

Import Dateinamen verwenden

In diesem Beispiel enthalten die Dateinamen der importierten Datei bereits Indexinformationen, die wir in Kopffelder schreiben wollen.

Was passiert mit importierten Dokumenten?

Beim Import von Dokumenten können einzelne Dateien in mehrere resultieren.

Ein Beispiel dafür sind E-Mails die mehreren Anhängen, die alle als einzelne Eingangsdokumente behandelt werden sollen. Des weiteren werden auch XML-Anlagen innerhalb von PDFs oder direkt als Anhang für diese Funktionalität berücksichtigt.

Damit die Import Dateinamen sowie deren Zuordnung zur verarbeiteten Datei in SQUEEZE also nicht verloren gehen, werden diese Informationen gespeichert.

Zu jedem SQUEEZE Dateinamen gibt es dann also mindestens einen oder im Zweifel mehrere Import Dateinamen.

AdditionalInfo

Zu jedem SQUEEZE Vorgang (xDoc) gibt es `AdditionalInfo`, welche zusätzliche Informationen zu den Vorgängen und Dateien abspeichert.

Aktuell ist dies eine JSON Datei die am Vorgang im Repository innerhalb des "Work" Ordners abgespeichert ist.

Darin enthalten sind neben Pipeline Keys auch der/die Dateiname/n der importierten Datei/en.

FileImportMap

Die AdditionalInfo eines SQUEEZE Vorgangs (xDoc) enthält also auch eine `FileImportMap`.

Diese stellt ein Mapping zwischen den Dateinamen der importierten Dateien zu den jeweiligen SQUEEZE Dateinamen, die zur Bearbeitung / Extraktion verwendet werden.

Attachments behalten ihre originalen Dateinamen, der Zugriff zu ihnen ist über das Attachment Handling des SQUEEZE Vorgangs (xDoc) möglich und wird in diesem Beispiel nicht näher beschrieben.

Datenstruktur

In der `_additionalInfo.json` sieht das Mapping so aus:

Mit dem Zugriff auf ein Mapping (also einen SQUEEZE Dateinamen) erhält man ein Array mit den Import Dateinamen, aus welcher dann die SQUEEZE Datei geworden ist.

```
{
  "additionalInfo": {
    "IMPORT_FILENAMES": {
      "0002_64999a0020378394230462.xml": [
        "import.pdf"
      ],
      "64999a0020378394230462.pdf": [
        "another.pdf",
        "import.pdf"
      ]
    }
  }
}
```

In diesem Beispiel JSON gibt es also eine Datei namens "import.pdf" welche zusätzlich XML Content enthält und durch die Stapelklasseneigenschaft wurde "another.pdf" mit "import.pdf" zusammengefügt. Zu den SQUEEZE Dateinamen (hier die Schlüssel "0002_64999a0020378394230462.xml" und "64999a0020378394230462.pdf") werden die ursprünglichen Dateinamen gespeichert.

Die `FileImportMap` konzentriert sich nur auf den JSON Schlüssel "IMPORT_FILENAMES".

Das `->get()` der `FileImportMap` gibt also das Array mit den originalen Dateinamen der angegebenen SQUEEZE Datei zurück.

Beispiel

Zugriff im UserExit

Um auf die originalen Dateinamen beim Import zuzugreifen werden die `AdditionalInfo` benötigt,

die wiederum die Zugriffsklasse `FileImportMap` enthält.

Mit dieser ist es einfach, zu allen SQUEEZE Dateinamen die jeweiligen Import Dateinamen zu erhalten.

Außerdem kann man zusätzlich Dateinamen einem SQUEEZE Dateinamen zuordnen.

Das `->set()` fügt den Import Dateinamen zum Mapping des übergebenen SQUEEZE Dateinamen hinzu oder legt dieses Mapping an, sollte es noch nicht existieren.

```
// get AdditionalInfo
$additionalInfo = $xDoc->loadAdditionalInfoTyped();
// get FileImportMap
$fileImportMap = $additionalInfo->getFileImportMap();

// get all mapped import file names
$allSqueezeDocumentsToImportDocuments = $fileImportMap->getAll();
foreach($allSqueezeDocumentsToImportDocuments as $squeezeFileName => $importDocuments) {
    foreach($importDocuments as $importFileName) {
        // do something
    }
}

// get import file names for squeeze file name
$importFileNamesOf = $fileImportMap->get($squeezeFileName);
foreach($importFileNamesOf as $importFileName) {
    // do something
}

// add import file name for squeeze file name
$fileImportMap->set($importFileName, $squeezeFileName);
```

Revision #3

Created 5 July 2023 08:56:11 by Maximillian Weitze

Updated 5 February 2024 09:19:34 by Vahdettin Balum