How to Sync Statuses in GitHub

In this article, we describe how to sync status between instances GitHub and Jira.

You need to configure the mapping between statuses on both sides.

GitHub to Jira sync

GitHub Outgoing sync rules

Send local status to the remote side:

```
replica.status = issue.status
```

Jira Incoming sync rules

Set the local status based on the received status value from the remote side:

```
def statusMap = [

    // "remote status name": "local status name"
  "open" : "To Do",
      "closed" : "Done"    ]
def remoteStatusName = replica.status.name
issue.setStatus(statusMap[remoteStatusName] ?: remoteStatusName)
```

Jira to GitHub sync

Jira Outgoing sync rules

Send local status to the remote side:

```
replica.status = issue.status
```

GitHub Incoming sync rules

Set the local status based on the received status value from the remote side:

```
def statusMap = [

    // "remote status name": "local status name"
    "To Do" : "open" ,
      "Done" : "closed"    ]
def remoteStatusName = replica.status.name
issue.setStatus(statusMap[remoteStatusName] ?: remoteStatusName)
```

Alternatively, you can set the default local status using the getStatus node helper method.

```
issue.setStatus("status name")
```

Have more questions? Ask the community

Product

About Us ⬈

Release History ⬈

Glossary ⬈

GitHub to Jira sync

API Reference ⬈

Jira to GitHub sync

Security ⬈

Pricing and Licensing ⬈

Resources

Subscribe for a weekly Exalate hack ⬈

Academy ⬈

Blog ⬈

YouTube Channel ⬈

Ebooks ⬈

Still need help?

Join our Community ⬈

Visit our Service Desk ⬈

Find a Partner ⬈