

TFS Integration with JIRA and GitHub

The integration of TFS with JIRA and GitHub ensures completely traceability of all workitems in the ecosystem. With this integration, the product management team will have real-time visibility into all defects, commit trends, and commit volume.

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 helps the teams get complete context of the business requirements.

Best-of-breed systems such as TFS, JIRA, and GitHub bring rich functionalities to the ecosystem. By integrating TFS, JIRA, and GitHub, the product development team will have real-time visibility into the defects logged by QA team and commits made by the development team. It is also easier for product development team to enforce authentic commits against each work item, and access the changes/edits made to the commits files.

How TFS - JIRA - GitHub 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 JIRA or TFS workitems by automating the state transition on GitHub commit

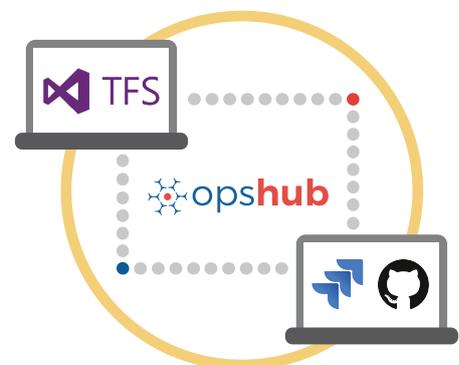
How OpsHub Integration Manager integrates TFS, JIRA, and GitHub

OpsHub Integration Manager integrates TFS, JIRA, and GitHub - all systems with each other bi-directionally. It ensures that all historical and current 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 TFS can be tracked from TFS 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 TFS. 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 Project Management team can also view the development progress and commit details from TFS itself.



With TFS + JIRA + GitHub integration, enterprises can:

- Make better and faster decisions
- Ensure complete traceability of a 'requirement'
- Meet all compliance requirements
- Ensure quality delivery in stipulated time
- Leverage the best of functionality and collaboration in the delivery ecosystem



Entities that can be synchronized between TFS, JIRA & GitHub

The popularly synchronized entities between TFS, JIRA & GitHub are on the left:

Benefits of integration for TFS, JIRA & GitHub users

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

Features of OpsHub Integration Manager



Supports unidirectional as well as bi-directional synchronization between 50+ systems



Maintains complete history and audit trail among integrated systems



Allows traceability between code to requirement, tickets to defects, and many other entities



Provides a robust failure management and recovery mechanism



Can be hosted by OpsHub, installed on-premise, or deployed in a customer cloud

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