

How to Choose a Destination Project Based on the Parent Option of a Cascading Select Custom Field

Last Modified on 04/03/2024 6:21 am EDT

Introduction

Imagine you have two instances, **SD** and **DEV**, and you need to synchronize issues from SD to DEV. In SD you have a *Cascading Select List customField*, having two values: *product (parent)* & *version (child)*. You want to select the correct DEV project based on the cascading field parent value 'product'. Let's say you have the following products and corresponding DEV projects:

- *product_X* => PRODX
- *product_Y* => PRODY
- *product_Z* => PRODZ

The scripts below help to map the parent value from the cascading select list to the projects.

Sending side

Outgoing sync

```
1 replica.customFields."Product & Version" = issue.customFields."Product & Version"
```

Receiving side

Map the parent value from the cascading select to the projects and if a mapping for the option is not found or the option is not selected, use PRODX as a default.

Incoming sync

```
1 def projectMapping = [  
2 "product_X": "PRODX"  
3 "product_Y": "PRODY"  
4 "product_Z": "PRODZ"  
5 ]  
6  
7 issue.projectKey = projectMapping[replica.customFields."Product & Version"?.value?.parent?.value] ?: "PRODX"
```

Drop the synchronization (indicate that you don't want to create an issue on your side) if you don't find a mapping.

Incoming sync

```
1 def projectMapping = [  
2 "product_X": "PRODX"  
3 "product_Y": "PRODY"  
4 "product_Z": "PRODZ"  
5 ]  
6  
7 def projectKey = projectMapping[replica.customFields."Product & Version"?.value?.parent?.value]  
8  
9 if (projectKey != null) {  
10 issue.projectKey = projectKey  
11 // your entire create processor goes here
```

Product ...

[About Us](#)

[Release History](#)

[Glossary](#)

ON THIS PAGE

[API Reference](#)

[Security](#)

[Pricing and Licensing](#)

Resources

[Academy](#)

[Blog](#)

[YouTube Channel](#)

[Ebooks](#)

Still need help?

[Join our Community](#)

[Visit our Service Desk](#)

[Find a Partner](#)