

CA Agile Central (RALLY) Integration with GitHub & Jenkins

The integration of RALLY with GitHub and Jenkins gives the development team complete control over the codes being committed in the source code repository. It also creates complete traceability for all workitems in RALLY. With complete traceability for each workitem in the ecosystem, it is easier for enterprises to fulfil compliance requirements.

Integration overview

In an Application Lifecycle Management (ALM) ecosystem, the choice of systems and the collaboration between the cross-functional teams play a great role. While the choice of systems impacts the productivity of a team, the cross-functional collaboration brings in collective wisdom to take better decisions, faster.

Best-of-breed systems such as RALLY, Jenkins, and GitHub bring rich functionalities to the ecosystem. When GitHub is integrated with RALLY and Jenkins, all stakeholders have real-time visibility into the commits made by the development team. It is also easier to enforce authentic commits against each work item and access the changes/edits made to the commit files from RALLY itself.

How RALLY - GitHub - Jenkins integration is beneficial for an enterprise

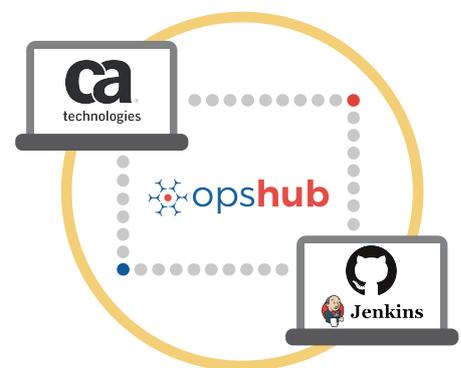
- Track commit volume, track commit trends and edits/changes to commit files in real time
- Enforce authentic commits to make sure each commit is happening against a scheduled and open workitem
- Eliminate manual effort to close RALLY workitem by automating the state transition on GitHub commit

How OpsHub Integration Manager integrates RALLY, GitHub, and Jenkins

OpsHub Integration Manager integrates RALLY with GitHub and Jenkins. It ensures that all data is available to each user, in that user's preferred system, with full context, in real-time. All the details related to a commit made against a work-item in RALLY can be tracked from RALLY itself. For example, for each commit that development team makes in GitHub, GitHub synchronizes a 'commit entity' linked to the specific requirement id back to RALLY. Each 'commit entity' includes information such as 'who did the commit?', 'when was the commit done?', and 'which part of the code was committed?'. The integration of Jenkins also ensures elimination of developer's effort to close RALLY workitem by automating the state transition on GitHub commit.

With RALLY + GitHub + Jenkins integration, enterprises can:

- Make better and faster decisions
- Enforce checkpoints to ensure quality check-ins
- Ensure complete traceability of a 'requirement'
- Ensure quality delivery in stipulated time
- Leverage the best of functionality and collaboration in the delivery ecosystem



Entities that can be synchronized between RALLY and GitHub

The popularly synchronized entities between RALLY and GitHub are shown on the left:



Benefits of integration for RALLY and GitHub users

RALLY users	GitHub users
Each commit can be traced back to its respective workitem at any given point in time from GitHub itself	Complete traceability from RALLY to source code in GitHub
Enforced checkpoints ensure that no mandatory steps/checks are missed while making a commit – this leads to high success rate for commits	Visibility into the volume, quality of commits, and commit trends in real-time
	Reduced dependency on manual communication to track the completion of a task

Features of OpsHub Integration Manager



Unidirectional as well as bi-directional synchronization



Full history and audit trail for integrated systems



Complete traceability of work items as well as non-work items



Robust failure management and recovery mechanism

Pre-requisites to run OpsHub Integration Manager

Supported Operating Systems

Windows

- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2 (64 bit)

Linux

- RHEL 5.2 + (64 bit)
- RHEL includes Cent OS and Fedora

Tested on the following versions:

- CentOS release 5.5 (Final)
- CentOS release 5.6 (Final)
- CentOS Linux release 7.1.1503 (Core)
- Fedora 20

Database Prerequisites

The underlying database should be installed to install and run OpsHub Integration Manager. The database user created for OpsHub Integration Manager should have schema level and read write privileges.

- MySQL Server
- MS SQL
- Oracle
- HSQLDB