

# How to calculate Risk ranking using ScriptRunner add-on

 Keep in mind, this setup is dependant on implementation of a third party solution - as such, it is not supported by SoftwarePlant. This page serves to advise you on availability of such option.

## User story

As a Risk manager, I would like to order risks visualised by BigPicture. I need a list which has the most dangerous risks at the top and the least significant at the bottom.

## Custom solution

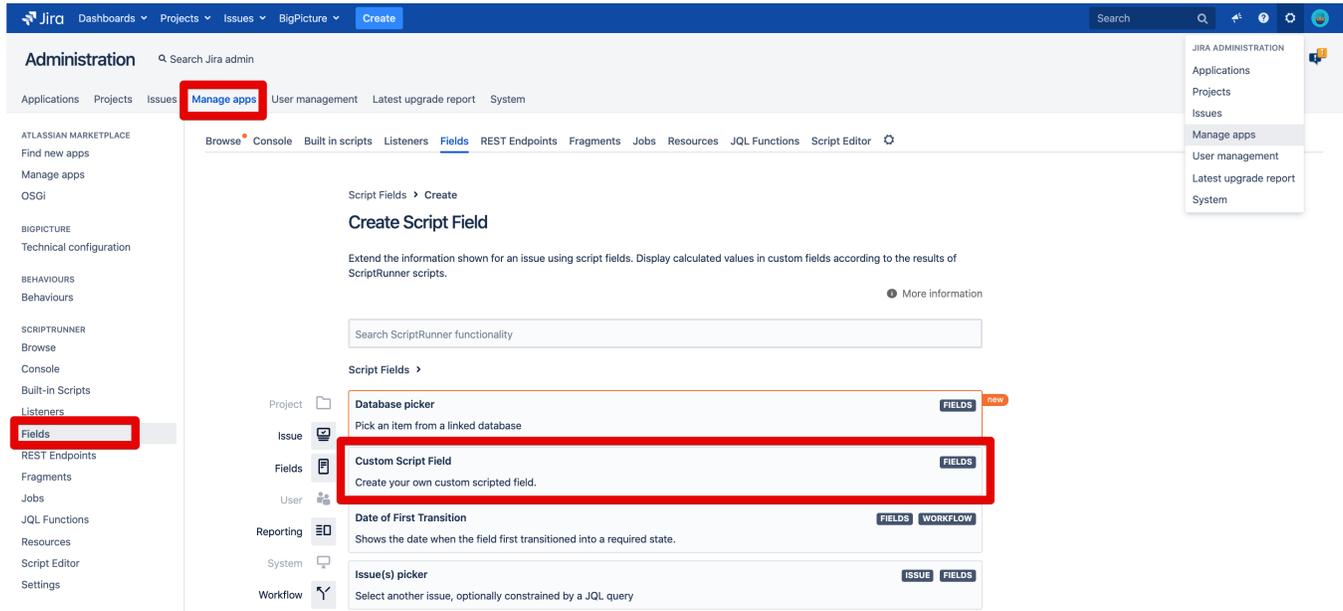
### Assuming that

?- the Jira instance has ScriptRunner installed ([link to marketplace add-on](#))

?- there are two custom fields, created automatically by BigPicture called *Risk probability* and *Risk consequence*

### Configuration needed

If the above is satisfied, then go through the steps of creating a ScriptRunner Scripted field which requires a Jira Admin access. Screenshot bellow marks the key steps to reach this configuration:



The screenshot shows the Jira Administration console. The top navigation bar includes 'Jira', 'Dashboards', 'Projects', 'Issues', 'BigPicture', and 'Create'. The left sidebar shows the 'Administration' menu with 'Manage apps' highlighted in red. The main content area is titled 'Create Script Field' and includes a search bar for ScriptRunner functionality. Below the search bar, there is a list of script fields with the following items:

- Database picker** (Project icon): Pick an item from a linked database. (FIELDS) NEW
- Custom Script Field** (Fields icon): Create your own custom scripted field. (FIELDS)
- Date of First Transition** (Reporting icon): Shows the date when the field first transitioned into a required state. (FIELDS) WORKFLOW
- Issue(s) picker** (System icon): Select another issue, optionally constrained by a JQL query. (ISSUE) (FIELDS)

In the next step paste the code below into the text area marked on the screenshot:

### Custom Script Field [?](#)

Create your own custom scripted field.

Field Name

The name of the custom field that you are creating

Field Description

The description of the custom field that you are creating

Note

An optional note, used only for your reference.

Template **Text Field (multi-line)**

Template to represent your field. Match to the output of your script.



```

SCRIPT FILE
Script
1 import com.atlassian.jira.component.ComponentAccesso
2 import com.atlassian.jira.issue.CustomFieldManager
3 import com.atlassian.jira.issue.Issue
4 import com.atlassian.jira.issue.fields.CustomField
5 import org.apache.log4j.Logger
6
7 /**
8  * Script which reads text value from two custom fie
9  * By default, two custom fields which are taken int
10 * - 'Risk probability'
11 * - 'Risk consequence'
12 *
13 * For each field a set of possible values is define
14 * Each value is assigned a numerical value which is
15 * Assignment of text to a number is done in 'switch
16 * To map a new text value to a number, insert 3 lin
17 * case "Input the text value to map":
18 *     consequenceNumberValue = "Input the number t
19 *     break;
20 * Example of such new mapping for Risk consequence:
21 * case "Critical":
22 *     consequenceNumberValue = 9;
23 *

```

?

#### Code with custom logic calculating the numerical *Risk ranking*?

```

import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.CustomFieldManager
import com.atlassian.jira.issue.Issue
import com.atlassian.jira.issue.fields.CustomField
import org.apache.log4j.Logger

/**
 * Script which reads text value from two custom fields and converts then to a numeric value.
 * By default, two custom fields which are taken into consideration:
 * - 'Risk probability'
 * - 'Risk consequence'
 *
 * For each field a set of possible values is defined in this script.
 * Each value is assigned a numerical value which is later used for multiplication.
 * Assignment of text to a number is done in 'switch' blocks.
 * To map a new text value to a number, insert 3 lines:
 * case "Input the text value to map":
 *     consequenceNumberValue = "Input the number to be mapped as a n integer e.g. 5";
 *     break;
 * Example of such new mapping for Risk consequence:
 * case "Critical":
 *     consequenceNumberValue = 9;
 *     break;

```

```

* Example for new Risk probability mapping:
* case "Very low":
*     probabilityNumberValue = 1;
*     break;
*
* Script can be extended to calculate the Ranking with different formula and using different amount of data.
* */

def log = Logger.getLogger("scripts.risk.RiskRankingFieldFormula")
Issue issue = issue;
String riskProbabilityName = "Risk probability";
String riskConsequenceName = "Risk consequence";
CustomFieldManager customFieldManager = ComponentAccessor.getCustomFieldManager();
try {
    List<CustomField> listOfConsequenceCustomFields = customFieldManager.getCustomFieldObjectsByName
(riskConsequenceName);
    CustomField riskCustomField;
    if (listOfConsequenceCustomFields.isEmpty()) {
        throw new Exception("Could not find a custom field with name '" + riskConsequenceName + "'");
    } else {
        riskCustomField = listOfConsequenceCustomFields.get(0);
    }

    String consequenceTextValue = issue.getCustomFieldValue(riskCustomField);
    Integer consequenceNumberValue;

    switch (consequenceTextValue) {
        /** Insert new mappings after this line*/
        case "Low":
            consequenceNumberValue = 1;
            break;
        case "Medium":
            consequenceNumberValue = 2;
            break;
        case "High":
            consequenceNumberValue = 3;
            break;
        default: //Setting the value to zero will force the multiplication
            consequenceNumberValue = 0;
    }

    List<CustomField> listOfProbabilityCustomFields = customFieldManager.getCustomFieldObjectsByName
(riskProbabilityName);
    CustomField probabilityCustomField;
    if (listOfProbabilityCustomFields.isEmpty()) {
        throw new Exception("Could not find a custom field with name '" + riskProbabilityName + "'");
    } else {
        probabilityCustomField = listOfProbabilityCustomFields.get(0)
    }
    String probabilityTextValue = issue.getCustomFieldValue(probabilityCustomField)
    Integer probabilityNumberValue;
    switch (probabilityTextValue) {
        /** Insert new mappings after this line*/
        case "Low":
            probabilityNumberValue = 1;
            break;
        case "Medium":
            probabilityNumberValue = 2;
            break;
        case "High":
            probabilityNumberValue = 3;
            break;
        default:
            probabilityNumberValue = 0;
    }
    /** Multiply both numbers values to receive a final Risk ranking as an integer*/
    Integer calculatedRiskRanking = probabilityNumberValue * consequenceNumberValue;
    return calculatedRiskRanking;
} catch (Exception e) {

```

```
log.error("Caught exception: " + e);
// e.printStackTrace()
/** Return -1 to indicate that something did not go as planning in the algorithm*/
return -1;
}
```

## The result

This will add to Jira a Script field which has to be added to Jira view screens in order to see the result of the script.

## Sorting capabilities

After adding the Script field, you need to configure the field to be sortable.

After this configuration you will be able to sort Jira Filters by the *Risk ranking*

Search

✓ ORDER BY "Risk ranking" DESC

1-6 of 6

T	Key	Summary	Risk ranking ↓
<input checked="" type="checkbox"/>	SAM-1	This is your first task	9
<input checked="" type="checkbox"/>	SAM-2	Workflows and statuses	3
<input checked="" type="checkbox"/>	SAM-5	Keyboard shortcuts	0
<input checked="" type="checkbox"/>	SAM-3	Editing tasks	0
<input checked="" type="checkbox"/>	SAM-4	Searching for information	0
<input checked="" type="checkbox"/>	SAM-6	What's next?	0

1-6 of 6

## Searching capabilities - requires additional configuration step

To enable searching by the "Risk ranking" field, there is one additional step required.

To configure this navigate to *Custom fields* configuration page in Jira administration.

## Custom fields

Optimize

Add custom field

Add extra fields to your issues to define them more precisely. From simple text fields to development summaries, you can create different types of custom fields and configure how they're displayed to your users. Here you can manage your existing custom fields, or create new ones.

Q risk Project: All Type: All Screen: All

Name	Type	Available Contexts	Screens	Actions
<b>Risk consequence</b> Risk Consequence field used by BigPicture plugin. (Only Issues with b...	Select List (singl...	Global (all projec...	2 screens	
<b>Risk probability</b> Risk Probability field used by BigPicture plugin. (Only Issues with bot...	Select List (singl...	Global (all projec...	2 screens	
<b>Risk ranking</b> Risk ranking calculated as a multiplication of two factors used by BigP...	Scripted Field	Global (all projec...	2 screens	

- Configure
- Edit
- Translate
- Screens
- Delete

On the *Edit* page set the *Search Template* to *Number Searcher*.

In order to see the effect, you need to perform the re-index of your Jira.

As a result, you will be able to query the data in Jira Filters

✔ "Risk ranking" > 3

1-1 of 1

T	Key	Summary	Risk ranking
<input checked="" type="checkbox"/>	SAM-1	This is your first task	9

Have fun => and in case of questions, contact us?via the?Service Desk. We are always happy to help.