

Adding Model Sets With Unified Grid Decoder

Replaces AddLocalGrib for releases previous to OB13.1.2

Document Contents:

[Background](#)

[Adding a New Model Definition](#)

[Restart ingestGrib](#)

[GFE ParamterInfo Files](#)

Background

One of the biggest changes that Unified Grid brings with it is the concept of floating grids. The concept is that the geospatial information can change for a model, and that model can be ingested without configuration changes and displayed correctly.

To support this, unified grid identifies models based on center, subcenter and processid. This information, and geospatial projection, is read directly from the raw grib/grib2 file upon ingest.

If EDEX decodes a raw grib file that has a geospatial definition xml file which matches, it will use that file for continuing the identification process. If it does not have a unique center/subcenter/processid combination, it will use the data extracted from the raw grib file.

Another way to say this is as long as there is a unique center, subcenter and processid defined in the system with a grid model identification xml file, a model can be identified and named correctly upon decode/storage without a specific geospatial file setup.

To this point, the only purpose of grid definition files is to be able to name models differently. We can have more than one 218 model for example as its just a name field.

A catch to the grid definition files is that a grid still may not be named properly if additional fields are included, but are pointing to incorrect information. For example, if you have a <grid> tag which points back to a geospatial file through the <name> tag, and the information in the geospatial file does not match the raw grib then the model still will not be named properly.

EDEX reads the grid model definition and geospatial definition xml files when it starts, and stores the information into maps within memory. This information is used in the definition and storage of grid models.

The location of these files should be as follows:

/awips2/edex/data/utility/edex_static/base/grib

FILE OR DIRECTORY	DESCRIPTION
dataFieldTable.txt	Deprecated file for redefining parameter names. (better to use parameter/definition/parameters.xml)
defaultSubGridCenterPoint.xml	If defined, this is the center point to use for sub gridding. If not defined the center point of the CWA is determined from the maps database.
grids/	Geospatial definition xml files, if needed.
largeGribPatterns.xml	Files matches this definition cause the ingestGrib jvm to halt all other decode threads and process just this file to overcome memory limitations when decoding large grib files.
models/	Model definition files.
postProcessModels/postProcessedModels.xml	Defines models and associated post processor to run after decoding.
subgrids/	Subgridding definition xml
tables/	Baseline parameter tables.
thinnedModels/	Defines grids that need to be stitched together, i.e. UKMET.
ucar/	Defines custom grib 1 parameter table extensions for the UCAR grib 1 decoder.

/awips2/edex/data/utility/edex_static/base/grid

FILE OR DIRECTORY	DESCRIPTION
parameterInfo/	GFE Parameter Info definitions.

Currently, before Unified Grid, sites would put model definitions in `/awips2/edex/data/utility/common_static/site/LLL/grib/models`. Long term they should be moved to `edex_static/site/LLL/grib/models` as the information is unique to the grib decode process, although today (13.2.1) `common_static` is still read. This changes in 13.4.1 if the `gribMode` is changed in `/awips2/edex/bin/setup.env` to *future*.

Adding a New Model Definition

IMPORTANT NOTE: When setting up Local Model Ingest in AWIPS II, be sure not to use an underscore (“_”) in your model name or in any parameter names if you are setting up parameter info for that model for GFE.

When a grid dataURI has an underscore in it it causes the grid inventory to reinitialize every time any product comes in for that grid causing many unnecessary and time consuming reinitializations which could realize themselves as a CAVE hang or lock-up.

Since there is no need for a geospatial definition file to be created anymore, start by creating a model definition file in the following directory.

/awips2/edex/data/utility/edex_static/site/LLL/grib/models/

The definition file can contain multiple definitions, so you can create one which contains a common group. For example, you can create one LDAD.xml which includes all definitions for models which are ingested through your LDAD server. You could also choose to create one file per model and name it in a descriptive way that identifies the model defined inside.

A model definition consists of the following XML:

Table 1. Important XML Definitions in a grib model definition .xml

XML Tag	DESCRIPTION
<model>	Beginning of grid information section
<title>	Title of model - does NOT have to be unique
<name>	Name of model - must be UNIQUE
<center>	
<subcenter>	
<grid>	Grid - specified by the <name> tag of a geospatial definition xml file
<process>	List of generating process
<id>	ID of generating process (part of <process> tag)
<paramInfo>	Name of a parameter info file used for GFE.
<dt>	The time step of the model in hours

The setup of the file is as shown below:

```
<model>
  <title></title>
  <name></name>
  <center></center>
  <subcenter></subcenter>
  <grid></grid>
  <process>
    <id></id>
  </process>
  <paramInfo></paramInfo>
  <dt></dt>
</model>
```

All of this information can be extracted from the raw grid file itself, or you may know it from the originating source of the model. Alternatively, you can use **wgrib** or the **wgrib2** programs to get this information. **IDV** is another program which can be used for this. This document will not cover the use of these programs.

Finally, you can have AWIPS II ingest one raw grid file, and see how it is defined as an unknown GridModel in /awips2/edex/data/hdf5/grid/

GridModel:7:0:161

The above has the following attributes:

Center:: 7
SubCenter:: 0
Process ID: 161

This information is enough to define a grid so it can be identified and properly named in the system, however, the rest of the information is useful also. The only **necessary** fields to name a model properly are as follows:

```
<title>  
<name>  
<center>  
<subcenter>  
<process>  
  <id></id>  
</process>
```

It is recommended that you do NOT include a <grid> line. This will allow your grid to “float” in the event that the provider changes the geospatial information.

Now, to create the file follow these steps:

1. Change directories to /awips2/edex/data/utility/edex_static/site/LLL/grib/models/ using the cd command.
2. Create the file, and add in the appropriate information. Again, the file name is arbitrary, but choose one that makes sense to you. Below is an example which comes from the above HRRR example. Remember to choose a name of the file which is representative of the model:

vi HRRR.xml

```
<?xml version="1.0" encoding="UTF-8"?>  
<gribModelSet>  
  <model>  
    <title>HRRR</title>  
    <name>HRRR</name>  
    <center>59</center>  
    <subcenter>0</subcenter>  
    <grid>9001</grid>  
    <process>  
      <id>125</id>  
    </process>  
    <dt>1</dt>  
  </model>  
</gribModelSet>
```

NOTE: The <grid> MUST match the <name> field in the geospatial information xml IF YOU CHOOSE TO USE IT. It is recommended NOT to use it. The rest of the information comes from the steps gathered above. <dt> is the time step of the model in hours. For example, the normal ETA is in 6hr steps. RUC is 1hr.

Now just ensure proper ownership of awips:awips

chown awips:awips HRRR.xml

Restart ingestGrib

After creating all these files you should restart all instances of ingestGrib with the following command on each EDEX server:

**service edex_camel stop ingestGrib
service edex_camel start ingestGrib**

GFE ParamterInfo Files

See Appendix V of the AWIPS II SMM for information on this topic.