Artifact Rules

Artifacts are files that registered in Continua CI as important files. By registering a file as an artifact, it can then be viewed through the browser or downloaded to the current user's computer.

> **Workspace rules**
>
> Please note that all Artifact Rules refer to the server's workspace, NOT the agent's workspace. The file must exist in the server's workspace before it can be registered as an artifact. To transfer files from the agent to the server, use Workspace Rules.

1. Artifacts without view types

Rule Example Type 1 - Simple Artifacts

```
[1.1] Output\**.xml
[1.2] Output\*\**.exe
[1.3] Output\log.txt
```

[1.1] - Registers all files with the "**.xml**" extension in the **Output** directory as artifacts.

[1.2] - Registers all files with the "**.exe**" extension in any **directories** underneath the **Output** directory. It won't match any "**.exe**" files directly in the **Output** directory.

[1.3] - Registers the single file **log.txt** in the **Output** directory.

2. Artifacts with view types

Rule Example Type 2 - Artifact view types

```
[2.1] nunit | Output\test*.xml
[2.2] image | Output\**.png
```

[2.1] - Registers all files that start with "**test**" and end with "**.xml**" in and only in the **Output** directory as artifacts and registers them as viewable by the **nunit** viewer.

[2.2] - Registers all files with the "**.png**" extension in the **Output** directory and all directories underneath it as artifacts and registers them as viewable by the **image** viewer.

3. Extracting archive artifacts.

Add the colon ":" operator after a file path with a zip extension on the left-hand side of a rule to specify that the archive file should be extracted to the path on the right side of the rule. Archive rules are processed before all other rules. Note that only zip archive files are currently supported.

Rule Example Type 3 - Archive extracting rules

```
3.1 Report.zip: > Report/ExtractedFiles
3.2 Report.zip: -> Report/ExtractedFlattened
3.3 Report.zip: >> Report/ExtractedFiles
3.4 Report.zip: ->> Report/ExtractedFlattened

3.5 **/Report.zip: > Report/ExtractedFiles
3.6 Report*.zip: -> Report/ExtractedFlattened
3.7 Report?.zip: >> Report/ExtractedFiles
3.8 */Report.zip: ->> Report/ExtractedFlattened
```

The archive files in the server workspace are automatically extracted at the end of the stage. Note that the operators for preserving and emptying the destination folder are also taken into account when extracting.

[3.1] - Extracts all the files in the archive **Report.zip** to the folder **Report/ExtractedFiles**. Preserves the directory structure within the zip file.

[3.2] - Extracts all the files in the archive **Report.zip** to the folder **Report/ExtractedFlattened**. Flattens the zip directory structure so that all files are extracted directly to **Report/ExtractedFlattened**.

[3.3] - Extracts all the files in the archive **Report.zip** to the folder **Report/ExtractedFiles**. Empties the destination folder before extracting files.

[3.4] - Extracts all the files in the archive **Report.zip** to the folder **Report/ExtractedFlattened**. Empties the destination folder before extracting files. Flattens the zip directory structure so that all files are extracted directly to **Report/ExtractedFlattened**.

[3.5] - Extracts all the files in all archive files named **Report.zip** in any subfolder recursively to the folder **Report/ExtractedFiles**. Preserves the directory structure within the zip file.

[3.6] - Extracts all the files in all archive files with a name starting with **Report** and ending with **.zip** to the folder **Report/ExtractedFlattened**. Flattens the zip directory structure so that all files are extracted directly to **Report/ExtractedFlattened**.

[3.7] - Extracts all the files in all archive files with a name starting with **Report**, followed by a single character, followed by **.zip** to the folder **Report /ExtractedFiles**. Empties the destination folder before extracting files.[3.8] - Extracts all the files in all archive files named **Report.zip** in any top-level subfolder to the folder **Report/ExtractedFlattened**. Empties the destination folder before extracting files. Flattens the zip directory structure so that all files are extracted directly to **Report/ExtractedFlattened**.

4. Extracting archives using wildcards

You can also add a pattern to specify which files to extract from the server archive file.

Rule Example Type 4 - Archive extracting rules with wildcards

```
4.1 Report.zip:/*.html > ReportHtmlFiles
4.2 Report.zip:/**.html > Report/HtmlFiles
4.3 Report.zip:/Main/**.xml > MainReport/XmlFiles
4.4 Report.zip:/Report/**/Main/**.html > MainReport/HtmlFiles
```

[4.1] - Extracts all the **html** files in the root folder of the archive **Report.zip** to the folder **ReportHtmlFiles**. Preserves the directory structure within the zip file.

[4.2] - Extracts all the **html** files in the archive **Report.zip** to the folder **Report/HtmlFiles**. Preserves the directory structure within the zip file.

[4.3] - Extracts all the **xml** files under the Main folder in the archive **Report.zip** to the folder **MainReport/XmlFiles**. Preserves the directory structure under the Main folder within the zip file.

[4.4] - Extracts all the **html** files under the Report folder in the archive **Report.zip** which are under a sub-folder named Main to the folder **MainReport /HtmlFiles**. Preserves the directory structure under the Report folder within the zip file.