

Accessing Uploaded Files

This document describes the technical approach to accessing and using files that were uploaded during form submission.

Files are propagated to all actions in parameters and can be referenced by the input name i.e. *file-upload_0_2*

With SaveToJCR Action

When creating an action that requires accessing an uploaded file from the JCR, **SaveToJCR action** *must* be the **FIRST** action placed on a form!

File paths are replaced by absolute url after *SaveToJCR* action executes. The result of asking for "*file-upload_0_2*" is:

http://localhost:8080/files/live/sites/mySite/formFactory/results/ff1/submissions/05/01/08/37/33/37b7d633-7b2f-4f5d-80b1-f13d33600791/file-upload_0_2/Screen%20Shot%202018-03-20%20at%203.05.25%20PM.png

Which is the url of the file in JCR.

[*Note encoded characters should be replaced by their respective value]

You are then able to retrieve the file node by making a call: `session.getNode(url);`

Without SaveToJCR Action

When files are uploaded, they are saved under the server's default temporary directory. For Tomcat it would be saved under tmp folder.

The file path is provided in the parameters variable of the action. Access it by specifying the name of the file input i.e. *file-upload_0_2*

You are able to also access the original file name and file type.

Accessing File Example

We will look at a code example illustrated below. This is a simple procedure to locate the files within the parameters variable, which is passed to our action.

Retrieve all fileUploadDefinition nodes from the form to obtain the name of the input

```
NodeIterator childrenOfType = JCRContentUtils.getDescendantNodes(formStructure,
    "fcnt:fileUploadDefinition");
List<String> fileInputName = new ArrayList<String>(4);
while (childrenOfType.hasNext()) {
    JCRNodeWrapper children = (JCRNodeWrapper) childrenOfType.next();
    fileInputName.add(children.getName());
}
```

We can then iterate over the file names and verify which parameter matches our filename

```
for (String fileName : fileInputName) {
    if (parameters.containsKey(fileName)) {
        List<String> fileInfo = parameters.get(fileName);
        String fileTempPath = fileInfo.get(TEMP_PATH_INDEX); // 0
        String fileOrigName = fileInfo.get(ORIGINAL_NAME_INDEX); // 1
        String fileType = fileInfo.get(TYPE_INDEX); // 2

        .... Do stuff
    }
}
```

***Note** - The above structure is only valid before SaveToJCR action has executed.

After SaveToJCR action has execute, **fileInfo** object will only contain the absolute url at *index 0*. The rest of the file information can be obtained by accessing the node, using the url, from the JCR.

Action Ordering

Form Factory does not restrict you in the composition of your actions.

In terms of action ordering, you have full control of how to order actions. When it comes to form submission, there should be no difference as to which action is performed first, actions do not interact with each other.

Code can be adapted to work with raw files directly from tmp folder, in this case *SaveToJCR* is not required.

If a later action requires the file to be in the JCR, *SaveToJCR* should be placed before said action.