

AWIPS II Topic of the Week -- EDEX

All:

Over the next several weeks we will introduce you to the basic elements of the new AWIPS II architecture. (*For a high level introduction to the overall architecture, see the weekly topic from 5/30.*) First off is the EDEX. The EDEX is the part of AWIPS II which will ingest and store data. EDEX stands for Environmental Data EXchange (EDEX). Like all of AWIPS II, the EDEX uses many open source projects. The backbone of EDEX is the Mule Enterprise Service Bus (ESB). The Mule ESB manages the exchange of information between a collection of loosely coupled "Components". Components do things like read files, handle external requests, and talk to databases. The Components communicate with one another (and with the outside world) through Endpoints. An Endpoint can be a file on the disc or, most commonly, a message queue from which other Components are reading. The connections between the EndPoints and Components are defined by configuration files written in xml. As an example, when a satellite product arrives at a site to be ingested, it is picked up by a staging Component, which moves the product to a staging area for further processing. The staging Component sends notification of this event to Components which can decode, index and store the image by writing to its outbound endpoint. The staging Component does not know which Components (if any) will continue to process the image it has moved. Those details are configured (aka "wired") into the Mule ESB with xml. This loose coupling gives us great modularity and flexibility. If we need to change the way data is decoded, we need only update and test one component with the rest of the system remaining the same.

Thanks,

Ronla