

Subject: jediTV and edrCanner User's Guide, D42.0.1 (v1)
Authors: Cecilia Cheng
Developer: Zhang Fan Xing, Cecilia Cheng
Date: October 17, 2012

1. Background

JEDI stands for Java EDR (Experimental Data Record) Display Interface. It is an essential EDR quick-view tool that has made important contributions to many past and current NASA/JPL space missions, such as Cassini, MER, MSL and Phoenix. It takes an EDR and a configuration file called personality and create displays that show the image and labels. Since the majority of the time for displaying the EDR lies in the creation of the image, performance can be increased if these images are "canned".

2. edrCanner

The EDR Canner takes anEDR and multiple personality files as inputs, and generate jpegs and json text files as output files. These "canned" objects are stored until deleted by the user.

3. jediTV

The jediTV page is served out from the tomcat server. The html page will use the "canned" objects and render a display that is shown in the browser.

4. Configuration

- Copy the delivered jediTV template directory in your local space:
`% cp -R $V2TOP/webapps/jediTV .../path/to/directory/jediTV`
- Copy the correct libraries to jediTV. This step will be automated in the next delivery:
`% mkdir .../path/to/directory/jediTV/lib`

Copy these libraries files form \$V2HTML/lib/x86-linux
libclib_jiio.so
libmlib_jai.so

Copy these jar files form \$V2HTML/jars
clibwrapper_jiio.jar
edr-canner-1.0.0.jar
jai_codec.jar
jai_core.jar
jai_imageio.jar

```
jopt-simple-4.3.jar
json_simple-1.1.jar
jujube.jar
mlibwrapper_jai.jar
slf4j-api-1.6.1.jar
slf4j-jdk14-1.6.1.jar
treevotee-0.9.3p3ee.jar
vicario.jar
xercesImpl.jar
```

- Create a softlink in the tomcat directory and point to your jediTV directory

Note: Have the SA modify tomcat's context.xml to include:
`<Context crossContext="true" allowLinking="true">`

5. Run the EDR Canner

- Modify `.../path/to/directory/jediTV/bin/runCanner.sh` and specify the source EDR directory, the personality files to be used and the lib directory created in Step 4.1
- The script processes 1 EDR at a time. You may choose to write a wrapper script to select which EDRs to can, or have the pipeline create the canned objects. To do so, simply add the following line to the end of the JediNewEdr script. There's no changes to MATIS or the process definition files:

```
.../path/to/directory/jediTV/bin/runCanner.sh <name/of/EDR>
```

6. Create jediTV channels

- For each channel that jediTV will provide, create a corresponding directory in the jediTV/channels directory:
`% cp -R NewChannel name/of/new/channel`
- The default refresh rate for each channel is 10 seconds. To change this, modify the watch.html file in the channel directory:
`<META http-equiv="refresh" content="10">`
- The default jediTV/MSL.html page currently points to 2 default channels: Playback and Realtime. This file needs to be modified to point to the new channels that have been created.

7. Feeding jediTV

- The canned objects need to get to the channels directory so that they can be displayed in the browser.
- Two sample scripts have been delivered:
 - jediTV/bin/getRealtime.sh
 - jediTV/bin/getPlayback.sh
- Modify these sample scripts to point to the directory that has the canned objects. For simple maintenance, it is recommended that one script matches to one channel.
- Running these scripts in the background will continue to feed jediTV

8. Notes

- Canned objects need to be cleaned up by a process defined by the project. jediTV does not provide any cleanup of these files.
- The edrCanner can also take previously generated jpegs as inputs. This feature, however, has not been tested in D42.0.1.