old code sample
 - _ <New>: Fixed code sample
- <FileList>: List of changed files
- <Gerrid-ChangeId>: ID for Gerrit change

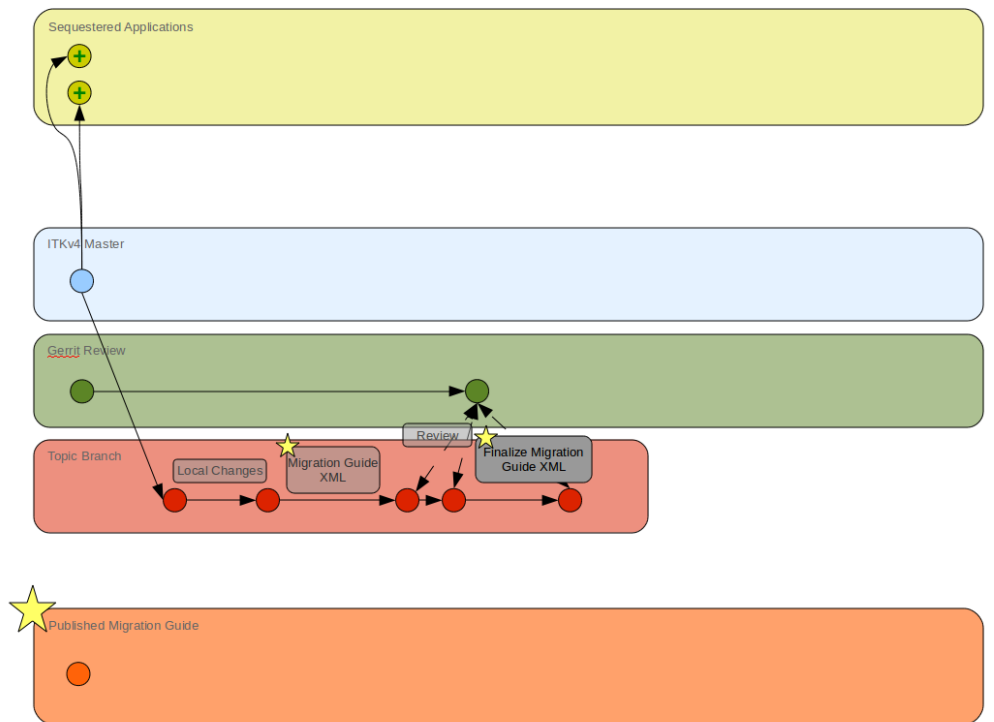
- **Optional Tags:**

- <MigrationGuideTag>: Tag for online categorization (eg: Statistics)
- <MigrationFix-Automatic>: Simple find-and-replace tag
- <MigrationFix-Manual>: Regex to match potential problem spots



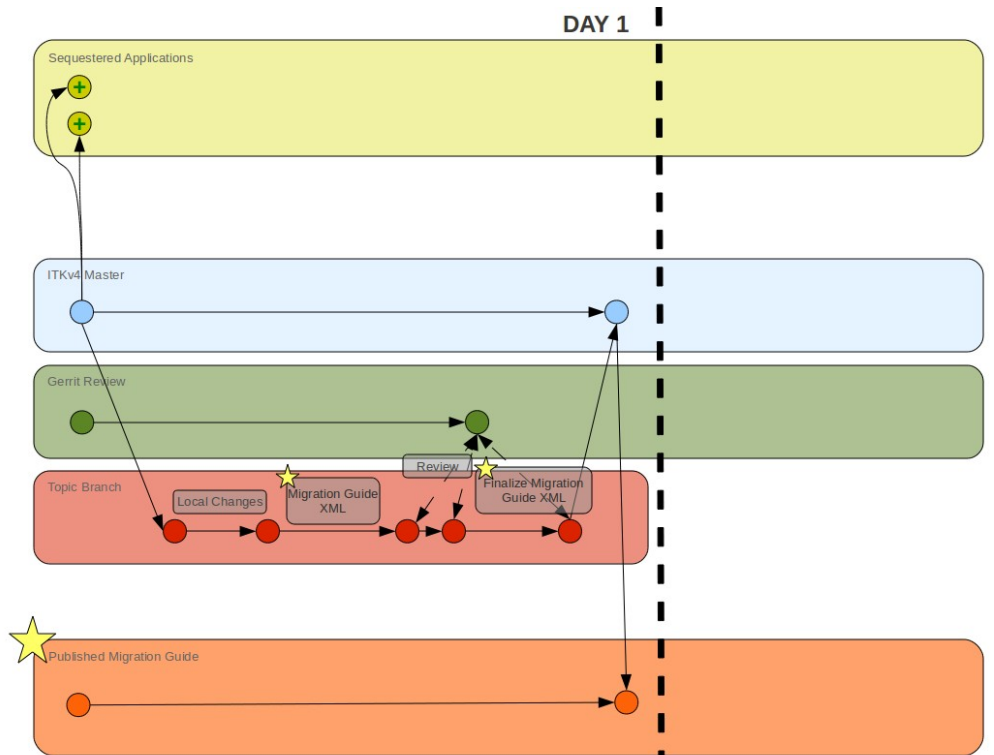
Finalize Migration Guide XML

- Gerrit review process proceeds as usual with any necessary revisions being made
- Reviewers must ensure that, if necessary, a Migration Guide XML document is present and is adequate



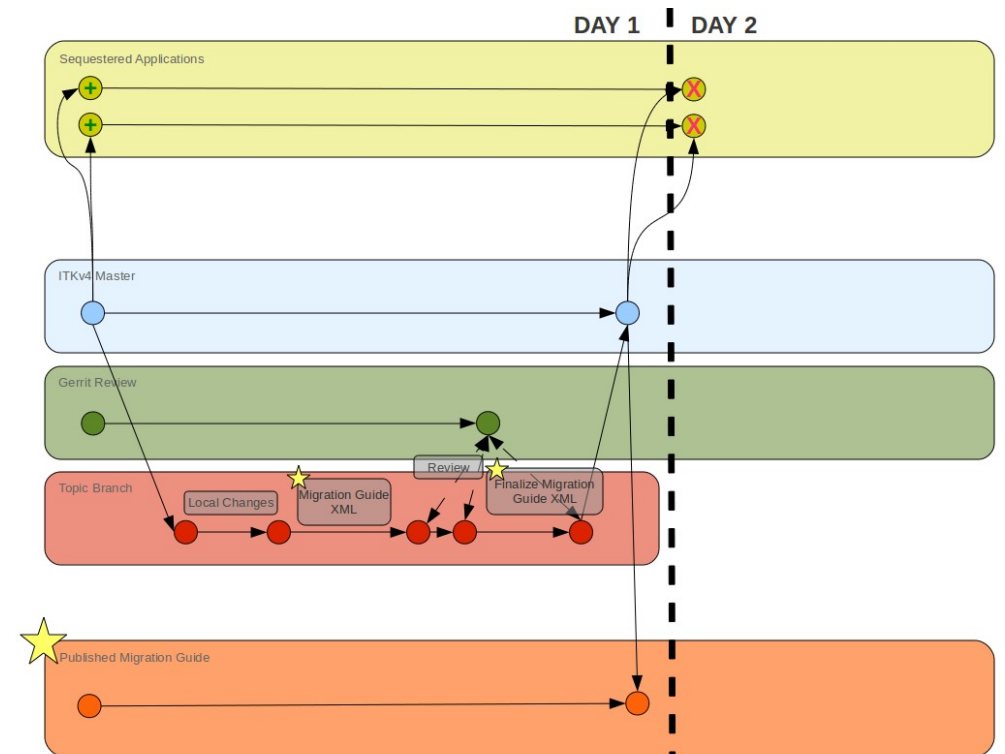
Publish Change

- Publish code to master in the normal way (push to stage and then merge)
- Once in master, the new Migration Guide XML document will automatically be published to the online Migration Guide



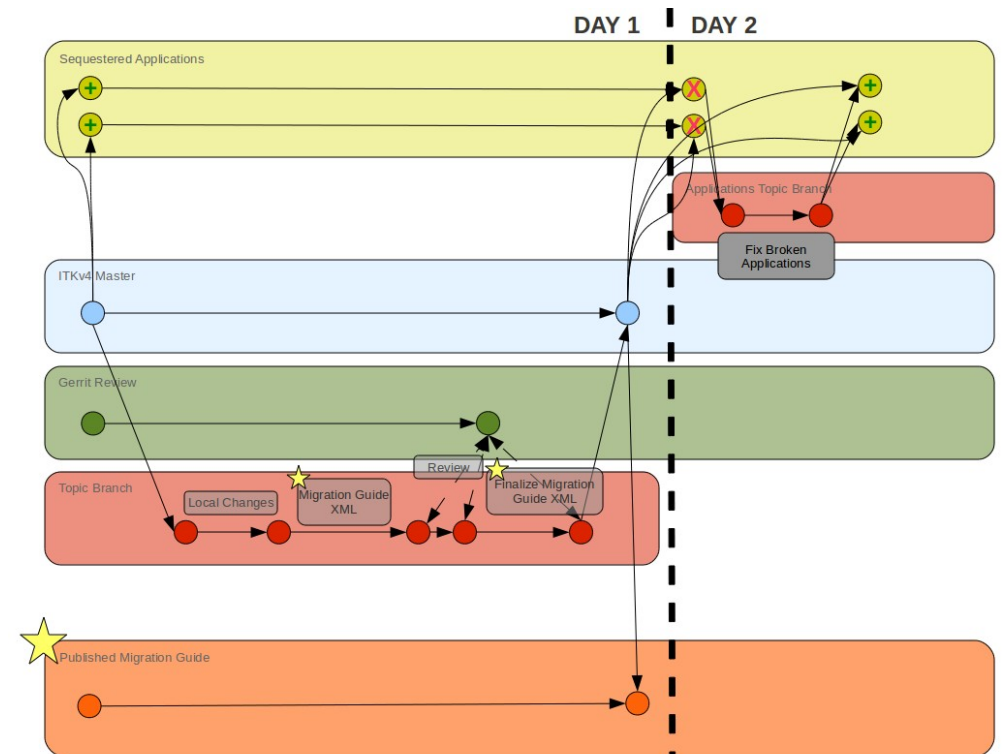
Sequestered Applications Break

- The Sequestered Applications will build using the latest version of ITKv4
- Assumed to build successfully before current API change made
- Applications will break once new API changes committed



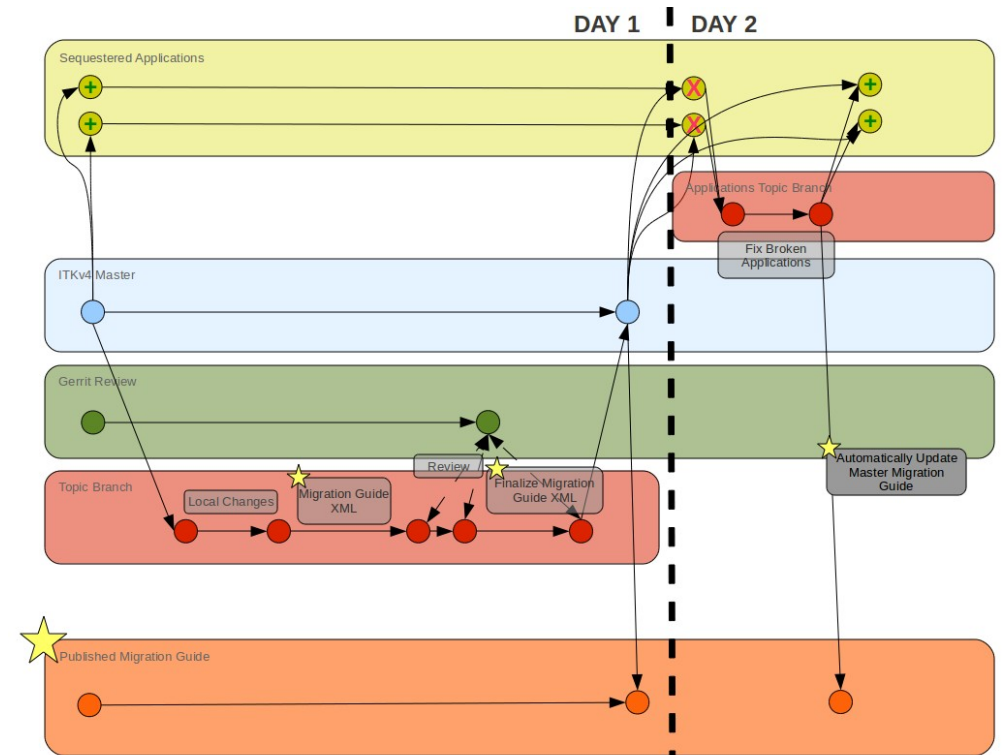
Fix Sequestered Applications

- Check out a copy of each broken application
- Make any necessary changes
- Commit those changes once completely fixed
- Check to make sure dashboards show up green



Augment Migration Guide

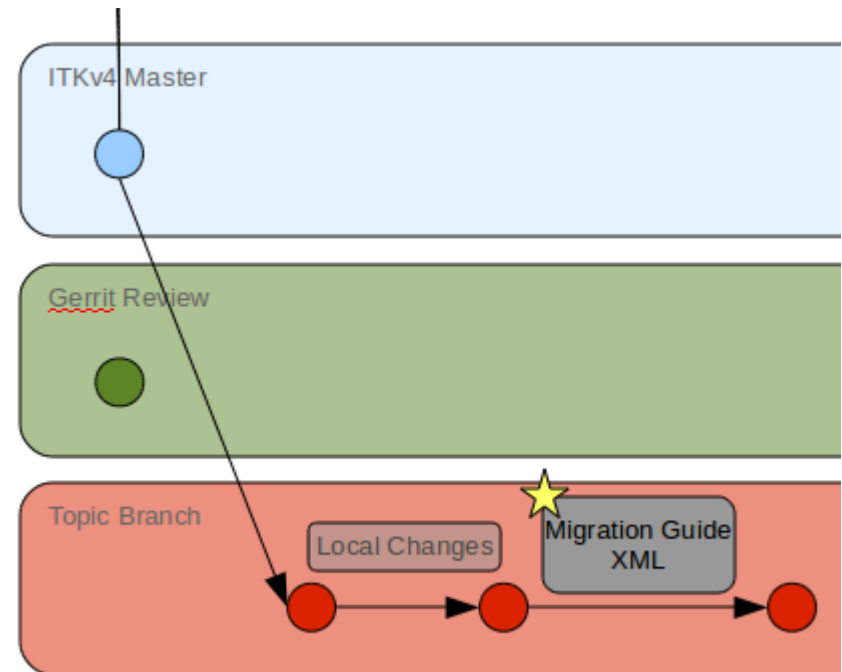
- When changes are committed to the Sequestered Applications, developer provides ID for associated Migration Guide page
- Change information is automatically added/linked to published Migration Guide page



Automatic Components

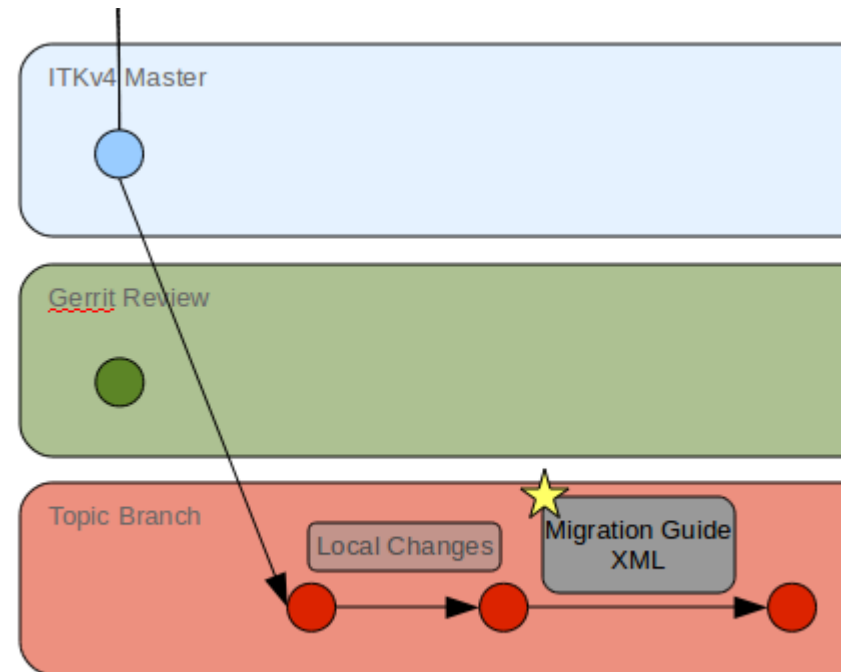
XML Document Generator

- Use a Git hook when pushing to Gerrit to:
 - Harvest information from Git commit for <Description> tag
 - Automatically populate <FileList> tag from Git commit
 - Automatically grab <Gerrit-ChangeId> tag
 - Dump diff into <SampleCode> tags
- Present developer with mockup XML document to edit before pushing
- Will need to edit <Description> and <SampleCode> tags



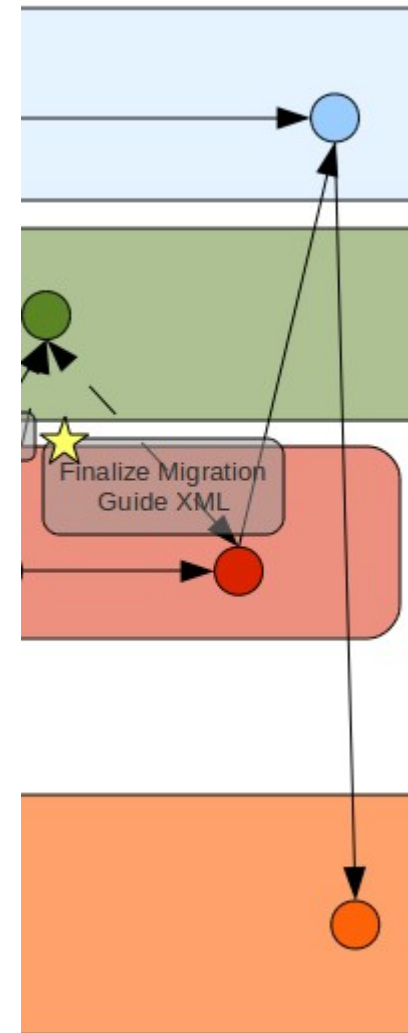
XML Document Generator

- Design Prototype:
 - Standalone script
 - Ask developer for file name
 - Parse Git logs
 - Put all text from commit messages in <Description>
 - Put all changes from test or example files in <SampleCode>
 - Put all changed file names in <FileList>
 - Put all associated Gerrit IDs in <Gerrit-ChangeId>
- Developer will have to manually edit and commit file before pushing to Gerrit



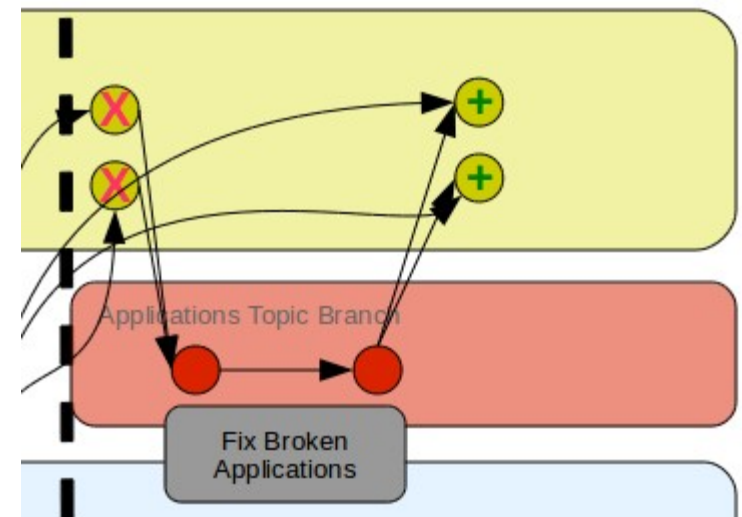
Migration Page Publisher

- Runs after XML document has been committed to Master
- Tool will parse XML document for:
 - <Description> tag
 - <SampleCode> tag
 - <FileList> tag
 - <Gerrit-Changeld> tag
 - Any <MigrationGuideTag> tags
- Tool will create a new page on Migration Guide (phpMyFAQ) from the parsed information
- Julien Jomier and Charles Marion



Migration Tool

- Aimed at users
- Will gather migration rules from:
 - <MigrationFix-Automatic> tags
 - <MigrationFix-Manual> tags
- Will parse entire user project to look for problem locations
- Automatic fix rules will be straightforward find and replace fixes
- Manual fix rules will be all other cases
- For manual fixes, tool will flag all lines of code that match a given regular expression
- Manual fixes will over-warn users, but will hopefully find most problem spots



Migration Page Linker

- Once Sequestered Applications have been manually fixed, the patches will be linked to the Migration Guide page for the related API change
- Use a Git hook in the Sequestered Application repositories to request a Migration Guide page ID from developer
- Linker will automatically add to the bottom of the appropriate Migration Guide page (possibly as a comment)

