



cockpit
IT Service Manager

Tickets - Synchronizing Cockpit ITSM with Jira Software

FAQ document

Table of contents

Introduction.....	3
I.Objective.....	3
II.Principles.....	3
Configuration.....	4
I.Types for tickets.....	4
II.Statuses.....	4
A.Cockpit ITSM.....	4
B.Jira Software.....	5
III.Priorities.....	5
A.Cockpit ITSM.....	6
B.Jira Software.....	6
IV.User.....	6
A.User management.....	6
B.API keys.....	7
V.Teams.....	7
A.Cockpit ITSM.....	7
B.Jira Software.....	7
Source fichier XML.....	8
I.<connector> tag.....	8
II.Synchronization process.....	10
A.Ticket type.....	10
B.Mapping the IDs of tickets.....	10
C.Creating a ticket.....	11
D.Updating a ticket.....	13
E.Closing of a ticket.....	15

Introduction

I. Objective

This goal of this document is to present a functional example of a synchronization between a Cockpit ITSM instance and a Jira Software instance.

II. Principles

- Please see the document “FAQ - Synchronization User Guide” for more information on the functions and features of the Cockpit ITSM menu.
- While reading this document, plus refer simultaneously to “FAQ - Jira XML Source Code Synchronization”, the source document that is explained in the present document.
- A firm understanding of Jira Software and Cockpit IT Service Manager is required.

Configuration

Objectives: Before writing the XML code for synchronizing a Cockpit ITSM portal with a Jira Software instance, the following programming elements must be taken into consideration.

I. Types for tickets

Cockpit ITSM can synchronize 3 types of tickets corresponding to the 3 types of ITIL processes (Request, Change, Incident).

Jira Software has several types of tickets (Task, Bug, etc.) and it is possible to create customized ticket types.

The different ticket types can be synchronized with the Cockpit ITSM, to identify the types and terms to be used in the XML code:

- Go to the Jira menu “Jira settings > Issues > Issue types”.
- To synchronize a ticket type, note the value of its “Name” field that will be used in the “XML” code (do not use a translation, click on “Edit” to get the exact term.

Note: With Jira, ticket types belong to screen schemes and a screen scheme is used by a project.
So when a ticket is created, the ticket types available depend on the ticket project.

The table below suggests a correspondence between the types of tickets that can be synchronized:

Cockpit ITSM	Jira Software
REQUEST	TASK
INCIDENT	BUG
CHANGE	Other (Story, etc.) or custom type

II. Statuses

Principle:

The statuses of the tickets will be mapped between Cockpit ITSM and Jira Software.

In the example below, we map the Jira status “Open” with the Cockpit status “New”:

```
<valueMap externalValue="Open" cockpitValue="New"/>
```

A. Cockpit ITSM

Menu: Tickets > Configuration > Incidents / Requests / Changes > Statuses

Principle:

Use the value in “Reference” field of the statuses to designate them in the XML code.

B. Jira Software

1. Statuses

Menu: Jira settings > Issues > Statuses

Principles:

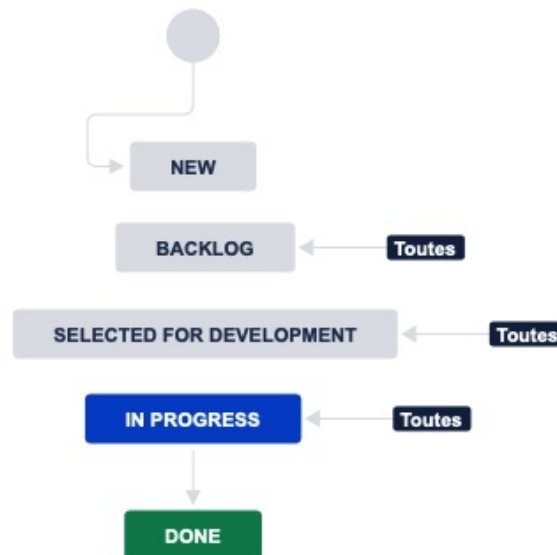
- Click on “Edit” action of the status you want to use in the XML file, use the value in the “Name” field.
- The Jira statuses correspond to the Cockpit statuses.

2. Workflows

Menu: Jira settings > Issues > Workflow schemes

Principles:

- The various statuses of tickets are determined to the workflow of the ticket.
- In a workflow some statuses can be assigned directly to tickets, other statuses can only be assigned by one or more transitions, i.e. according to the current status of the ticket. The aim is to impose a life cycle on tickets.
- In the example below, the status “In progress” can be assigned from any status but the status “Done” can only be assigned after the status “In progress”.



- This is an important element to take into account when setting up the XML file, in this example if a Cockpit ticket is taken into account and then closed (“Done” status) without going through an “In progress” status, the status of the ticket in Jira cannot be synchronized with that of Cockpit.

III. Priorities

Principles:

The priorities of the tickets will be mapped between Cockpit ITSM and Jira Software.

Example below, we have mapped the Jira priority “Highest” with the Cockpit priority “High”:

```
<valueMap externalValue="Highest" cockpitValue="High"/>
```

A. Cockpit ITSM

Menu: Tickets > Configuration > Incidents / Requests / Changes > Priorities

Principles:

- Use the value in “Reference” field of the priorities to designate them in the XML code.
- Check if the “Priority” field is selected in the ticket template used to create tickets during synchronization.
- Check the priorities that can be used in the organization’s catalog.

B. Jira Software

Menu: Jira settings > Issues > Issue attributes > Statuses

Principle:

Click on “Edit” action of the priority you want to use in the XML file, get the value in the “Name” field.

IV. User

Principle: A Jira user will be required to connect to the Jira Software portal as well as for actions on tickets.

It is strongly recommended to dedicate a user for synchronization actions.

A. User management

Menu: Go to “Settings > User management” and create an new user.

Configuration:

- In the tag <connector /> you will have to indicate the user’s email address in the “username” parameter.
- The user’s name (terms used in the “reporter” field of a ticket for example) will be used in the filters of the tags <create />, <update /> and <close />.
- Roles: the user must have sufficient rights to handle the tickets.
- Time Zone: In order to avoid synchronization problems, the Cockpit ITSM end user and the Jira user must be set with the same TimeZone that is declared in the “defaultTimeZone” parameter.

It is recommended to configure these 2 users in TimeZone UTC.

In the personal settings of the user, indicate “GMT Offset” in the region field and “(GMT+00:00) GMT” in the TimeZone field. In the “defaultTimeZone” parameter of the connector tag, you must indicate “UTC”.

Note: If you use a TimeZone other than UTC, the principle will be the same, indicate “Region/Time”, “Europe/Paris” for example.

B. API keys

Menu: In the account settings of the Jira user, go to the menu “Account settings > Security > API key > Create and manage API keys”.

Configuration: In the “password” parameter of the <connector /> tag, you must indicate the API token previously created.

V. Teams

Principle: Synchronize the teams that own the Cockpit and Jira tickets.

Note: In Cockpit ITSM a ticket necessarily belongs to a team.

A. Cockpit ITSM

Menu: Administration > Teams

Principles: Edit a team and retrieve its name, use the team name in the XML file.

B. Jira Software

Menu: Jira Software > People

Principle:

- Edit a team, take the ID appearing in the Url, example:

<https://xxxxx.atlassian.net/jira/people/team/ad77ab65-9522-497f-a6a0-01fb61476bb6>

Use the ID to designate the team in the XML file.

```
<lookup targetType="team" externalField="assignment_group" cockpitField="assignedTeam"
direction="IN">
  <valueMap externalValue="ad77ab65-9522-497f-a6a0-01fb61476bb6" cockpitValue="SUPPORT"/>
</lookup>
```


Source fichier XML

- Provide a concrete example of an XML file which will synchronize a Cockpit ITSM instance with a Jira Software instance.
- The example should serve as a basis for synchronizations with other Jira Software, allowing you to adapt the source file to your own particular context.

I. <connector> tag

```
<ticketSync xmlns="http://www.cockpit-itsm.com/TicketSyncConfiguration" description="Jira"
syncFrequencyInMinutes="5">
```

- Do not modify the "xmlns" attribute value.
- "**syncFrequencyInMinutes**" indicates the frequency of the synchronization in minutes.

```
<connector id="jira" username="user@company.com" password="Ry5ldxdfkcouHvwXFbse1D1D"
url="https://xxxxxxx.atlassian.net" defaultTimeZone="UTC" dateFormat="yyyy/MM/dd HH:mm">
```

- **Start of the <connector> tag.**
- "**id**" specifies the type of connector you will be using. For Jira Software, the connector type is "jira".
- "**username**" and "**password**" are respectively the user's email and the API token created in the previous part "User".
- "**url**" is the URL that is used to access to Jira Software.
- "**defaultTimeZone**": Default time zone used to interpret the date fields of the Jira Software instance. Indicate "UTC" if the time zone of the Jira user settings has been set in GMT+00:00 as seen in the previous part "User".
- "**dateFormat**": User date format, "yyyy/MM/dd HH:mm" is the default format. Check the format in the advanced settings of Jira.

```
<parameters>
  <entry>
    <string>traceDirectory</string>
    <string>/tmp/Jira</string>
  </entry>
```

- The log directory is found on the machine that hosts the portal. If you are in SaaS mode, you cannot access this element and must therefore remove the <entry /> tag.
- If you are in Premium mode, replace the "/tmp/Jira" path with the appropriate path.

```
<entry>
  <string>maxResults</string>
  <int>50</int>
</entry>
```

The number of recordings sent back to each synchronization is limited to 100 (limitation related to the Jira API). Adjust this limit according to ticket activity and synchronization frequency or set the maximum directly (100).

Note: If the record limit is exceeded during synchronization, excess data – e.g. creating a ticket – will not be taken into account.

```
<entry>
  <string>dateFields</string>
  <string>
    created,
    updated,
  </string>
</entry>
```

- Declaration of the date fields (creation and update) of Jira tickets.

```
<entry>
  <entry>
    <string>fields</string>
    <string>
      attachment,
      comment,
      created
      description,
      id,
      issuetype,
      priority,
      project,
      reporter,
      resolution,
      resolutiondate,
      status,
      summary,
      updated
    </string>
  </entry>
```

- List of ticket fields taken into account during synchronization.
- If you want other fields to be taken into account, they must be added to the API requests, then contact Cockpit ITSM Support to make this request. The added fields can be synchronized with Cockpit custom properties.

```
<entry>
  <string>journalFields</string>
  <string>comments,work_notes</string>
</entry>
```

- Allows you to include journal fields when creating a ticket.

- The second markup `<string>` contains the list of journal fields to synchronize. In the example above the fields “comments” and “work_notes”. If you do not want to synchronize these fields, do not indicate anything in this tag.

```
<entry>
  <string>readOnlyFields</string>
  <string>
    __type__,
    created,
    id,
    reporter,
    updated
  </string>
</entry>
```

- List of fields that will not be modified during synchronization (read-only).

```
<lastUpdateField>updated</lastUpdateField>
```

- Map the dates of the operations on the external tickets to maintain the chronological order of the operations on both sides (Cockpit ITSM and Jira).
- **End of the `<connector>` tag.**

II. Synchronization process

- In the following example we describe a synchronization process between the "Incident" tickets on the Cockpit ITSM side and the "Bug" on the Jira side.
- The synchronization will be done in both directions (Cockpit ITSM => Jira and vice versa).
- The process treats creations, updates, and closures separately.

A. Ticket type

```
<process ticketType="INCIDENT" discriminatorField="__type__" discriminatorValue="Bug"
ticketTemplate="JIRA">
```

- The synchronization process begins with the `<process />` tag.
- "**discriminatorField**" attribute: the "`__type__`" value corresponds to the type of ticket
- "**discriminatorValue**" attribute: corresponds to the INCIDENT process
- "**ticketTemplate**" attribute: Corresponds to the “Reference” field of the template that will be used when creating an Cockpit ITSM side ticket from Jira.

B. Mapping the IDs of tickets

```
<ticketIdMap externalField="id" cockpitField="externalReference"/>
```

- The <ticketIdMap /> tag maps Jira tickets with Cockpit ITSM tickets:
 - "**externalField**" attribute: contains the value of the "id" field, which corresponds to the ID field of the Jira ticket.
 - "**cockpitField**" attribute: contains the ID of the Cockpit ITSM ticket.
- If a ticket ID is not associated with another ID, then the operation is considered a creation.
- When a ticket ID is associated with another ID, then the operation is considered an update or a closure, depending on the status.

C. Creating a ticket

```
<create direction="BOTH">
```

- The <create /> tag is used to create tickets.
- If the "**BOTH**" value appears in the "direction" attribute, this means that the tickets created in Jira will also be created in Cockpit, and vice versa. It is therefore possible to synchronize both systems in just one direction by using the values "IN" (to Cockpit) and "OUT" (to Jira).

```
<filters>
  <filter direction="IN">
    <filterRule field="__operation__" value="CREATE"/>
    <filterRule field="reporter" value="API cockpit" inverted="true"/>
  </filter>
</filters>
```

- First of all, apply a filter to the incoming tickets (the "IN" in the "direction" attribute) to ignore the Jira tickets created by the "cockpit" user, i.e. the Jira user in charge of creating tickets (see "Configuration" part).
- Without this filter, you risk entering into an infinite loop.
- If the solutions are synchronized only goes one way, this filter is not useful.

1. Mapping of statuses and priorities: Jira => Cockpit ITSM

```
<mappers>
  <copy externalField="id" cockpitField="externalReference" direction="IN"/>
  <copy externalField="summary" cockpitField="title" stripHtml="true" direction="IN"/>
  <copy externalField="description" cockpitField="description" stripHtml="true" direction="IN"/>
  <copy externalField="created" cockpitField="creationDate" direction="IN"/>
```

- The <mappers> tag is used to indicate that we will map the different values of a field.
- The <copy> tag is used to map fields without modifying their contents. Example: text fields such as "ticket title".
- The "stripHtml" attribute is used to indicate whether or not we want to keep the rich text format:
 - True = the rich text is not kept
 - False = the rich text is kept
 - Default value if you don't set this line = False

- In this part we copy informations such as description, ticket ID, etc.

```
<lookup targetType="status" externalField="status" cockpitField="status" direction="IN">
  <valueMap externalValue="Open" cockpitValue="New"/>
  <valueMap externalValue="Reopened" cockpitValue="New"/>
  <valueMap externalValue="To Do" cockpitValue="To do"/>
  <valueMap externalValue="In Progress" cockpitValue="In progress"/>
  <valueMap externalValue="Building" cockpitValue="Waiting"/>
  <valueMap externalValue="*" cockpitValue="New"/>
</lookup>
```

- We use the <lookup> tag to map those fields which have predefined values (status, etc.).
- First, we map the field IDs with the attributes **externalField** et **cockpitField**.
- Then the field values are mapped with **externalValue** and **cockpitValue**.

Note: Use a <valueMap> tag with a value of "*" to ensure that a ticket is created even if a Jira status is not provided in the mapping.

- A **direction** attribute with a value of "IN" means we synchronize the tickets in the following direction: Jira => Cockpit. All subsequent <valueMap /> attributes also have the value "IN".
- All ticket fields that are not mapped will be ignored.

```
<lookup targetType="priority" externalField="priority" cockpitField="priority" direction="IN">
  <valueMap externalValue="Highest" cockpitValue="High"/>
  <valueMap externalValue="High" cockpitValue="High"/>
  <valueMap externalValue="Medium" cockpitValue="Medium"/>
  <valueMap externalValue="Low" cockpitValue="Low"/>
  <valueMap externalValue="Lowest" cockpitValue="Low"/>
</lookup>
```

- In this section <lookup> the priorities of Jira tickets are mapped to the Cockpit ITSM priorities. All priorities must be mapped to synchronize all tickets.

```
<lookup targetType="team" externalField="assignment_group" cockpitField="assignedTeam"
direction="IN">
  <valueMap externalValue="ad77ab65-9522-497f-a6a0-01fb61476bb6" cockpitValue="NETWORK"/
>
  <valueMap externalValue="*" cockpitValue="SUPPORT"/>
</lookup>
```

- Synchronization of teams, in this example the tickets of Jira team "ad77..." are synchronized with the Cockpit Network team.
- With the value " * ", tickets of all other Jira teams are synchronized with the Cockpit Support team.
- Without the value " * ", it is possible to synchronize only few teams.

2. Mapping of statuses and priorities: Cockpit ITSM => Jira

- The principle is the same as in the previous part.
- All "**direction**" attributes have the value "OUT".

```
<set externalField="__type__" cockpitField="N/A" value="Bug" direction="OUT"/>
<copy externalField="project" cockpitField="N/A" value="SUPPORT_COCKPIT" direction="OUT"/>
<copy externalField="summary" cockpitField="title" stripHtml="true" direction="OUT"/>
<copy externalField="description" cockpitField="description" wikiMarkup="true" direction="OUT"/>
```

- In Jira a ticket belongs to a project, here the project is “SUPPORT_COCKPIT”.
- This means that a ticket type (e. g. Bug) from the Cockpit always creates a ticket in the same project.
- The attribute “wikiMarkup” is the equivalent of “stripHtml” but on the Jira side. In the “OUT” synchronisation it is therefore necessary to use this attribute to keep or not the rich text:
 - True: Rich text is kept
 - False: Rich text is not kept

```
<lookup targetType="priority" externalField="priority" cockpitField="priority" direction="OUT">
  <valueMap externalValue="Highest" cockpitValue="High"/>
  <valueMap externalValue="High" cockpitValue="High"/>
  <valueMap externalValue="Medium" cockpitValue="Medium"/>
  <valueMap externalField="Low" cockpitValue="Low"/>
  <valueMap externalField="Lowest" cockpitValue="Low"/>
</lookup>
</mappers>
</create>
```

- The Cockpit priorities are mapped with Jira priorities.
- The statuses are not mapped because when a ticket is created in Jira, the status depends on the workflow of the project where the ticket is created.
- End of the ticket creation process.

D. Updating a ticket

```
<update discriminatorField="__operation__" discriminatorValues="UPDATE" attachmentPrivacy="PUBLIC"
direction="BOTH">
```

- The `<update />` tag is used to update tickets.
- "**discriminatorField**" attribute: the "`__operation__`" value indicates that we are interested in operations.
- "**discriminatorValue**" attribute: the "UPDATE" value indicates that we will only take update operations into consideration.
- "**direction**" attribute: the updating of the tickets will be done in both directions, from Jira to Cockpit ITSM and vice versa.

```
<filters>
  <filter direction="IN">
    <filterRule field="updateAuthor" value="API Cockpit" inverted="true"/>
    <filterRule field="reporter" value="API Cockpit" inverted="true"/>
  </filter>
  <filter direction="OUT">
```

```

    <filterRule field="privacy" value="PUBLIC"/>
  </filter>
  <filter direction="OUT">
    <filterRule field="type" value="EDIT_PROPERTIES"/>
  </filter>
  <filter direction="OUT">
    <filterRule field="type" value="EDIT_TREATMENT"/>
  </filter>
  <filter direction="OUT">
    <filterRule field="type" value="ATTACHMENT_ADD"/>
  </filter>
</filters>

```

- Just like for the creation of tickets, a filter is applied to the incoming tickets to disregard the Jira tickets created by the "API Cockpit" user and thus avoid a loop.
- A filter is applied to the outgoing tickets to only take into account those Cockpit ITSM ticket upgrades which are performed with a confidentiality level of "Public".
- Two filters on outgoing tickets are applied to update Jira tickets when the properties of Cockpit ITSM tickets are modified:
 - "EDIT_PROPERTIES": Priority changes are synchronized
 - "EDIT_TREATMENT": Status changes are synchronized

The principle is as follows:

- Priority and status changes of Cockpit ITSM tickets are made in private mode, if the filter only taking into account public exchanges is set, priority and status changes would not be taken into account during synchronization.
This is why it is necessary to add the 2 filters "EDIT_PROPERTIES" and "EDIT_TREATMENT".
- If the "PUBLIC" filter is not set, all Cockpit ITSM ticket changes, public and private, are taken into account, then it is not necessary to add the "EDIT_PROPERTIES" and "EDIT_TREATMENT" filters.

1. Mapping of statuses, priorities and teams : Jira => Cockpit

```

<mappers>
<copy externalField="description" cockpitField="transientMessage" stripHtml="true" direction="IN"/>
<template externalField="body" cockpitField="transientMessage" direction="IN"><![CDATA[$data["body"]]
<br />
$data["updateAuthor"]]]></template>
<copy externalField="updated" cockpitField="lastUpdate" direction="IN"/>

<lookup targetType="status" externalField="status" cockpitField="status" direction="IN">
  <valueMap externalValue="Open" cockpitValue="New"/>
  <valueMap externalValue="Reopened" cockpitValue="New"/>
  <valueMap externalValue="To Do" cockpitValue="To do"/>
  <valueMap externalValue="In Progress" cockpitValue="In progress"/>
  <valueMap externalValue="Building" cockpitValue="Waiting"/>
</lookup>

```

```
<lookup targetType="priority" externalField="priority" cockpitField="priority" direction="IN">
  <valueMap externalValue="Highest" cockpitValue="High"/>
  <valueMap externalValue="High" cockpitValue="High"/>
  <valueMap externalValue="Medium" cockpitValue="Medium"/>
  <valueMap externalValue="Low" cockpitValue="Low"/>
  <valueMap externalValue="Lowest" cockpitValue="Low"/>
</lookup>
```

- Same principles as for the creation of tickets.
- The tag `<template />` allows you to add text and variables.

In this example we add to the description of the ticket (“body”) a line break (`
`) then the name of the Jira user who added the exchange (“updateAuthor” variable).

It would also have been possible to add text.

To see the list of available variables, contact the Cockpit ITSM support.

2. Mapping of statuses, priorities and teams : Cockpit => Jira

```
<copy externalField="id" cockpitField="externalReference" direction="OUT"/>
<copy externalField="body" cockpitField="transientMessage" wikiMarkup="true" direction="OUT"/>

<lookup targetType="status" externalField="status" cockpitField="status" direction="OUT">
  <valueMap cockpitValue="To Do" externalValue="New"/>
  <valueMap cockpitValue="To Do" externalValue="To do"/>
  <valueMap cockpitValue="In Progress" externalValue="In progress"/>
  <valueMap cockpitValue="Building" externalValue="Waiting"/>
  <valueMap cockpitValue="Done" externalValue="Solved"/>
</lookup>

<lookup targetType="priority" externalField="priority" cockpitField="priority" direction="OUT">
  <valueMap externalValue="Highest" cockpitValue="High"/>
  <valueMap externalValue="High" cockpitValue="High"/>
  <valueMap externalValue="Medium" cockpitValue="Medium"/>
  <valueMap externalValue="Low" cockpitValue="Low"/>
  <valueMap externalValue="Lowest" cockpitValue="Low"/>
</lookup>
</mappers>
</update>
```

- Same principles as for the creation of tickets.

E. Closing of a ticket

```
<close discriminatorField="status" discriminatorValues="Done" attachmentPrivacy="PUBLIC"
direction="BOTH">
  <filters>
    <filter direction="IN">
      <filterRule field="updateAuthor" value="API Cockpit" inverted="true"/>
      <filterRule field="reporter" value="API Cockpit" inverted="true"/>
    </filter>
  </filters>
```


- The <close /> tag is used to close tickets.
- Just like for the creation of tickets, a filter is applied to the incoming tickets to disregard the Jira tickets created by the "API Cockpit" user and thus avoid a loop.
- "**discriminatorField**" attribute: the "status" value indicates that we are interested in ticket status.
- "**discriminatorValue**" attribute: the "Done" value indicates that we will only take the closing of tickets into consideration.
- "**direction**" attribute: the updating of the tickets will be done in both directions, from Jira to Cockpit ITSM and vice versa.

1. Mapping of fields : Jira => Cockpit

```
<mappers>
<copy externalField="body" cockpitField="transientMessage" direction="IN"/>
<copy externalField="updated" cockpitField="lastUpdate" direction="IN"/>
<copy externalField="updated" cockpitField="realSolutionDate" direction="IN"/>

<lookup targetType="status" externalField="status" cockpitField="status" direction="IN">
  <valueMap externalValue="Done" cockpitValue="Solved"/>
  <valueMap externalValue="*" cockpitValue="Solved"/>
</lookup>

<lookup targetType="priority" externalField="priority" cockpitField="priority" direction="IN">
  <valueMap externalValue="Highest" cockpitValue="High"/>
  <valueMap externalValue="High" cockpitValue="High"/>
  <valueMap externalValue="Medium" cockpitValue="Medium"/>
  <valueMap externalValue="Low" cockpitValue="Low"/>
  <valueMap externalValue="Lowest" cockpitValue="Low"/>
</lookup>
```

- The Cockpit fields "lastUpdate" and "realSolutionDate" are mandatory, they will contain the same value from the Jira field "updated".
- The Cockpit field "realSolutionDate" corresponds to the archive of the ticket made by the end users, the ticket cannot be reopened.

2. Mapping of fields : Cockpit => Jira

```
<copy externalField="id" cockpitField="externalReference" direction="OUT"/>
<copy externalField="body" cockpitField="response" wikiMarkup="true" direction="OUT"/>

<lookup targetType="status" externalField="status" cockpitField="status" direction="OUT">
  <valueMap externalValue="Done" cockpitValue="Solved"/>
  <valueMap externalValue="Done" cockpitValue="*" />
</lookup>
</mappers>
</close>
```

Document end