FlexDeploy – ConfigNOW Plugin

This document outlines the final design scope and architectural design of the ConfigNOW plugin for FlexDeploy.

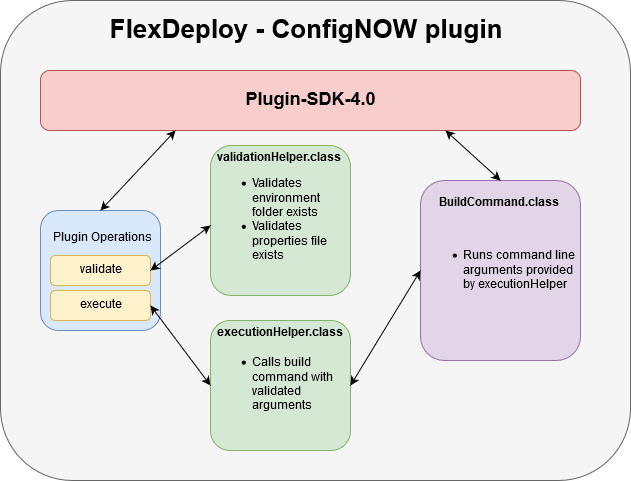
# Scope

The FlexDeploy – ConfigNOW plugin will:

* Have individual operations for a set list (see operation list) of out of the box ConfigNOW commands, which are used through the FlexDeploy UI.
* Include an operation for users to run custom ConfigNOW commands. **(Custom command files will have to be provided by the user)**
* The desired properties file name will be provided by the user, the provided file will include the correctly configured properties for the desired execution endpoint.
* Not allow customisation or modification of any properties files from the FlexDeploy UI. **(All property files should be correctly stored in source control)**
* Any property files should be stored in source control under a folder which is named the same as the FlexDeploy environment code *(ie, folder called DEV if the FlexDeploy environment is called DEV)*
* Expect that the target server has all required infrastructure correctly installed and configured to support the desired ConfigNOW command to be ran.
* Expect that the target server has a compatible version of Jython installed and running.

# Architecture

The plugin utilises the FlexDeploy Plugin-SDK-4.0 to provide logging and execution context, this SDK library is expected to have been externally tested and validated. The plugin is java based and packaged as a JAR file. Each plugin operation has its own class, which is a very simple class, calling validationHelper.class and executionHelper.class, these files then call BuildCommand.class to perform the ConfigNOW command from command line arguments. Each plugin operation has unit tests written for it using Junit to ensure all parameter validation and run execution work as expected when ran individually.



## Build

The build process for the plugin does not utilise any actual ConfigNOW plugin operations, it simply prepares for their use. The build process utilises existing FlexDeploy plugin operations. The build workflow initially pulls ConfigNOW from the users source control (this version of ConfigNOW should include any custom files, and all files should be in the correct file structure). Once the files have been copied from source control, they are then zipped and saved into the FlexDeploy artifacts directory to allow them to be used by the deploy workflow.

## Deploy

The first step of a deployment is to unzip the files from the FlexDeploy artifacts directory across to the plugin which is housed on the deployment target endpoint. The next step is to run the workflow operation command on the deployment endpoint. This will result in the desired ConfigNOW command being executed on the endpoint.

# Usage

Most ConfigNOW operations are supported with their own plugin operation. Some however, are not supported as they return lists to the command line and this is unnecessary for deployments. However, should a user wish to run these commands, they can run them using the custom command operation.

All plugin operations validate that a valid environment folder exists and also validates that the provided properties file exists. If these validations fail, an exception will be thrown and execution of the plugin operation will be terminated.

A list of supported plugin operations are shown in the table below.

|  |  |  |
| --- | --- | --- |
| Operation name | Description | Required Inputs |
| activateComposite | Runs the ConfigNOW command ‘activate\_composite’ from command line. | Properties file name |
| addTomcatUser | Runs the ConfigNOW command ‘add\_tomcat\_user’ from command line. | Properties file name |
| addUser | Runs the ConfigNOW command ‘add\_user’ from command line. | Properties file name |
| configureDatasource | Runs the ConfigNOW command ‘configure\_datasource’ from command line. | Properties file name |
| configureDomain | Runs the ConfigNOW command ‘configure\_domain’ from command line. | Properties file name |
| configureNodeManager | Runs the ConfigNOW command ‘configure\_node\_manager’ from command line. | Properties file name |
| createDomain | Runs the ConfigNOW command ‘create\_domain’ from command line. | Properties file name |
| customCommand | Runs the ConfigNOW command provided in the command file name input. | Properties file name, command file name |
| DbSwitchover | Runs the ConfigNOW command ‘db\_switchover’ from command line. | Properties file name |
| deployApps | Runs the ConfigNOW command ‘deploy\_apps’ from command line. | Properties file name |
| deployComposite | Runs the ConfigNOW command ‘deploy\_composite’ from command line. | Properties file name |
| deployCompositeCp | Runs the ConfigNOW command ‘deploy\_composite\_with\_cp’ from command line. | Properties file name |
| deployFuseApp | Runs the ConfigNOW command ‘deploy\_fuse\_app’ from command line. | Properties file name |
| deployJboss | Runs the ConfigNOW command ‘deploy\_jboss\_app’ from command line. | Properties file name |
| deployOsbConfig | Runs the ConfigNOW command ‘deploy\_osb\_config’ from command line. | Properties file name |
| deployTomcat | Runs the ConfigNOW command ‘deploy\_tomcat\_app’ from command line. | Properties file name |
| opatchApply | Runs the ConfigNOW command ‘opatch\_apply’ from command line. | Properties file name |
| opatchLsInventory | Runs the ConfigNOW command ‘opatch\_lsinventory’ from command line. | Properties file name |
| opatchRollback | Runs the ConfigNOW command ‘opatch\_rollback’ from command line. | Properties file name |
| passwordEncrypter | Runs the ConfigNOW command ‘password\_encrypter’ from command line. | Properties file name |
| reloadTomcat | Runs the ConfigNOW command ‘reload\_tomcat\_app’ from command line. | Properties file name |
| remoteDeployApp | Runs the ConfigNOW command ‘remote\_deploy\_app’ from command line. | Properties file name |
| remoteUndeployApp | Runs the ConfigNOW command ‘remote\_undeploy\_app’ from command line. | Properties file name |
| runCli | Runs the ConfigNOW command ‘run\_cli’ from command line. | Properties file name |
| setDefaultComposite | Runs the ConfigNOW command ‘set\_default\_composite’ from command line. | Properties file name |
| setupOerReports | Runs the ConfigNOW command ‘setup\_oer\_reports’ from command line. | Properties file name |
| shutdownAllServers | Runs the ConfigNOW command ‘shutdown\_all\_servers’ from command line. | Properties file name |
| startAmq | Runs the ConfigNOW command ‘start\_amq’ from command line. | Properties file name |
| startComposite | Runs the ConfigNOW command ‘start\_composite’ from command line. | Properties file name |
| startFuse | Runs the ConfigNOW command ‘start\_fuse’ from command line. | Properties file name |
| startJboss | Runs the ConfigNOW command ‘start\_jboss’ from command line. | Properties file name |
| startKarafSession | Runs the ConfigNOW command ‘start\_karaf\_session’ from command line. | Properties file name |
| startTomcat | Runs the ConfigNOW command ‘start\_tomcat’ from command line. | Properties file name |
| startWlsAdmin | Runs the ConfigNOW command ‘start\_wls\_admin’ from command line. | Properties file name |
| stopAmq | Runs the ConfigNOW command ‘stop\_amq’ from command line. | Properties file name |
| stopComposite | Runs the ConfigNOW command ‘stop\_composite’ from command line. | Properties file name |
| stopFuse | Runs the ConfigNOW command ‘stop\_fuse’ from command line. | Properties file name |
| stopJboss | Runs the ConfigNOW command ‘stop\_jboss’ from command line. | Properties file name |
| stopWlsAdmin | Runs the ConfigNOW command ‘stop\_wls\_admin’ from command line. | Properties file name |
| undeployComposite | Runs the ConfigNOW command ‘undeploy\_composite’ from command line. | Properties file name |
| undeployFuseApp | Runs the ConfigNOW command ‘undeploy\_fuse\_app’ from command line. | Properties file name |
| undeployJboss | Runs the ConfigNOW command ‘undeploy\_boss\_app’ from command line. | Properties file name |
| undeployTomcat | Runs the ConfigNOW command ‘undeploy\_tomcat\_app’ from command line. | Properties file name |