

## Reprocessing Guide for WES-2 Bridge.

Prior Work: See Guide for Trimming AWIPS-2 Cases for WES-2 Bridge. Trimming cases prior to reprocessing will save lots of time later.

**Hint:** You can save time by reprocessing only the particular data types you need rather than the whole case (this depends on the reason to reprocess; sometimes you have no alternative but to reprocess the entire case).

Other important background information: The document called “Complete Port Assignments for EDEX Instances” contains the port numbers required for a given instance. We will almost always use EDEX\_00 for reprocessing, and the rawPlay software also uses the default port numbers for EDEX\_00 for your convenience. Even so, the port numbers relevant to reprocessing data for EDEX\_00 are repeated here for your convenience:

- edexHttpPort : 9581 (“E”)
- jmsPort : 5672 (“P”)
- qpidHttpPort : 8180 (“Q”)
- qpidJmxPort : 8999 (“J”)

### Programs and their locations:

- rawPlay: /w2b/utilities/rawPlay4.py
- qpidMonitor: /w2b/utilities/qpidmonitor/qpidmonitor.py
- jconsole (packaged with Java as part of any system)

### Process for Reprocessing:

1. Use WES-2 Bridge to start up the EDEX\_00 instance.

```
cd /w2b/wes
./wes.sh
```

2. Verify the EDEX\_00 instance is running (if this is the very first time for running this instance, it could take a long time for the EDEX database to completely initialize.

To check out EDEX status:

```
cd into EDEX logs location
    cd /usr/local/edex-environment/EDEX_00/edex/logs or
```

tail appropriate logs

```
tail -f edex*YYYYMMDD*.log
tail -f edex-ingest-YYYYMMDD.log edex-ingestGrib-YYYYMMDD.log
(and other datatype-specific logs)
```

normal output: routes starting up and a message that says EDEX IS OPERATIONAL

bad messages include “no connection to underlying database” or EDEX continually restarting

3. When EDEX is stable, run the rawPlay program. You need to give the appropriate switches:

- r (root) = directory where data are located. This should be a directory that contains a lot of data type directories (it should look like /data\_store on the operational AWIPS)
- x (edex) = switch that says actually send data to EDEX for processing
- d (data) = lists of datatypes to process (e.g., radar, grib, grib2, binlightning, sat) (-A feeds all datatypes rather than specifying each one individually).
- s (start) = starttime of when to start the feeding process (YYYYMMDD\_HHMM)
- e (end) = endtime of when to stop feeding (YYYYMMDD\_HHMM)
- v (verbose) = give verbose messages
- l (log) = make a log file in the directory you're running rawplay from.
- p (regular qpid port) = specify the JMS port noted above as “P”
- q (qpid http port) = specify the QPID http port noted above as “Q”

4. Monitor the process using qpidmonitor. It takes two switches:

- s for the **system** (host) you're connecting to. This could be localhost, wes2-xxx,
- p for the port to connect to. This is the same as the qpid http port (“Q”) above. The default value is 8180, so it works without specifying this parameter using EDEX\_00 instance.

The filter line in the qpidmonitor GUI takes a regular expression.  
 (Hint: use “ingest,external” or “ingest,external,generate”)

Queue Name	Tot Enqueue	Tot Dequeue	Msg Depth	Enqueue Rate	Dequeue Rate
external.dropbox	12648749	12648748	1	0.02	0.02
Ingest.acars	61477	61477	0	0.00	0.00
Ingest.airep	66360	66360	0	0.00	0.00
Ingest.airmet	1252	1252	0	0.00	0.00
Ingest.atcf	0	0	0	0.00	0.00
Ingest.binlightning	9570	9570	0	0.00	0.00
Ingest.bufrascap	4634	4634	0	0.00	0.00
Ingest.bufrhdw	44721	44721	0	0.00	0.00
Ingest.bufrmos	1935	1935	0	0.00	0.00
Ingest.bufrmthdw	3578	3578	0	0.00	0.00
Ingest.bufrncwf	1939	1939	0	0.00	0.00
Ingest.bufrrobs	0	0	0	0.00	0.00
Ingest.bufrquiskcat	0	0	0	0.00	0.00
Ingest.bufrsigwx	616	616	0	0.00	0.00
Ingest.bufrssmi	5432	5432	0	0.00	0.00
Ingest.bufrua	5535	5535	0	0.00	0.00
Ingest.ccfp	243	243	0	0.00	0.00
Ingest.convsigmat	515	515	0	0.00	0.00
Ingest.cwa	540	540	0	0.00	0.00
Ingest.dhr	649423	649420	3	0.00	0.00
Ingest.dpa	248780	248780	0	0.00	0.00
Ingest.ffg	2756	2756	0	0.00	0.00
Ingest.GeoMag	0	0	0	0.00	0.00
Ingest.goessounding	271985	271985	0	0.00	0.00
Ingest.GribDecode	5525690	5525688	2	0.01	0.01
Ingest.GribSplit	5562595	5562585	10	0.01	0.01
Ingest.handleoup	0	0	0	0.00	0.00
Ingest.idft	112	112	0	0.00	0.00
Ingest.intsigmet	116	116	0	0.00	0.00
Ingest.Matbudro	0	0	0	0.00	0.00

Last Updated 17.13.25

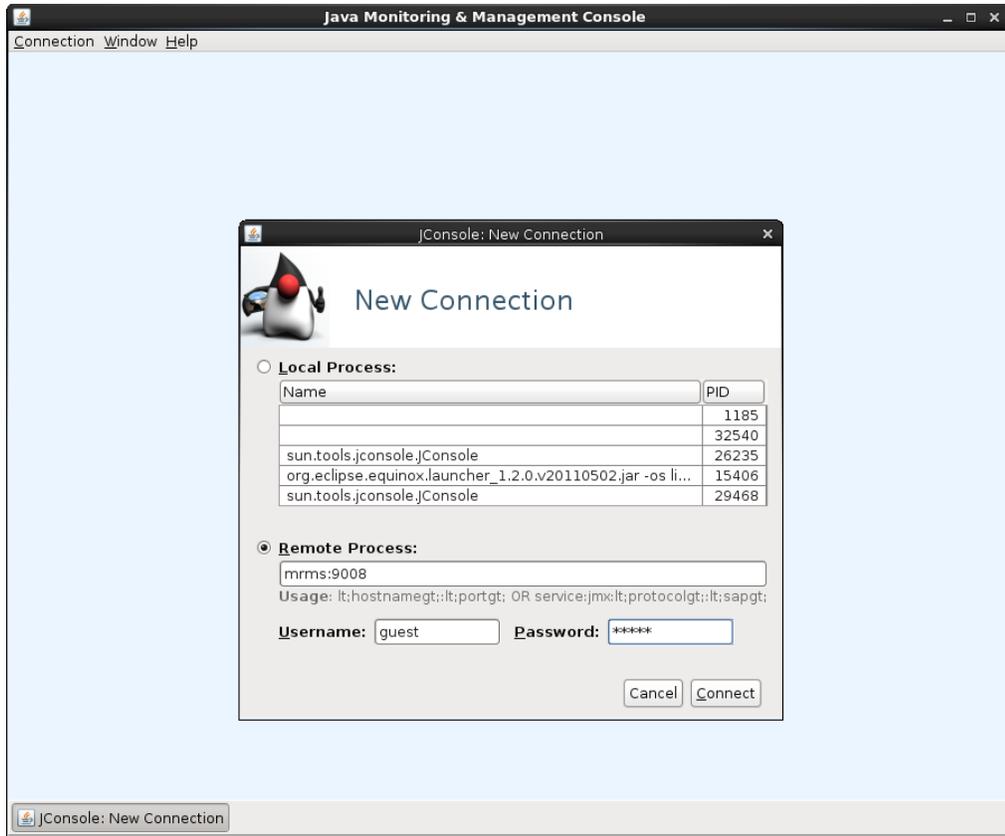
## 5. Monitoring and Clearing queues

If a qpid queue is backed up or you need to clear out all the queues to start over, there are two ways to do it:

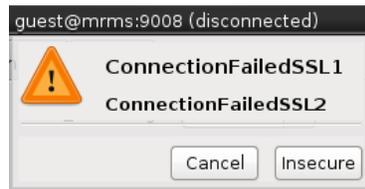
- Method 1: Use jconsole

At a terminal prompt, type `jconsole`

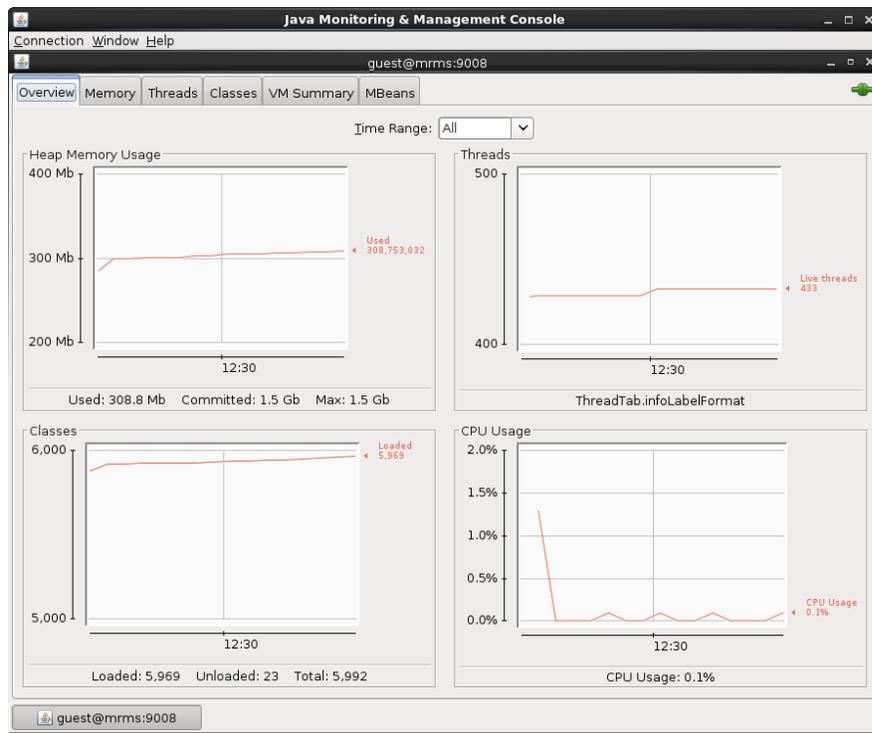
When the jconsole GUI comes up use remote process to connect to the name of the host and with the qpidJMXport (e.g., localhost:8999 for EDEX\_00; port number “J” from above). The Username and Password are both guest.



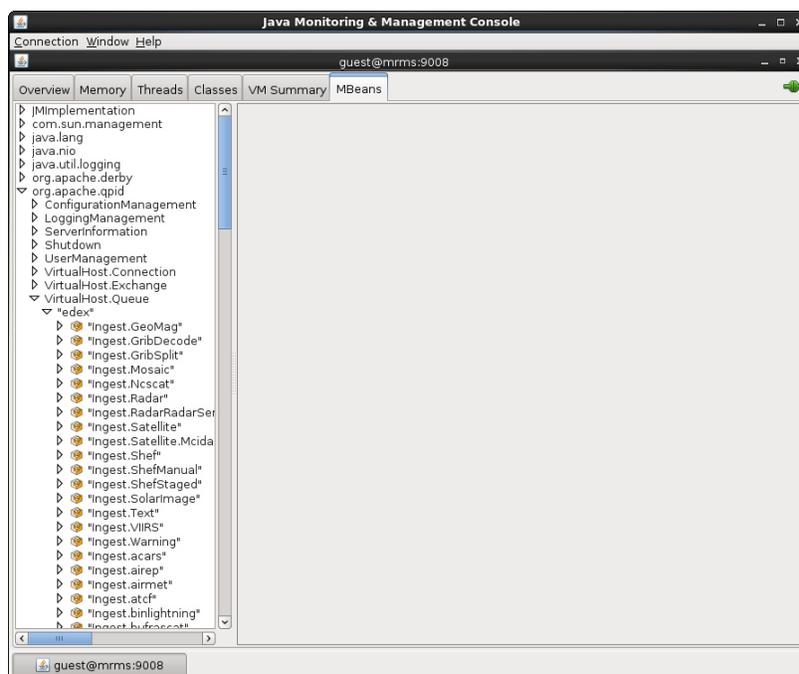
It's OK to use an insecure connection when that question comes up.



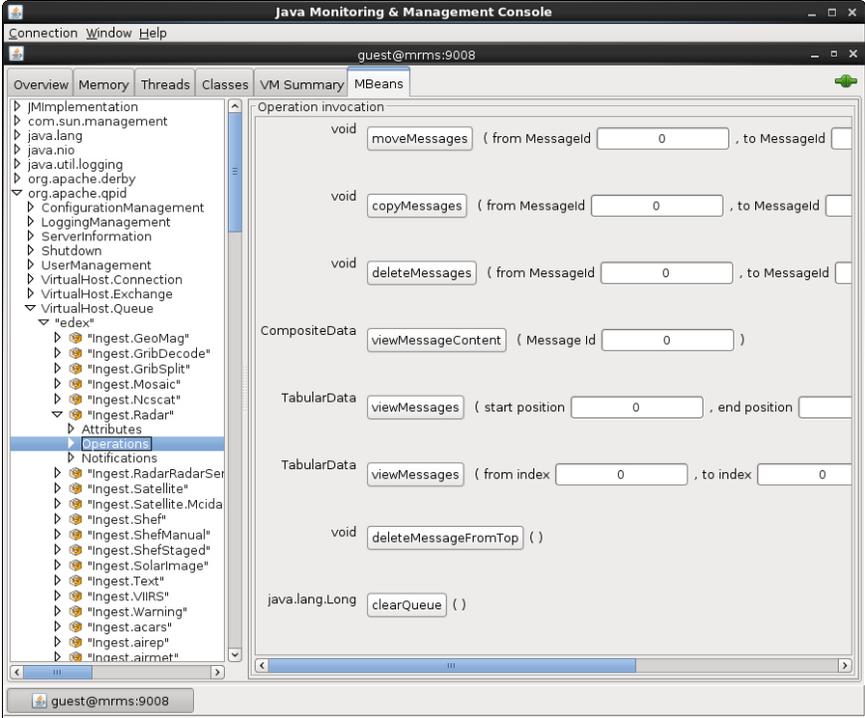
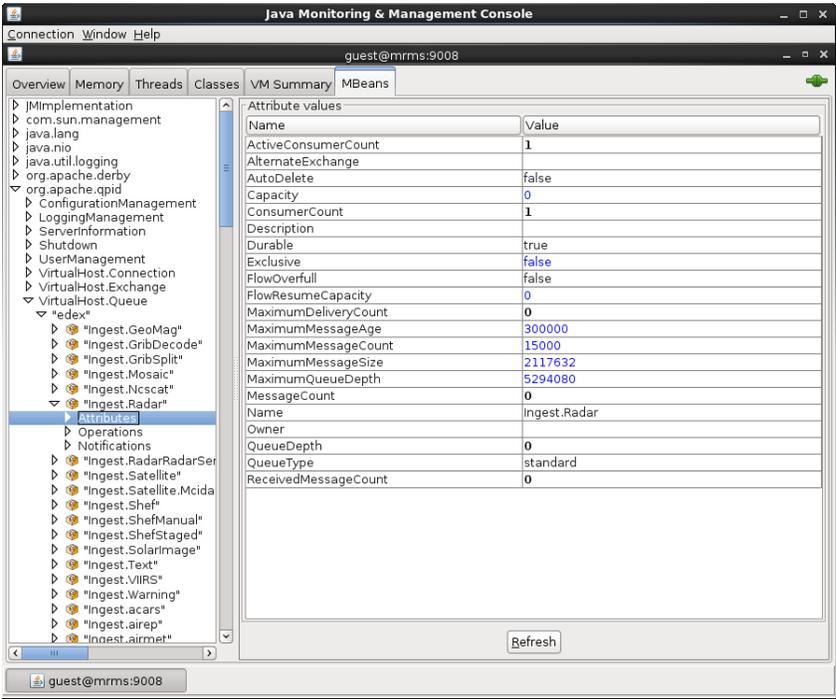
When the jconsole monitoring GUI comes up, go to the MBeans tab.



On the MBeans Tab, at the left, open up the tree for org.apache.qpid  
 ➤VirtualHost.Queue➤edex. Here you will see individual queues.



You can also examine various attributes (messageDepth is how many messages are currently on a queue) and clear out individual queues.



- Method 2: The second way to clear out a queue is to completely stop qpid. You can stop the whole instance or just the qpid part of the instance. Once qpid is stopped, you can delete the message store (e.g., delete this directory for the EDEX\_00 instance: /usr/local/edex-environment/EDEX\_00/qpid/messageStore/edex)

### **Additional Information:**

### **Starting and Stopping individual EDEX instance components**

```
cd /usr/local/edex-environment/EDEX_00/edex-environment
```

For each component run the appropriate script (as root):

```
/bin/bash edex_camel start          (or stop)
/bin/bash edex_postgres start       (or stop)
/bin/bash qpidd start               (or stop)
/bin/bash httpd-pypies start        (or stop)
```

### **Archiving Reprocessed Data**

Once a case (or part of it) is reprocessed, you can use it in the EDEX that it is located in, but it may also need to be archived. You can use the AWIPS archiver on it for whichever instance it was created with or can use the WES-2 Bridge New Case option to make a case out of it. Both functions are roughly equivalent. If you want to use the AWIPS archiver, you will need to wait until the dBArchive process runs, generally at :40 after the hour.