



Moving and Renaming Files Between Directories

This topic describes how to move specific files from a source directory to a destination directory, and then specify a new name for the files that were moved.

This how-to includes the following component that is packaged in the *samples_move_files.zip* archive:

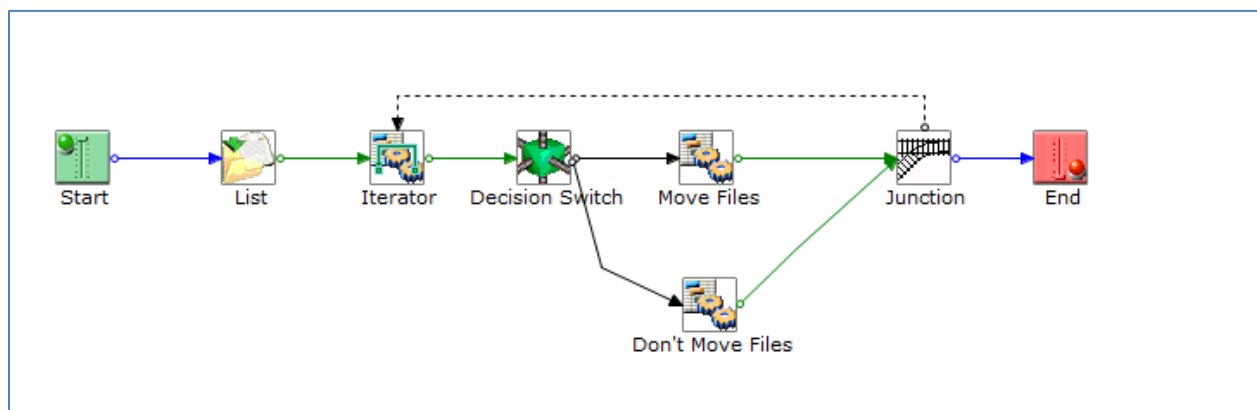
- **Move_Files.iwp** – Sample process flow.

This how-to includes the following topics:

- [Capturing the Content From a Directory](#)
- [Processing Files One At a Time](#)
- [Routing](#)
- [Moving Files](#)

Capturing the Content From a Directory

The following image shows a process flow that includes a *List* object, which is a configured *File Directory Contents Service* (*com.ibi.agents.XDFileDirListAgent*).



In this example, the *XDFileDirListAgent* reads and returns the contents of the *C:\temp\test\in* directory in the following XML document:

```
<?xml version="1.0" encoding="UTF-8" ?>
<dir base="C:\temp\test\in" count="4">
  <item type="file" lastmod="1537187412248" lastmodstr="17 Sep 2018 12:30:12 GMT">fileLLL.zip</item>
  <item type="file" lastmod="1537187412248" lastmodstr="17 Sep 2018 12:30:12 GMT">fileMON.zip</item>
  <item type="file" lastmod="1537187412248" lastmodstr="17 Sep 2018 12:30:12 GMT">fileTUE.zip</item>
  <item type="file" lastmod="1537187412248" lastmodstr="17 Sep 2018 12:30:12 GMT">fileWED.zip</item>
</dir>
```

One of the requirements is to move the following files to a new directory:

- *fileMON.zip*
- *fileTUE.zip*
- *fileWED.zip*

However, the *fileLLL.zip* file must not be moved. In addition, a timestamp must be appended to the name of each file that was moved.

Processing Files One At a Time

The *Iterator* object is an *XML Iterator (XDIterXMLSplit)* that has been configured to split the input document on the item level using the following setting:

```
_xpath(/dir/item)
```

Routing

The *Decision Switch* object uses the following iWay Functional Language (iFL) substring function:

```
_substr(_xpath(/dir/item),4,7)
```

This function will read the *fileMON.zip*, *fileTUE.zip*, and *fileWED.zip* files and set a switch that identifies only those cases where files must be moved. All other cases (files) will be routed through the default line, and not to be moved.

Moving Files

The *Move Files* object contains a configured *File Operations Service (com.ibi.agents.XDPFFileOpsAgent)* with the *Operation* parameter set to *move*.

The value for the *File (from)* parameter is taken from the values provided by the *XDFileDirListAgent*, for example:

```
_concat(_xpath(/dir/@base), '\\', _xpath(/dir/item))
```

This results in the following:

```
C:\temp\test\in\fileMON.zip
```

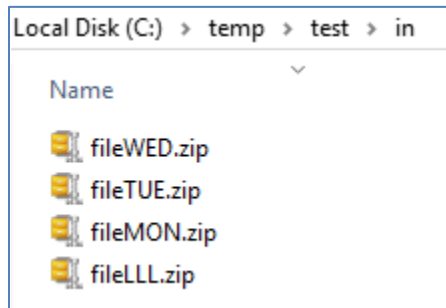
The value for the *File (to)* parameter specifies the location of the destination directory and an iFL function that will append the moved file with a timestamp:

```
_concat('C:\\temp\\test\\out\\', _before(_xpath(/dir/item), '.'), '_*.zip')
```

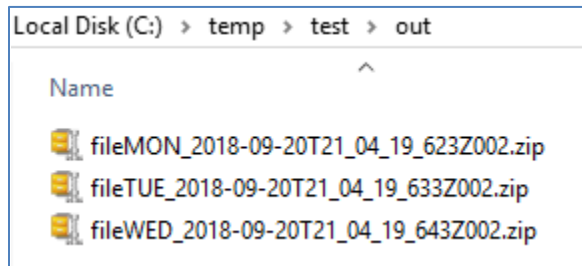
This results in the following:

```
C:\temp\test\out\fileMON_2018-09-20T20_30_37_729Z001.zip
```

For demonstration purpose, the following image shows the contents of the source (input) directory, where all four files are available:



The following image shows the contents of the destination (output) directory after the three required files (*fileMON.zip*, *fileTUE.zip*, and *fileWED.zip*) have been moved and appended with a timestamp.



The following image shows the *fileLLL.zip* file in the source (input) directory, which has not been moved as required.

