

How to Contribute

Code Contribution Process

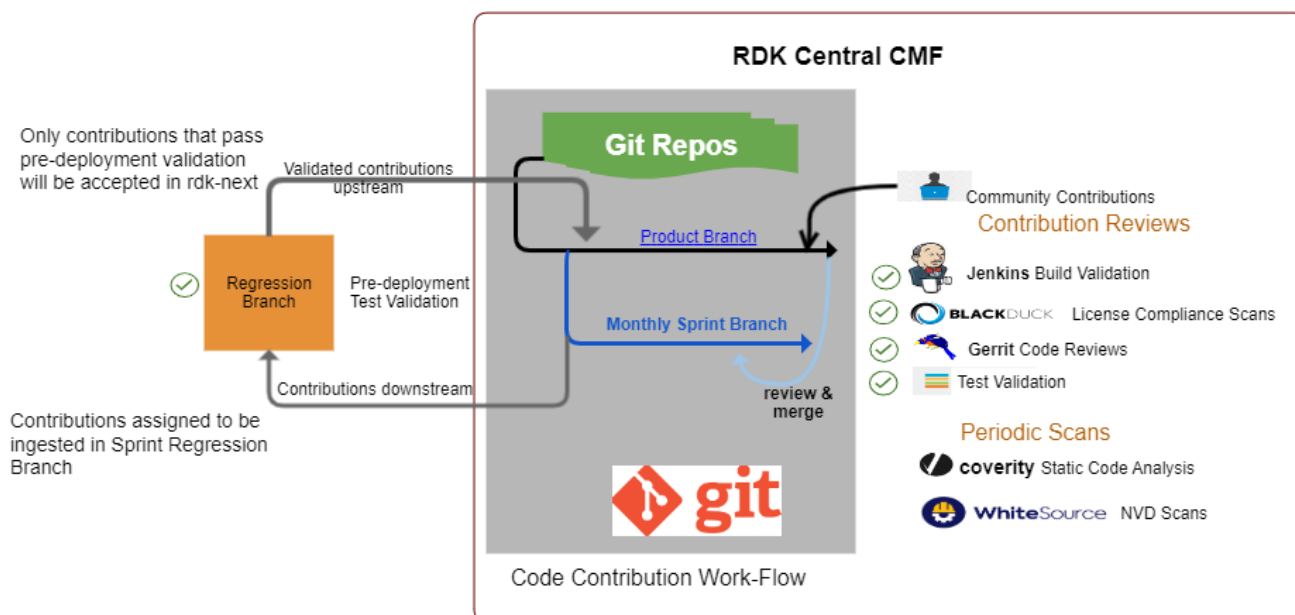
- [Code Contribution Workflow](#)
- [Code Contribution Workflow Diagram](#)
- [Product Branch](#)
- [Monthly Sprint Branch](#)
- [Regression Branch](#)
- [Code Submission Process](#)

Code Contribution Workflow

- Deployment ready [Product Branch](#) has been created for RDK components that the community will push changes to review & It is with higher standards of test qualification
- Monthly Sprint Branch (rdk-dev-yymm) is a new CMF integration branch, created monthly and baseline off [Product Branch](#) Branch. This branch will be hosted per repository in conjunction with Product branch with the goal of incorporating community changes at the earliest juncture.
- Community changes, once approved, will be cherry-picked to Monthly Sprint branch (rdk-dev-yymm) and will thus be available prior to the completion of down-streaming/ round-trip process.
- Approved contributions will be down-streamed to **Regression Branch** for pre-deployment validation using their test process
 - Defects will be planned in monthly sprints
 - Features will be presented for Architecture Review to be scheduled to an upcoming sprint. Sprint timelines to be published to contributor.
 - Contributions pending validation will be available in monthly development iteration branches
- Down streamed Community changes, successfully merged to Regression branch, after pre-deployment test validation, the code changes will be cherry-picked to [Product Branch](#).

Code Contribution Workflow Diagram

1. User will do code contribution to rdk-next branch. This will undergo:
 - a. Code reviews
 - b. Build verification
 - c. License compliance scan
 - d. Test validation
2. Once successful, the change will get cherry-picked to **Monthly Sprint Branch** (rdk-dev-yymm)
3. This code is then down-streamed to **Regression Branch** where pre-deployment test validation are done
4. Once Comcast accepts, the change-set is cherry picked to [Product Branch](#).
5. Thus the change gets merged to [Product Branch](#)



Component owners/reviewers/approvers, defined as specific groups in Gerrit, will be added to the review by default. You may request additional feedback by specifically adding reviewers via the Gerrit web GUI.

Product Branch

Product branch is a deployment ready branch is created for RDK components that the community will push changes to review.

Refer to [Product Branch](https://code.rdkcentral.com) for the Components hosted in CMF Gerrit (<https://code.rdkcentral.com>)

Refer to [RDK Central GitHub Components & its Branches](https://github.com/rdkcentral/) hosted in <https://github.com/rdkcentral/>

Monthly Sprint Branch

Monthly Sprint Branch (rdk-dev-yymm) is a new CMF integration branch, created monthly and baseline off [Product Branch](#). This branch will be hosted per repository in conjunction with Product branch with the goal of incorporating community changes at the earliest juncture.

Once community changes is approved, will be cherry-picked to Monthly Sprint branch (rdk-dev-yymm) and will thus be available prior to the completion of down-streaming to Regression Branch / round-trip process.

Regression Branch

Regression branch is the branch used for validation of the contributions. Approved contributions will be down-streamed to **Regression Branch** for pre-deployment validation using their test process.

- Defects will be planned in monthly sprints
- Features will be presented for Architecture Review to be scheduled to an upcoming sprint. Sprint timelines to be published to contributor.
- Contributions pending validation will be available in monthly development iteration branches

Down streamed Community changes, successfully merged to Regression branch, after pre-deployment test validation, the code changes will be cherry-picked to Product Branch.

Code Submission Process

[Code Submission Process - RDK Central Gerrit](#)

[Code Submission Process - RDK Central GitHub](#)

RDK-V Development Workflow

This document describes the general RDK development work-flow and related topics.

Code Management Facility (CMF)

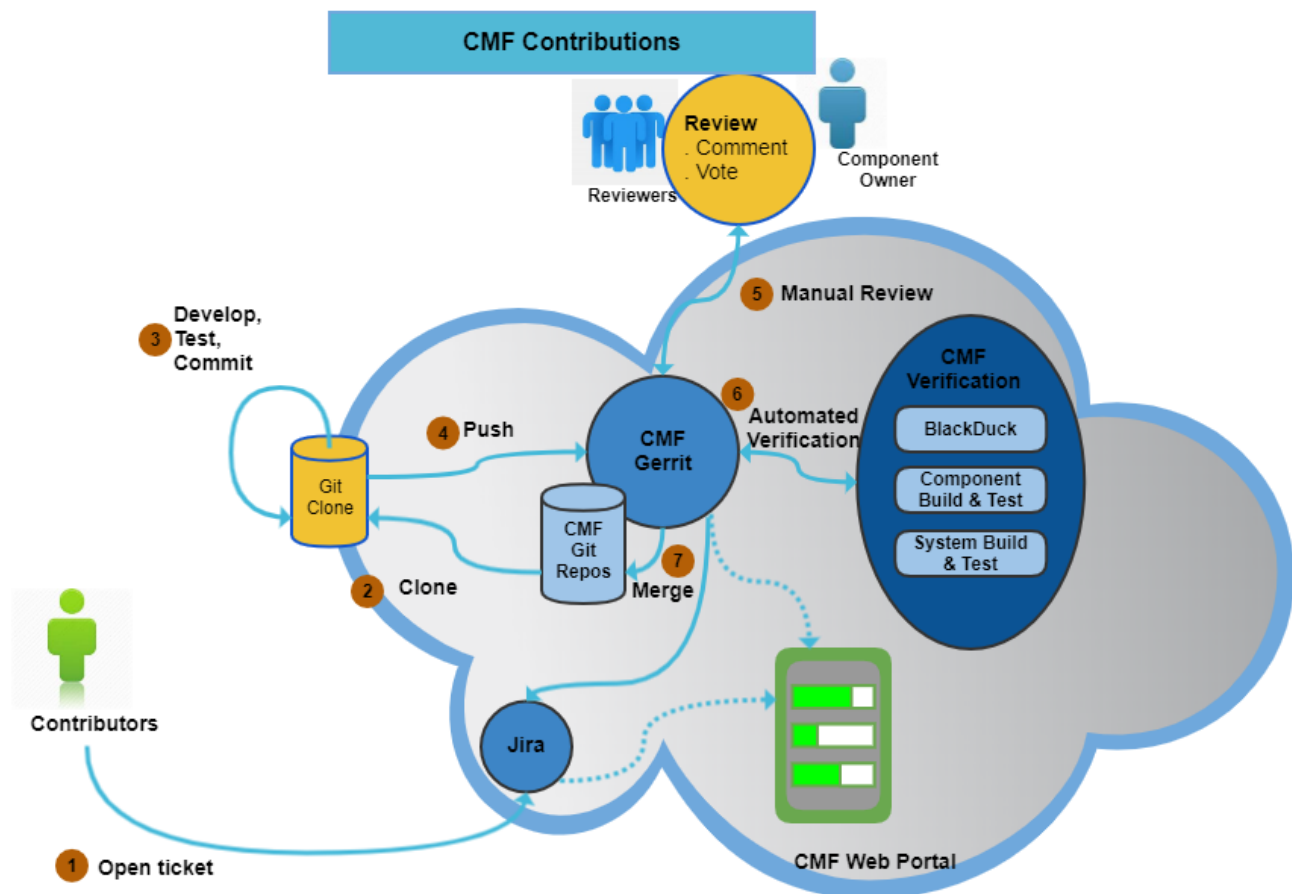
On a periodic basis, RDK code is tested and released to the community as [CMF releases](#). This will be generic RDK code without dependency to any platform. CMF code can be built for [emulator](#) or [raspberry-pi](#) or can be ported to a specific platform ([RDK Porting](#)).

Routine Releases

- Tagged every two weeks
- Built and tested
- Only tagged if Black Duck, Build and Test successful
- Changes pulled from Comcast every two weeks

CMF Contributions

While working with CMF stack, one might find ways to enhance RDK code by adding new features or bug fixes as RDK contribution. The general CMF contribution workflow is as follows:



Detailed information on contributing code changes to RDK can be found here: [Code Management Documentation](#)

Getting Support

Support tickets can be raised to get request support from RDK Community Support team. This can be for the bugs you faced, doubts you have or any code contributions which you think might enhance RDK.

RDK Support ticket can be raised here: <https://jira.rdkcentral.com/jira>

Categories of support tickets are:

| Issue type | Name | Description |
|----------------------------------|-------------------------|---|
| General Support Issues | RDK_JIRA_<Company_Name> | Raise a JIRA ticket in your Company Specific Project. This project is accessible only to Preferred member companies. |
| Code Management Facility Support | RDKCMF | Code contributions to CMF as part of bug fixing or feature enhancement comes under this category. Updating of code from RDK to CMF is also tracked in these. |
| RDK Support | RDK | Anybody with RDK Support credentials can create this category of tickets. |
| Raspberry-pi (H DP) Support | RPI | Rapsberry-pi support tickets (only for Preferred members) |
| Default | RDK_JIRA_TRIAGE | By default, support type will be RDK_JIRA_TRIAGE. All normal users without RDK support credentials can create these JIRA tickets via mailing to support@rdkcentral.com. This will be filtered to appropriate category by triage team. |