Migrate JWT parser expressions

The JWT Cloud expression parser differs from the JWT DC expression parser; therefore, the migrating parser expressions is often a complex task.

Besides field codes and parser functions, additional mappings of parsing modes, time macros, etc., apply. This page provides an overview of what is mapped and how.

Parsing modes

The parser in JWT DC offers different modes, which can be mapped to the modes provided by the JWT Cloud expression parser.

<table>
<tr><th>JWT DC</th><th>Mapping</th><th>JWT Cloud</th><th>Notes</th></tr>
<tr><td>Basic text mode</td><td>✓</td><td>General mode</td><td>Replace the field codes by the ones provided in JWT Cloud, for details see Migrate field codes.</td></tr>
<tr><td>Advanced text mode</td><td>✓</td><td>General mode</td><td>The whole parser expression (which has been migrated as described in Migrate field codes and Migrate JWT parser functions) has to be enclosed by %{ }

If you use the "+" operator with a numeric value as the first operator in the expression, it has to be converted to a text using toString() or if using a numeric field code, change it to a text filed code.

<b>Examples</b>
<table>
<tr><th>JWT DC</th><th>JWT Cloud</th></tr>
<tr><td><code>{issue.cf10004} +1</code></td><td><code>%{toString({issue.cf10004}) +1}</code>
<code>%{%{issue.cf10004} +1}</code></td></tr>
<tr><td><code>"This is a concatenated value "+ {issue.cf10004} +1</code></td><td><code>%{"This is a concatenated value "+ {issue.cf10004} +1}</code></td></tr>
</table>
</td></tr>
<tr><td>Logical mode</td><td>✓</td><td>Logical mode</td><td>For differences between JWT DC and JWT Cloud logical mode, see Migrate logical expressions</td></tr>
<tr><td>Numeric mode</td><td>✓</td><td>General mode</td><td>The whole parser expression (which has been migrated as described in Migrate field codes and Migrate JWT parser functions) has to be enclosed by %{ }</td></tr>
<tr><td>Text list mode</td><td>✓</td><td>General mode</td><td>The whole parser expression (which has been migrated as described in Migrate field codes and Migrate JWT parser functions) has to be enclosed by %{ }</td></tr>
<tr><td>Issue list mode</td><td>✓</td><td>General mode</td><td>The whole parser expression (which has been migrated as described in Migrate field codes and Migrate JWT parser functions) has to be enclosed by %{ }</td></tr>
</table>

<table>
<tr><td>JQL mode</td><td>✓</td><td>General mode</td><td>The field codes which are used in the JQL query have to be migrated according to Migrate field codes. The whole expression has to be placed inside %{issuesFromJQL()}, re-writing the JWT DC expression as an addition of strings, such that the field codes will be evaluated before executing the JQL.

<b>Example</b>
<table><tr><th>JWT DC</th><th>JWT Cloud</th></tr><tr><td><code>issuetype = "%{issue.issueType}" AND project = "%{00019}"</code></td><td><code>%{issuesFromJQL("issuetype = '" + %{issue.issueType} + "' AND project = '" + %{issue.project.name} + "'")}</code></td></tr></table></td></tr>
<tr><td>Mixed mode</td><td>✓</td><td>General mode</td><td>Replace <code>{{{</code> and <code>}}}</code> by <code>%{</code> and <b>}</b> (and migrate the fields and functions as described in Migrate field codes and Migrate JWT parser functions)</td></tr>
</table>

List operators

The list operators APPEND, EXCEPT, INTERSECT and UNION available in JWT DC have no direct equivalent in JWT Cloud. However, you can use the expression parser functions append(), except(), intersect() and union() instead.

Field codes

Most of the field codes known in JWT DC are also available in JWT Cloud - see Migrate field codes.

Parser functions

Many of the parser functions known in JWT DC are also available in JWT Cloud - see Migrate JWT parser functions.

If you still have questions, feel free to refer to our support team.