

# How to Stop Sync on HP ALM/QC

Last Modified on 01/15/2026 5:27 pm EST

**EOL NOTE:** We are moving Exalate for HP QC/ALM into basic maintenance mode. This transition period lasts until June 2024, after which support is halted completely. For more information, please see <https://exalate.com/blog/end-of-support-hp/>.

There are certain cases when you need to stop a defect synchronization at some point. Since the Exalate app for HP ALM/QC is installed on a separate server it does not have a built-in feature that helps to stop sync directly on a defect. A typical use case is when you, for example, need to stop one specific project synchronization. This article shows how to stop a defect synchronization. Please follow the steps below to stop the synchronization:

## 1. Stop the Exalate server for HP ALM/QC

In order to do that you should execute the query on the physical server where the Exalate server is installed.

**Note :** You need to get root access to the host and execute the query as a root user.

```
/etc/init.d/hpqcnode stop
```

## 2. Uninstall the Exalate for HP QC from the host by running

```
yum remove exalate-hpqcnode-0.9.20.rpm  
//replace 0.9.20 with your exalate-hpqcnode version
```

## 3. Ensure that Exalate does not try to sync anything from Jira to HP ALM/QC

In order to do that disable the Connection from JIRA instance side: Jira - Administration - Exalate - Connections - edit Connection - toggle the "Active" box to change the status.

4. To dispose of the Exalate's sync panel for the issues, navigate to [Support Tools](#) and then clean up sync by Connection.

5. Remove the post-function to Unexalate the issue on JIRA Server.

You can find the list of workflows that use Exalate's post functions by executing an SQL query to your Jira Database.

```
SELECT WORKFLOWNAME FROM JIRAWORKFLOWS WHERE DESCRIPTOR LIKE '%com.exalate%'
```

To find which particular transition is using Exalate's post-functions conditions and validators:

1. Export workflows into **xml**
2. Open the file in a text editor
3. Find all occurrences of **com.exalate**

4. Find the closest enclosing tag **<action>** and you find an id and a name of the transition which uses Exalate post-function is and

In this example the transition id=711 and transition name="Escalate"

```
<action id="711" name="Escalate">
  <meta name="jira.description"></meta>
  <results>
    <unconditional-result old-status="Not Done" status="Done" step="1">
      <post-functions>
        ...
        <function type="class">
          <arg name="full.module.key">com.exalate.jiranodeescalate-now</arg>
        <arg name="RelationID">2</arg>
        <arg name="class.name">com.exalate.trigger.EscalateNow</arg>
      </post-functions>
    </unconditional-result>
  </results>
</action>
```

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 [?](#)