J005-002 Room: C513 Time: May 28 9:15-9:30

Pandora Project: a trial to develop an HTTP-based distributed database

# Eizi TOYODA[1]

[1] JMA NPD

1. Introduction

In operational weather service, vast and various numerical data is used. For example, numerical models read/write at fixed grid points, station data are given on scattered points, and floating buoys have unpredictable orbits. The development of the weather service enlarges data size and diversifies data formats; the cost in data handling is growing on everyday. There are three difficulties: data sharing among distributed remote hosts, inconsistency of data naming conventions, and support for diverging variety of file formats.

The Pandora project is trying to solve these difficulties. Detail of data distribution is concealed within HTTP URI relative to a virtual server that has all data. Path component of URI is composed with data search keys (called indices for array). Data format is identified by MIME media type, and trivial conversion is done on the HTTP relay server.

2. Data Model and Identifier

Traditionally, there are three data models: array model for gridded point value, relation model for tabular data, and tree structure for disk file directories. Mathematically, the data models are essentially same: every identifier is made from list of denumerable keys (indices or path elements) and can be mapped mutually. The efficiency of mapping is, however, of great interest. Array indices are easily mapped into relational table entry, and table can be easily mapped into tree structure. Thus the Pandora project uses tree-structured identifier for data: that was URI.

At present time, URI mapping rules are defined for various gridded data: NWP products, for geostationary satellite images, radar images. Only level layer access is implemented. Support for tabular data like station observations is in experimental state.

3. Servers

Current implementation of Pandora server is a set of PHP scripts. They work as CGI program under control of Apache HTTP server. Task of Pandora server has three aspects: to serve local data access, to convert data format automatically, and to relay request for remote data.

Service is identified with URI and MIME media type. The client can specify acceptable media types in two ways: traditional URI suffix (like `.html' for `text/html') and content negotiation with `Accept:' request header. The servers will response in requested media type directly if it has corresponding driver.

Automatic media type conversion may be performed if there is mismatch between acceptable/available media types and if conversion can be defined.

Relaying operation is designed with care of communication control and performance, since the Pandora service is planned to be used as main service of JMA's domestic network. First, TCP port can be configured for each relay setting entry. This will serve as service class flags; they can be regulated by bandwidth controllers on the market. Second, relay setting may have redundant routing. Finally, relayed communication can be cached; this will improve performance on narrow or large-latency networks.

At present time, work is in progress on three topics: explicit cache controlling, reducing index polling with update notice, and partial array access.

4. Clients

In most basic sense, all HTTP clients will work as Pandora client. Even WWW browser can also be used to examine Pandora dataset.

Existing graphic/analysis applications can be Pandora client if their file access is rewritten as HTTP accesses. Only this will liberate the application from data location and data format. At present time, this work is done for a JMA's operational program `GMFV'.

More generalized and popular API like netCDF is in plan. This will make many well-known software tools to deal with latest NWP result on remote servers.