AWIPS-2 Reference Sheet

EDEX Distribution Files

Updated for AWIPS Build OB13.5

Introduction

Distribution files tell EDEX how to invoke a decoder plugin on a raw data file. The base files are in /awips2/edex/data/utility/edex_static/base/distribution/ and site level files are in …./edex_static/site/<your site>/distribution . Each plugin has a distribution file that contains which files that plugin can process. Most of these files are named according to its corresponding plugin (e.g., radar.xml, satellite.xml).

Recall that raw files are written to /data_store and a message is sent via QPID to the EDEX distribution service from either the LDM, the RadarServer, or LDAD. The distribution service compares the file header that's sent in the message against the regular expressions in every distribution file. When it finds a match, then the raw file is placed in a queue for the matching plugin to decode and process. The distribution files are used to match file headers as well as filenames, which is how files dropped into EDEX's manual endpoint (/awips2/edex/data/manual) are processed. For data from your local RPG, the RadarServer actually writes its raw files according to the AWIPS-1 radar directory structure (/data_store/radar/<radarid>/<product mnemonic>/….). Those files would have to be renamed before being dropped into the manual endpoint for manual reprocessing. For example, a particular 0.5° base reflectivity product from the KOAX radar written by the RadarServer would be located in /data_store/radar/koax/Z/elev0_5/res0_25/az0_5/level256/koax.153.20120106_1625. (In this filename, the "153" designates the radar product number; in this case, 153 stands for a super-resolution 8-bit base reflectivity product.) By looking at the radar.xml file below, you can see that a koax.* filename would not be matched. In real-time processing, the radar server adds "RadarServer" to the beginning of the file header sent in the message via QPID (for example, "RadarServer.koax.153.20120106_1625"). So if these radar files needed to be manually reprocessed using the manual endpoint for any reason, either a site level override of the distribution file needs to include a pattern that starts with "koax", or the koax.* files need to be renamed to RadarServer.koax.* prior to copying them into the manual endpoint.

Editing

Because these files are in edex_static/base/distribution, they have to be manually edited using a text editor (because edex_static files are not editable in the localization perspective). You should not edit the base files, but you should copy the base version to your site and then edit the site version. The regular expressions in the distribution files need to correspond with the regular expressions in the LDM pqact.conf file. If patterns exist in pqact.conf but are not in the distribution files, then raw data files will be written to /data_store but would never be ingested and processed by EDEX. Entries for these non-ingested files would be written to the unrecognized files log in /awips2/edex/logs.

Examples

Note: all files referenced here are located in /awips2/edex/data/utility/edex_static/base/distribution.

obs.xml: processes any file header that starts with "SA" or "SP", which should match any WMO header that contains METAR data (e.g., SAUS, SPUS, SACN, SAMX, etc.)

```
<requestPatterns xmlns:ns2="group">
    <regex>^S[AP].*</regex>
</requestPatterns>
```

text.xml: processes lots of WMO patterns. The second pattern `^S[A-CEG-Z].*` **matches any header that starts with "S" except for "SD" or" SF", so it also matches the "SA" and "SP" files that the obs.xml plugin matches. This means that METARs are processed by both plugins simultaneously. The obs plugin processes the metars for display in CAVE, and the text plugin processes the metars for the text database and the text workstation.**

```
<requestPatterns>
    <regex>^[ACFNRUW][A-Z].*</regex>
    <regex>^S[A-CEG-Z].*</regex>
    <regex>^T[BCX].*</regex>
    <regex>^SF[A-OQ-TV-Z].*</regex>
    <regex>^SDUS1.*</regex>
    <regex>^SDUS4[1-6].*</regex>
    <regex>^SDUS9[^7].*</regex>
    <regex>^SFU[^S].*</regex>
    <regex>^SFUS4[^1].*</regex>
    <regex>^SFP[^A].*</regex>
    <regex>^SFPA[^4].*</regex>
    <regex>^SFPA4[^1].*</regex>
    <regex>^BMBB91.*</regex>
    <regex>^N.*</regex>
    <regex>^F[EHIJKLMQVWX].*</regex>
</requestPatterns>
```

radar.xml: matches files from the LDM and the RadarServer. The RadarServer's file header messages always begin with "RadarServer"; they don't have WMO headers.

```
<requestPatterns >
    <regex>^SDUS[234578]. .*</regex>
    <regex>^RadarServer.*</regex>
</requestPatterns>
```

dhr.xml: matches a subset of RadarServer files and some of the radar files that arrive over the SBN. The numbers at the end of the RadarServer regular expressions refer to the radar product numbers (32 = DHR, 80 = STP, 138 = digital storm total precip; these numbers are defined in interface control documents published by the Radar Operations Center at http://www.roc.noaa.gov).

```
<requestPatterns xmlns:ns2="group">
    <regex>^SDUS8. .... .*</regex>
    <regex>^SDUS5. .... .*</regex>
    <regex>^RadarServer.*.32</regex>
    <regex>^RadarServer.*.80</regex>
    <regex>^RadarServer.*.138</regex>
</requestPatterns>
```