

Network Cloud Family Workflow Process

PLEASE REFER TO R1 NETWORK CLOUD RELEASE DOCUMENTATION

[NC Family Documentation - Release 1](#)

THIS DOCUMENTATION WILL BE ARCHIVED

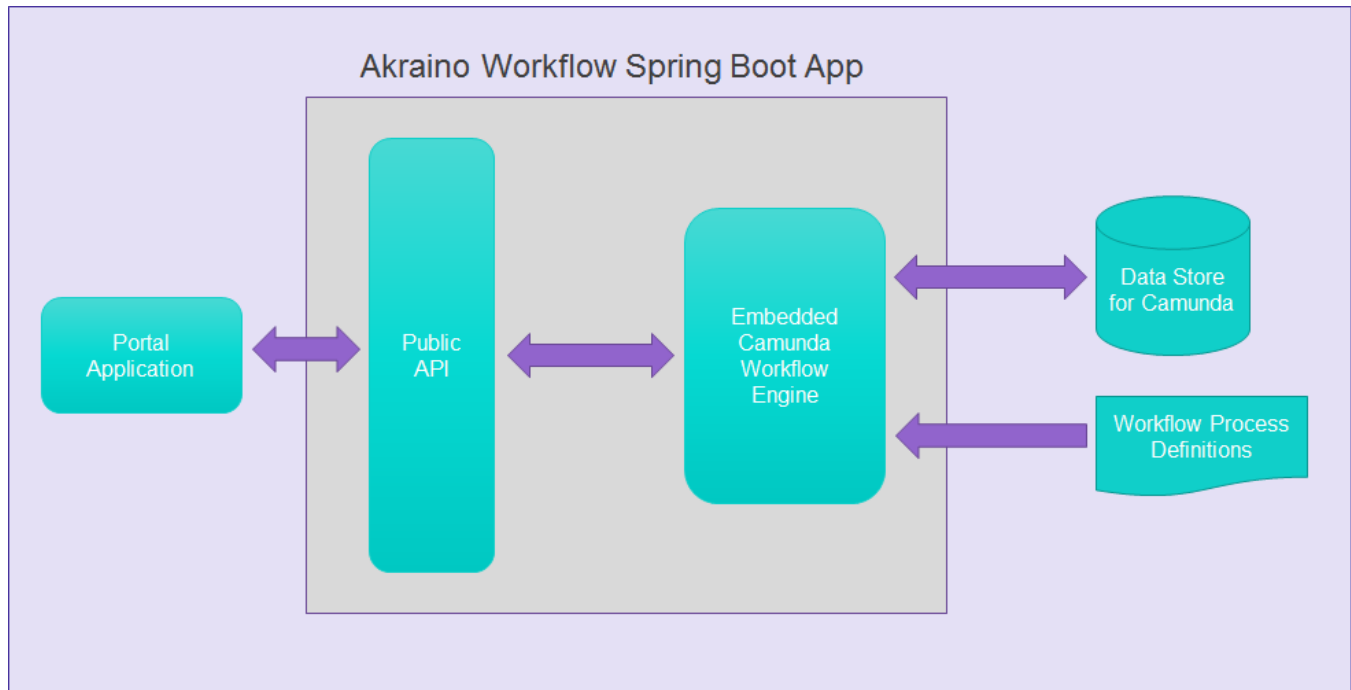
Contents

- [High Level Architecture](#)
- [Portal application](#)
- [Akraio Public API](#)
 - [Build](#)
 - [Deploy](#)
 - [Tempest](#)
 - [Apache](#)
 - [ONAP](#)
 - [Airship](#)
- [Camunda BPM Engine](#)
- [Business process workflows](#)
 - [Build Business process workflow](#)
 - [Deploy Business process workflow](#)
 - [Tempest Business process workflow](#)
 - [Apache Business process workflow](#)
 - [ONAP Business process workflow](#)
 - [Airship Business process workflow](#)

Akraio Edge Stack Portal and Workflow engine us hides the complexity of the install and management tools of the large number of edge sites. Akraio Edge Stack portal could be agnostic to specific blueprint and support multiple blueprints from the community.

High Level Architecture

Akraio workflow process is designed by using spring boot and embedded Camunda workflow engine. The workflow process and Spring boot application provide external rest API. Each external API is responsible for invoking respective Camunda workflow process. Camunda reads the process definitions from the document store or the classpath.



Portal application

The portal application is responsible to initiate the build and deployment process. It is responsible for consuming the public API provided by Akraino spring boot application. It is also responsible for updating the appropriate response, which is provided by the Akraino API.

Akraino Public API

Akraino public API defines the rest API for each portal process. The portal application uses this API to execute various process like build, deploy etc.

Akraino Seed code supports the following APIs

- Build
- Deploy
- Apache
- ONAP
- Airship

Below section provides additional details about the workflow used within the seed code.

Build

This API is used to build and verify the tar file.

Building the tar file will be done by using the shell script, and the same script is executed as part of the workflow process. The rest endpoint for this API is /build/.

The input parameters for this endpoint is as below.

```
{
  "filepath": "string",
  "fileparams": "string",
  "targetfolder": "string"
}
```



- "filepath" path parameter is used to pass the script filename including the file path, which is used to create the tar file.
- "fileparams" parameter is used to pass the parameters to the script file.
- "targetfolder" is the folder where generated YAML files will be stored, and it is used by the java process to validate the generated YAML files.

The output of the build API is given below

```
{
  "message": "string",
  "status": 0
}
```



Status Code 200 = Build Completed

Status Code other than 200 = Build API Failed

For each status, an appropriate message is returned.

Deploy

The rest endpoint for this API is /deploy/

The input parameters for this endpoint is below.

```
{
  "sitename": "string",
  "filepath": "string",
  "fileparams": "string",
  "winscpfilepath": "string",
  "winscpfileparams": "string",
  "remotserver": "string",
  "port": 0,
  "username": "string",
  "password": "string",
  "destdir": "string",
  "remotefilename": "string",
  "remotefileparams": "string",
  "deploymentverifier": "string",
  "deploymentverifierfileparams": "string",
  "noofiterations": 0,
  "waittime": 0,
  "postverificationscript": "string",
  "postverificationScriptparams": "string"
}
```



- "filepath" parameter is used to execute the local shell script.
- "fileparams" is used to pass the parameters to the local shell script
- "winscpfilepath" is used to move the files from the local to the remote machine.
- "winscpfileparams" is used to pass the parameters to the SCP shell script.
- "remotserver", "username", "password", and "port" is used to connect to the remote machine to execute the script.
- "destdir" is the directory where the remote script file is located.
- "remotefilename" is the shell script file which needs to be executed in the remote machine.
- "remotefileparams" is used to pass the parameters to the remote script file.
- "deploymentverifier" is a script file, which is used to verify the deployment process.
- "deploymentverifierfileparams" is the input parameter to the deployment verification script.
- "noofiterations" is used to execute the deployment verification script multiple times within specified intervals.
- "waittime" is used to specify the deployment verification script intervals.
- "postverificationscript" is used to pass the post verification script filename.
- "postverificationScriptparams" are used to pass the parameters to the post verification script.

The output of the deployed API is given below

```
{
  "message": "string",
  "status": 0
}
```



Status Code 200 = Deploy Completed

Status Code other than 200 = Deploy API Failed

For each status, an appropriate message is returned. Deployed API will send intermediate status by consuming Portal status update API.

The intermediate status response is given below.

```
{
  "siteName": "string",
  "buildStatus": "string",
  "createTarStatus": "string",
  "genesisNodeStatus": "string",
  "deployToolsStatus": "string",
  "deployStatus": "string",
}
```

After completion of the respective status, API will start sending response to the Portal API, which is used to display the intermediate deploy status in Portal.

Tempest

The rest endpoint for this API is /tempest/

The input parameters for Tempest is given below.

```
{
  "sitename": "string",
  "filename": "string",
  "fileparams": "string",
  "filetrasferscript": "string",
  "filetransferparams": "string",
  "remotserver": "string",
  "port": 0,
  "username": "string",
  "password": "string",
  "destdir": "string",
  "deploymentverifier": "string",
  "verifierparams": "string",
  "noofiterations": 0,
  "waittime": 0,
}
```



- "filename" parameter is used to execute the local shell script
- "fileparams" is used to pass the parameters to the local shell script
- "filetrasferscript" is used to pass the script file which is used to move the files from local to remote machine.
- "filetransferparams" is used to pass the parameters to the SCP shell script.
- "remotserver", "username", "password", and "port" is used to connect to the remote machine to execute the script.
- "destdir" is the directory where the remote script file is located.
- "remotefilename" is the shell script file which needs to be executed in the remote machine.
- "remotefileparams" is used to pass the parameters to the remote script file.
- "deploymentverifier" is a script file, which is used to verify the deployment process.
- "deploymentverifierfileparams" is the input parameter to the deployment verification script.
- "noofiterations" is used to execute the deployment verification script multiple times within specified intervals.
- "waittime" is used to specify the deployment verification script intervals.

Apache

The rest endpoint for this API is /apache/

The input parameters for Apache is given below.

```
{
  "sitename": "string"
  "filename": "string",
  "fileparams": "string",
  "filetrasferscript": "string"
  "filetransferparams": "string",
  "remotserver": "string",
  "port": 0,
  "username": "string",
  "password": "string",
  "destdir": "string",
  "deploymentverifier": "string",
  "verifierparams": "string",
  "noofiterations": 0,
  "waittime": 0,
}
```



- "filename" parameter is used to execute the local shell script
- "fileparams" is used to pass the parameters to the local shell script
- "filetrasferscript" is used to pass the script file which is used to move the files from local to remote machine.
- "filetransferparams" is used to pass the parameters to the SCP shell script.
- "remotserver", "username", "password", and "port" is used to connect to the remote machine to execute the script.
- "destdir" is the directory where the remote script file is located.
- "deploymentverifier" is a script file, which is used to verify the deployment process.
- "noofiterations" is used to execute the deployment verification script multiple times within specified intervals.
- "waittime" is used to specify the deployment verification script intervals.

ONAP

The rest endpoint for this API is /onap/

The input parameters for ONAP is given below.

```
{
  "sitename": "string"
  "filename": "string",
  "fileparams": "string",
  "filetrasferscript": "string"
  "filetransferparams": "string",
  "remotserver": "string",
  "port": 0,
  "username": "string",
  "password": "string",
  "destdir": "string",
  "deploymentverifier": "string",
  "verifierparams": "string",
  "noofiterations": 0,
  "waittime": 0,
}
```



- "sitename" parameter is used to pass the site name.
- "filepath" parameter is used to execute the local shell script.
- "fileparams" is used to pass the parameters to the local shell script.
- "filetrasferscript" is used to pass the script file which is used to move the files from local to remote machine.
- "filetransferparams" is used to pass the parameters to the SCP shell script.
- "remotserver", "username", "password", and "port" is used to connect to the remote machine to execute the script.
- "destdir" is the directory where the remote script file is located.
- "deploymentverifier" is a script file, which is used to verify the deployment process.
- "deploymentverifier" is a script file, which is used to verify the deployment process.
- "noofiterations" is used to execute the deployment verification script multiple times within specified intervals.
- "waittime" is used to specify the deployment verification script intervals.

Airship

The rest endpoint for this API is /airship/

The input parameters for airship is given below.

```
{
  "sitename": "string",
  "filepath": "string",
  "fileparams": "string",
  "winscpdir": "string",
  "winscpfilepath": "string",
  "winscpfileparams": "string",
  "remotserver": "string",
  "port": 0,
  "username": "string",
  "password": "string",
  "destdir": "string",
  "remotefilename": "string",
  "remotefileparams": "string",
}
```



- "sitename" parameter is used to pass the site name.
- "filepath" parameter is used to execute the local shell script.
- "fileparams" is used to pass the parameters to the local shell script.
- "winscpdir" is used to pass the win SCP script file directory.
- "winscpfilepath" is used to pass the script file which is used to move the files from local to the remote machine.
- "winscpfileparams" is used to pass the parameters to the SCP shell script .
- "remotserver", "username", "password", and "port" is used to connect to the remote machine to execute the script.
- "destdir" is the directory where the remote script file is located.
- "remotefilename" is the remote shell script file where we need to execute in remote machine.
- "remotefileparams" is used to pass the parameters to the remote script file.

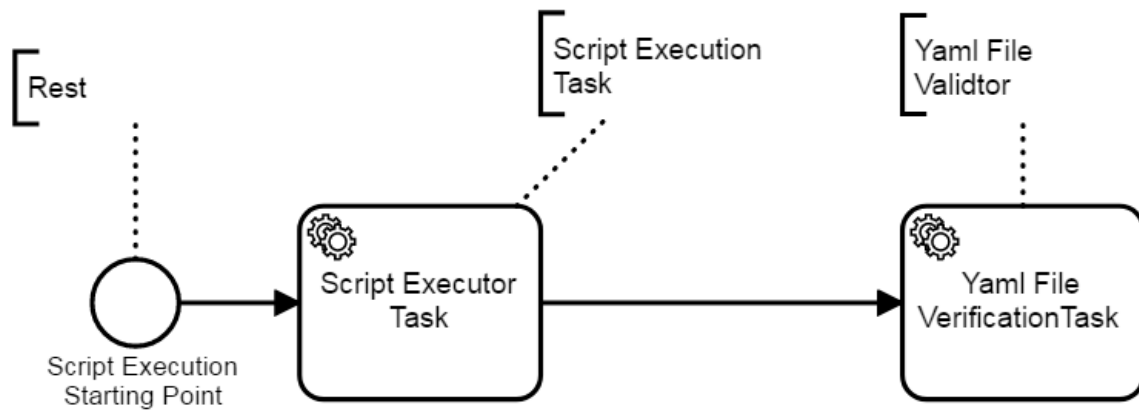
Camunda BPM Engine

Camunda is a lightweight Business process engine, which will take the business process definitions and execute the business flow. Akraino workflow process has different rest API and each rest API is backed by respective business process workflow. Camunda Business process engine is embedded into the spring boot application, and when we start the spring boot application, Camunda business engine will read the process workflows from the class-path.

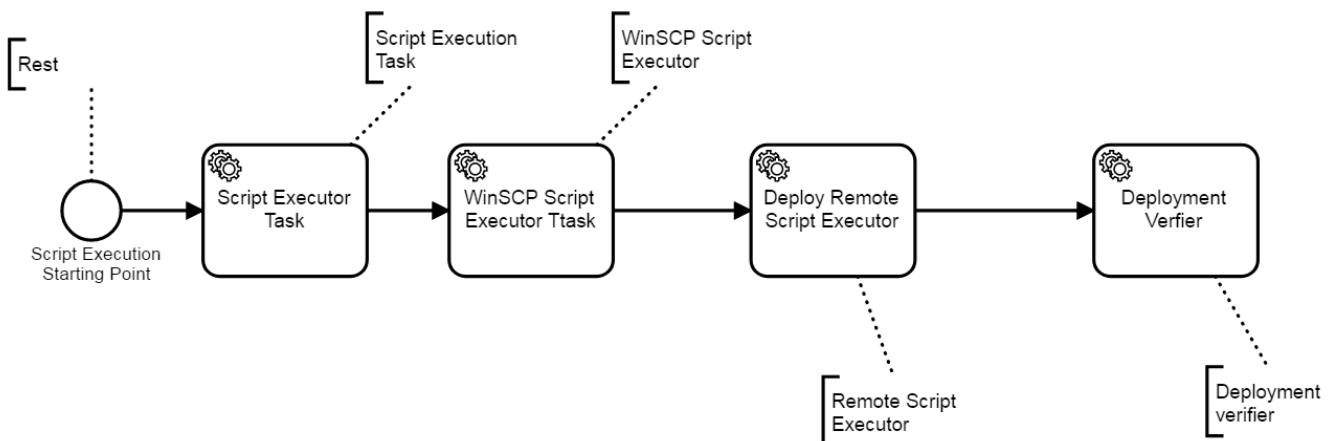
Business process workflows

All akraino rest APIs are backed by Camunda business process engine. Each rest API has its own business process. The business process workflows are given below.

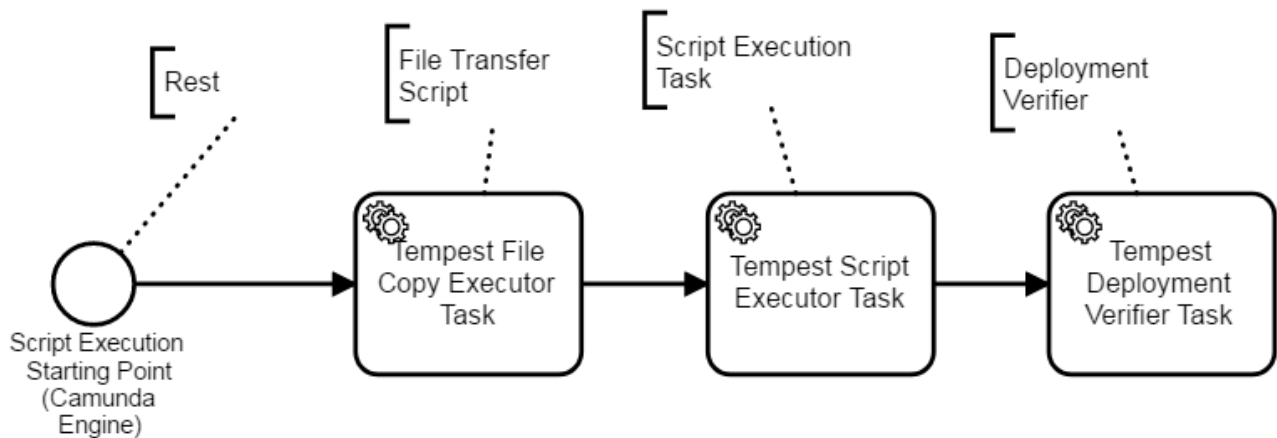
Build Business process workflow



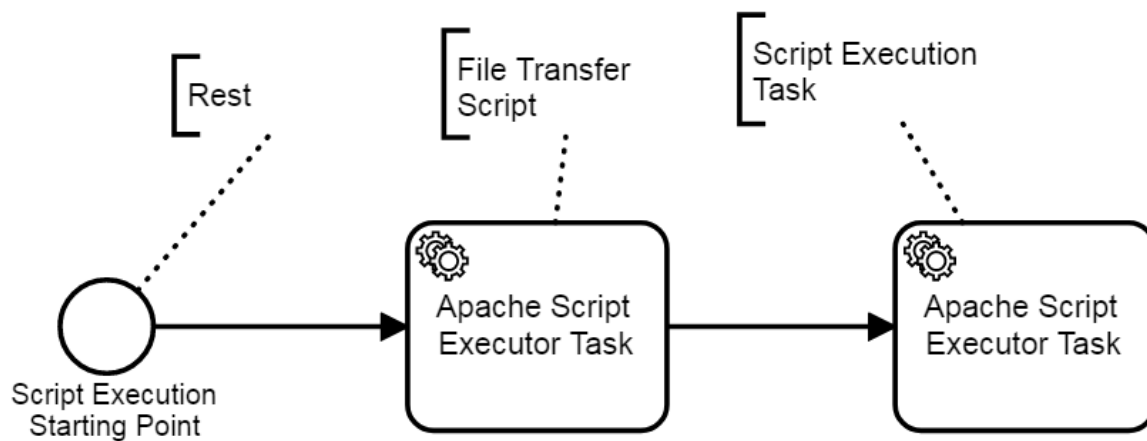
Deploy Business process workflow



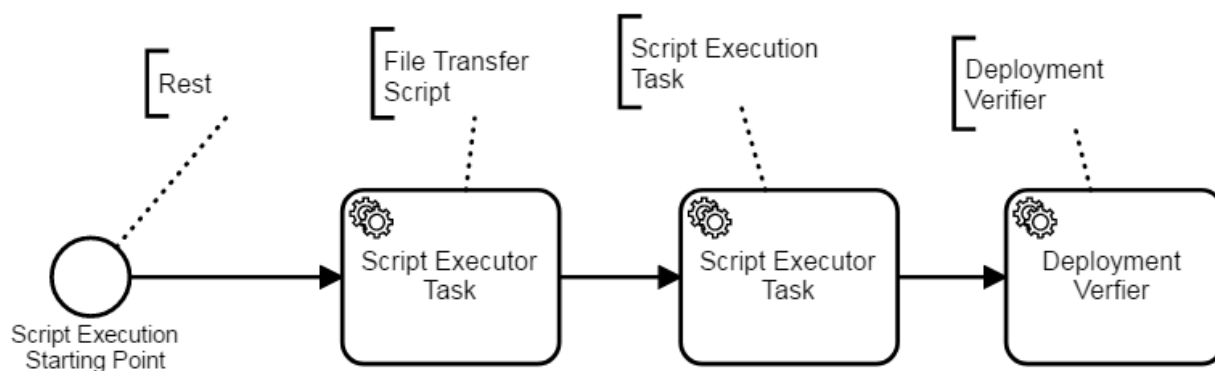
Tempest Business process workflow



Apache Business process workflow



ONAP Business process workflow



Airship Business process workflow

