

# Tools - Classpath Checker

## Purpose

The purpose of this tool is to help spot missing or wrongly located Jar files in a J2EE analysis, by finding inconsistencies between classpath configuration in Cast-MS and Jar files folders present on the disk.

It fetches from Management Base the list of folders declared as Archive Folders and the list of files declared as Archive Files, then checks if they exist on the disk, and checks if the folders contain some Jar files.

## Applicable in CAST Version

Release	Yes/No
8.3.x	✓
8.2.x	✓
8.1.x	✓
8.0.x	✓
7.3.x	✓

## Applicable RDBMS

RDBMS	Yes/No
Oracle Server	✓
Microsoft SQL Server	✓
CSS2	✓

## Usage

**classpath\_checker.py** is a Python script, so you need to launch it with Python 3.4 embedded in CAST AIP.

Mandatory arguments :

- **host** : the hostname of the CSS server where the management base is installed
- **mb** : the name of the management base

Option arguments :

- **port** : the port of the CSS server where the management base is installed (default : 2280)
- **user** : the user name used to connect to the CSS server (default : operator)
- **password** : the password used to connect to the CSS server (default : CastAIP)

Open a CMD window, and navigate to the folder where you installed classpath\_checker. Type the following command-line :

### Command-line

```
<CAST_INSTALL_PATH>\ThirdParty\Python34\python.exe classpath_checker.py -host <CSS_SERVER> -mb <MNGT_BASE>
```

## Output

```
Archive file S:\Sources\APP\Analyzed\component\libs\util.jar does not exist !
Archive folder S:\Sources\APP\Analyzed\component\third-party\libs does not exist !
Archive folder S:\Sources\APP\Analyzed\component\third-party\jars is empty !
Archive folder S:\Sources\APP\Analyzed\component\libs contains no JAR !
```

### Interpretation of the output

The first message means that file util.jar is not present at the expected path. Look for it in the deploy folder and if you find it, modify the classpath entry accordingly. If you do not find it, then ask development team for the file.

The second message means that folder libs does not exist or is not at the expected path. Look for it in the deploy folder and if you find it, modify the classpath entry accordingly. If you do not find it, see if you did not forget to extract some archive files that could contain it. Else, ask development team for the folder

The third message means that folder jars is empty. Ask development team what the folder is supposed to contain.

The fourth message means that folder libs contains files, but not JAR files. Ask development team what the folder is supposed to contain.

### Notes/comments

The tool is available here :

[svn://gargantua/SUPPORT/trunk/Tools/PYTHON/classpath\\_checker](svn://gargantua/SUPPORT/trunk/Tools/PYTHON/classpath_checker)

### Related Pages

[J2EE - application qualification specifics](#)