FTP Server Realization

The FTP Server functionality is implemented within the E2E Bridge using the Java Services mechanism: The Java based Apache FtpServer as integrated into the bridge as a frontend. The below describes how this import has been done for the presented example. For an easier example to use the Java Frontend please refer to chapter Java Services.

To realize the presented example, some preparations have to be done on Java side and on Bridge side. The full Java source code is included in the example jar files.

On this Page:

Preparations on Java side

1. Implement the Java Interface "BridgeJavaService".

```
public class FtpService implements
BridgeJavaService<FtpServerCallback> {
 private FtpServer server;

 public void initialize(FtpServerCallback ftpServerCallback) {
 ...
 ...
```

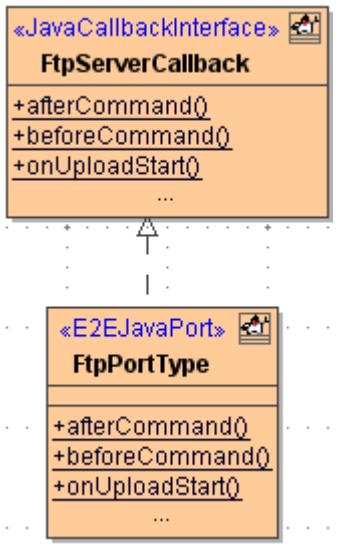
2. Provide an Interface to be Implemented by the Bridge.

```
public interface FtpServerCallback extends BridgeJavaCallback {
 public CallbackReply afterCommand(BridgeUser user, String command,
String argument, CallbackReply reply);

 public CallbackReply beforeCommand(BridgeUser user, String command,
String argument);

 public CallbackReply onUploadStart(BridgeUser user, String
uploadFile);
 ...
 ...
```

3. Prepare JAR-Files.

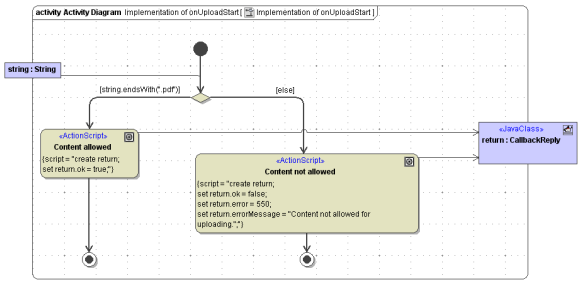
Preparations on Bridge side

1. Import the Prepared JAR File.
2. Define an E2EJavaPort Implementing the Offered Interface.
   Figure: FtpPortType implements the interface defined in Java



The class FtpPortType needs to contain all the operations defined in the interface. For the presented FTP functionality scope, only a few of these operations are actually used. Some of them are implemented by a specifying activity diagram, others by an action script. Those operations that are not needed in the example and thus have no implementation are marked as abstract.

3. Implement Operations by Assigning Activity Diagrams.
   Figure: Implementation of the operation onUploadStart



4. Run Component and Deployment wizards.

After completing the component and deployment wizards, the model is ready to be deployed.