

Tools - Path-Based Module Generator

What's new?

Please see [Path-Based Module Generator - Release Notes](#) for more information.

Purpose

The purpose of the Path-Based Module Generator is to create explicit-content modules based on the path of the source files containing the objects.

The tool scans the sub-folders of a provided root folder, that match a provided regexp filter, if any.

For each sub-folder, it checks that a corresponding module (named from the sub-folder name, prefixed with a given prefix) is already created; otherwise, it creates one based on an explicit list selecting all the objects whose absolute path starts with the sub-folder path.

When required, it cleans up the modules whose name is also prefixed with the given prefix, but no more matching with any sub-folder found in the root folder.

The Path-Based Module Generator is made of a Groovy script and a batch to launch it

- CreateModulesFromPath.cgroovy
- CreateModulesFromPath.bat

This tool is provided by Maintenance Team, and published as an extension **com.castsoftware.labs.tools.pathbasedmodulegenerator**

Applicable in CAST Version

Release	Yes/No
8.3.x	✓

Applicable RDBMS

RDBMS	Yes/No
CSS3	✓
CSS2	✓

Prerequisites

Follow the below prerequisites before running the tool:



CAST-AIP

CAST-AIP must be installed on the server where you launch the scripts.



Deployment Folder

The deployment folder should be accessible from the server where you launch the scripts. Based on this, it is recommended to run the scripts on the analysis server



BACKUPS

Take a backup of the triplet before proceeding

Configuration

The batch file must be configured with the following mandatory variables : FLATDIR, LOGDIR, ROOTDIR, CONNECTION_PROFILE, APP_NAME

There are optional variables : MODULE_PREFIX, PATH_FILTER, CLEANUP

Configuration - CreateModulesFromPath.bat

```
REM Mandatory
set "FLATDIR=<CAST AIP Installation PATH>"
set "LOGDIR=<Path to the log folder>"
set "ROOTDIR=<Modules Root Path>"
set "CONNECTION_PROFILE=<Connection Profile>"
set "APP_NAME=<Application Name>"
```

```
REM Optional
set MODULE_PREFIX=<Module Prefix>
set "PATH_FILTER=<Path Filter>"
set CLEANUP=<YES or NO>
```

FLATDIR must be set to the CAST Installation path
LOGDIR must be set to the folder where the log file should be generated
ROOTDIR must be set to the root folder (folder whose all subfolders will be scanned)
CONNECTION_PROFILE must be set to the connection profile to the application
APP_NAME must be set to the application name

MODULE_PREFIX is a string added at the beginning of subfolder names in order to build module names
PATH_FILTER is a regular expression to match for a subfolder name to generate a module
CLEANUP : if true, all existing modules that begin with the MODULE_PREFIX and do not match a subfolder name will be deleted

NB:

- *ROOTDIR should be the deployment folder or one of its sub-folders*
- *If MODULE_PREFIX is not defined, then CLEANUP option is automatically disabled. This is to avoid accidentally removing modules that were not generated by this tool*
- *If CLEANUP option leads to delete all user-defined modules (typically when user sets ROOTDIR to an empty folder), then a Full-Content Module is automatically generated and full-content mode is activated.*

Generate Path-Based Explicit-Content Modules

Usage

Launch batch CreateModulesFromPath.bat

User can now check the creation of the modules in the log file :

Log File

```
Running action: Executes a groovy script
Automatic Path Modules Creation:
  Application: Source Server / MyApp 14038
  Root folder: S:\Sources\MyApp\Analyzed
MOD
  Folder name filter: myapp.*
  Clean up obsolete modules
Checking if a module exists for folder myapp-jb
Creating a user-defined module for folder myapp-jb
Path Module Added: MOD_myapp-jb
Checking if a module exists for folder myappAnnotation-jb
Creating a user-defined module for folder myappAnnotation-jb
Path Module Added: MOD_myappAnnotation-jb
Checking if a module exists for folder myappAnnotationEar
Creating a user-defined module for folder myappAnnotationEar
Path Module Added: MOD_myappAnnotationEar
Checking if a module exists for folder myappAnnotationWar
Creating a user-defined module for folder myappAnnotationWar
Path Module Added: MOD_myappAnnotationWar
Checking if a module exists for folder myappAuth
Creating a user-defined module for folder myappAuth
Path Module Added: MOD_myappAuth
Checking if a module exists for folder myappBusinessLogic
Creating a user-defined module for folder myappBusinessLogic
Path Module Added: MOD_myappBusinessLogic
Removed full-content module
```

User can also check that the new modules were properly generated by opening CAST-MS on the connection profile, clicking the application and going to Modules tab

Related Pages

Notes