

- [Base Ticket](#)
- [Branch and Pull Request](#)
- [Integration Branch](#)
- [Confluence](#)
- [Jenkins – Copy](#)
- [Jenkins – From scratch](#)
- [Finishing](#)

A ticket to track down basic release tasks and to associate it with the branch and pull-request for modifying the ACS release version. In the ticket there will be three main tasks:

- Branch creation and pull-request to update the ACS release version
- Integration branch creation (integration/COMMON-YYYYMMM)
- Confluence page for the release notes
- Jenkins (or build system in use) jobs for ACS

A template for the ticket could be as follow:

Subject: Create COMMON-YYYYMMM core tasks

Description:

```
This ticket is to track the creation of repeatable tasks to be done for COMMON-YYYYMMM release:
* Branch + Pull request for change of tag version of ACS (.bash_profile.acs and ACS_VERSION files)
* Create confluence page with release notes and other general information for the release
* Create Jenkins tasks to generate ExtProds, build, test and release new versions
```

Do the following list of actions:

- Using JIRA-Bitbucket functionality, create the branch associated with the ticket
- Go to your repository and checkout the branch you just created
- Edit the following two files by changing the release version to the new one
 - ACS/ACS_VERSION
 - ACS/LGPL/acsBUILD/config/.acs/.bash_profile.acs
 - In this file look for the definition of 'ALMASW_RELEASE' environment variable
- Make your commit including the ticket number in the description
- Push your changes to Bitbucket

Do the following actions:

- Using Bitbucket functionality, create a branch called 'integration/COMMON-YYYYMMM' using the master as starting point

Do the following actions:

- Go to the ICTACS space and then to the [Release Notes](#) page
- Create a new page, which can be based on an older one (A template will be created as soon as we have permissions) with the name of the release (COMMON-YYYYMMM)
- Edit the Release History and Use in ALMA including any special characteristic of the branch (Java 11, Python 3, etc.) or any big change or concern with the release to be aware of
- Update the filter for the list of tickets
- Fill the initial state of the release notes and update them as new commits are being merged in. The recommended commands to use are:

```
git status
...

git log --no-merges --oneline master..integration/COMMON-YYYYMMM -- ACS
...

git log --no-merges --oneline master..integration/COMMON-YYYYMMM -- ARCHIVE
...

git log --no-merges --oneline master..integration/COMMON-YYYYMMM -- ICD
...
```

- Confirm the system requirements are still valid and update accordingly

Do the following actions:

- Create a view with the short name of the release (YYYYMMM)

- Copy jobs from previous releases
 - Update YYYYMMM to the actual one where required
 - Verify that labels in use for docker containers are the correct ones (CentOS version, correct Java, etc.)

TODO

After finishing the tasks summarized in here, we have our first implemented ticket. At this point we're ready to follow with all the other [Implementation](#).