

PETCT Registration Workflow Java CLI Client

1) **Java archive name:**

PETCT_Registration_CLI.jar

2) **Required libraries:**

- Location:
 <InstallationFolder>/lib subdirectory
- List of required libraries (all are included into distribution):
FastInfoSet.jar
activation.jar
http.jar
jaxb-api.jar
jaxb-impl.jar
jaxb-xjc.jar
jaxws-api.jar
jaxws-rt.jar
jaxws-tools.jar
jsr173_api.jar
jsr181-api.jar
jsr250-api.jar
mimepull.jar
saaj-api.jar
saaj-impl.jar
sjsxp.jar
stax-ex.jar
streambuffer.jar

3) **Getting execution help**

- Run application without arguments
- Example:

```
/usr/bin/java -jar <InstallationFolder>/PETCT_Registration_CLI.jar
```

- Output of execution without arguments:

Usage for REGISTRATION:

```
java -jar PETCT_Registration_ClientA.jar -slicerLocation /Applications/Slicer3-3.5-alpha-2010-05-02-darwin-x86 -resultFolder /cniNew/SPINE/Projects/PETCT/Run4P2_WS -timePoint 3 -petVolume /cniNew/SPINE/Projects/PETCT/Run4P2_WS/PET_st3-Subvolume-resample_scale-1.0.nrrd -ctVolume /cniNew/SPINE/Projects/PETCT/Run4P2_WS/CT_st3-Subvolume-resample_scale-1.0.nrrd -timePointRef 1 -ctVolumeRef /cniNew/SPINE/Projects/PETCT/Run4P2_WS/CT_st1-Subvolume-resample_scale-1.0.nrrd -roiVolume /projectFolder/ROI.nrrd
```

Usage for DICOM Conversion:

```
java -jar PETCT_Registration_ClientA.jar -useDicomConvert -slicerLocation /Applications/Slicer3-
```

```
3.5-alpha-2010-05-02-darwin-x86 -resultFolder /cniNew/SPINE/Projects/PETCT/Run4P2_WS
-timePoint 3 -petDicomFolder /cniNew/SPINE/Projects/PETCT/pa1/st1/se1/0001 -ctDicomFolder
/cniNew/SPINE/Projects/PETCT/pa1/st1/se1/0001
```

Usage for BLANK Label Map Creation for a given volume:

```
java -jar PETCT_Registration_ClientA.jar -blankLabelMap -slicerLocation /Applications/Slicer3-3.5-
alpha-2010-05-02-darwin-x86 -inputVolume /cniNew/SPINE/Projects/A/PET_st1-Cropped.nrrd
-outputLabelMap /cniNew/SPINE/Projects/A/ROI-label.nrrd
```

4) Good Programming Practice tip:

- When calling from another application, use whole paths for PETCT_Registration_CLI.jar and input file locations.
- Use system path to locate “java”/”java.exe”. To do so, before calling PETCT_Registration_CLI.jar from another application, fetch \$PATH as environmental variable in order to avoid hardcoding java executable location. Use hardcoded /usr/bin/java or similar notation only for initial development

See Java code example for calling PETCT_Registration_CLI.jar below

Comparison Tip: Use the same calling approach as for XNATRestClient.jar

5) Command Line Example for running DICOM conversion:

WORKFLOW CLI Call for DICOM convert:

```
java -jar <InstallationFolder>/PETCT_Registration_CLI.jar -useDicomConvert -slicerLocation
/Applications/Slicer3-3.7-alpha-2010-06-13-darwin-x86 -resultFolder
/SharedFiles/From8Core/PETCT/Results/A -timePoint 3 -petDicomFolder
/SharedFiles/From8Core/PETCT/DICOM/A/pa1/st1/se2 -ctDicomFolder
/SharedFiles/From8Core/PETCT/DICOM/A/pa1/st1/se1
```

6) Command Line Example for creation of blank label map

NOTE: USE FOR the TIME POINT 1

```
java -jar <InstallationFolder>/PETCT_Registration_CLI.jar -blankLabelMap -slicerLocation
/Applications/Slicer3-3.7-alpha-2010-06-13-darwin-x86 -inputVolume
/SharedFiles/From8Core/PETCT/Results/A/PET_st1-Subvolume-resample_scale-1.0.nrrd
-outputLabelMap /SharedFiles/From8Core/PETCT/Results/A/ROI-blank.nrrd
```

7) Command Line Example for running Registration pipeline:

WORKFLOW CLI Call for Registration:

```
java -jar <InstallationFolder>/PETCT_Registration_CLI.jar -slicerLocation /Applications/Slicer3-3.7-
alpha-2010-06-13-darwin-x86 -resultFolder /SharedFiles/From8Core/PETCT/Results/A -timePoint 3 -
petVolume /SharedFiles/From8Core/PETCT/Results/A/PET_st3-Subvolume-resample_scale-1.0.nrrd -
ctVolume /SharedFiles/From8Core/PETCT/Results/A/CT_st3-Subvolume-resample_scale-1.0.nrrd
-timePointRef 2 -ctVolumeRef /SharedFiles/From8Core/PETCT/Results/A/CT_st2-Subvolume-
resample_scale-1.0.nrrd -roiVolume /SharedFiles/From8Core/PETCT/Results/A/ROI-blank.nrrd
```

8) Java code example for calling PETCT_Registration_CLI.jar

```

// env vars to interact with Slicer and Java Applications (Linux/MacOS version)
String[] _envSlicerEnv = {"PATH=" + System.getenv("PATH"),
    "DISPLAY=" + System.getenv("DISPLAY"),
    "HOME=/tmp",
    "LD_LIBRARY_PATH=" + (null == System.getenv("LD_LIBRARY_PATH") ?
"LD_LIBRARY_PATH=0" : System.getenv("LD_LIBRARY_PATH"))
    };

.....
// command line for creation blank label map for a given PET volume
String commandLine = "java -jar " + _distributionFolder + File.separator + "
PETCT_Registration_CLI.jar" +
    "-blankLabelMap" +
    "-slicerLocation " + _configSlicerLocation +
    "-inputVolume " + imageVolumeIn +
    "-outputLabelMap " + labelVolumeOut;

// remove double blank spaces (Java specific for system calls)
.....
// Call CLI at last
Process p = Runtime.getRuntime().exec(commandLine, _envSlicerEnv);

// wait for process completion for blank label map creation
.....

```