How to Sync Sprints on Jira On-premise

This article shows how to synchronize sprints in Exalate for Jira on-premise.

> **Note**: You need to sync sprints with a trigger before syncing issues.

Introduction

Starting from version 4.7.0, you can add sync rules to synchronize sprint data. Exalate considers issues and sprints as independent entities. The scripts below use board IDs to identify what sprints to sync. For example, if you enter a Board ID `1`, you sync all sprints from that board.

> **Note**: Sprint synchronization in Jira on-premise is available only for connections in Script mode.

Source Side

Outgoing Sync

Use the following script to send the sprint data:

```
def boardIds = ["50", "80", "130", "144"] //Boards which sprints will get synced
if(entityType == "sprint" && boardIds.find{it == sprint.originBoardId}){
    replica.name = sprint.name
    replica.goal = sprint.goal
    replica.state = sprint.state
    replica.startDate = sprint.startDate
    replica.endDate = sprint.endDate
    replica.originBoardId = sprint.originBoardId
}
if(entityType == "issue"){
   //Executed when syncing an issue to a remote side
   replica.summary = issue.summary
   replica.description = issue.description
   replica.project = issue.project
   replica.type = issue.type
 //....
 //other script rules to sync issues
 //sprint....
   replica.customFields."Sprint" = issue.customFields."Sprint"
 }
```

Destination Side

Incoming Sync

Use this script to sync the received sprint data on your side:

```
//entityType represent the type of the remote side entity
if(entityType == "sprint"){
   //Executed when receiving a sprint sync from the remote side
   def sprintMap = ["162":"50", "197": "80", "214": "130", "225": "144"] //[remoteBoardId: localBoardId]

   sprint.name = replica.name
   sprint.goal = replica.goal
   sprint.state = replica.state?.toUpperCase()
   sprint.startDate = replica.startDate
   sprint.endDate = replica.endDate
   def localBoardId = sprintMap[replica.originBoardId]
   if(localBoardId == null){
      throw new com.exalate.api.exception.IssueTrackerException("No board mapping for remote board id "+replica.ori
ginBoardId)
   }
   sprint.originBoardId = localBoardId //Set the board ID where the sprint will be created
}
if(entityType == "issue"){
   //Executed when receiving an issue sync from the remote side
   issue.summary = replica.summary
   if(replica.projectKey == "SCRUM"){
      issue.projectKey = "SCRUM2"
   }
   if(replica.projectKey == "SCRUM2"){
      issue.projectKey = "SCRUM"
   }

   issue.typeName = replica.typeName

   def sprintV = replica.customFields.Sprint.value?.id.collect{ remoteSprintId ->
      return nodeHelper.getLocalIssueKeyFromRemoteId(remoteSprintId, "sprint")?.id?.toString()
   }.findAll{it != null}

   issue.customFields."Sprint".value = sprintV

}
```

```
if(entityType == "sprint" && sprint.originBoardId == "12"){
  //Only sync sprints on board 12
    if(sprint.id == "87") return //avoid syncing sprint with id 87
    replica.name = sprint.name
 replica.goal = sprint.goal
 replica.state = sprint.state
 replica.startDate = sprint.startDate
 replica.endDate = sprint.endDate
    replica.originBoardId = sprint.originBoardId
  }
//The rest of your outgoing script
```

Fields of the sprint entity

field name	type

field name	type
sprint.name	String
sprint.state	String
sprint.originBoardId	String
sprint.startDate	Date
sprint.endDate	Date
sprint.completeDate	Date
sprint.goal	String

How to Start Sprint Sync?

To start the synchronization of a sprint you need to create a trigger and select the **sprint** entity type. Sprints are synced after they are created or updated.

Create Trigger

Specify a JQL query to synchronize issues automatically. If the issue fits the query, it will be synced on the create/update issue event. **Find more details**.

Trigger will apply to selected entity type*

sprint

If sprint gets updated or created

Then sync with connection*

JCloud_to_JServer

Notes

Sync sprints from Boards with Board ID's specified in outgoing and incoming sync

Active?

Have more questions? Ask the community

Join our Community 🔗

Visit our Service Desk 🔗
Find a Partner 🔗