OPeNDAP's Server4

Building a High Performance Data Server for the DAP Using Existing Software

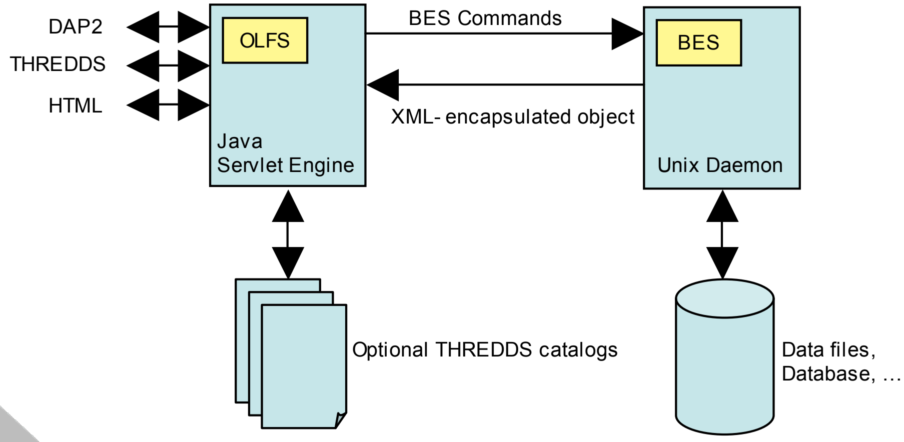
In 1993 the University of Rhode Island began a NASA-funded effort to develop a data system for ocean data that would take advantage of the Internet. Out of that effort, the Data Access Protocol, a server implementing the protocol and several compatible client applications were developed. The data server was based on the Common Gateway Interface (CGI) used for many web-based interfaces, largely because of the simplicity of those types of systems. As the use of the server spread, its limitations became more apparent. At the same time, other issues with the data system became apparent and new work was started to address those issues (such as metadata non-uniformity, searching difficulty and collection organization). Server4 is one aspect of that work. Its goals are to make it easier to solve these problems and to reuse as much of our (and others') existing DAP-compliant software as possible.

The essential features of Server4 are:

❑Two process design can be adapted to a variety of different configurations

❑The server supports several different application protocols for different types of information

❑Formal communication between the processes (to be published soon) will enable other groups to write their own front-ends



Server4 Supports several protocols and a high level of reuse using a two process architecture

Software Reuse

Software reuse is an important part of the Server4 effort. Since the start of work on the DAP, significant effort has been made developing DAP-compliant server software (client software too, although that is not the focus of this poster) and to make a new server we needed to use as much of that work (and knowledge) as possible.

In the OLFS, the THREDDS implementation is 100% from the Unidata THREDDS library. Other elements were written new for this project.

In the BES, the framework itself is an extended version of the Earth System Grid II data server developed by NCAR/ESSL/HAO. It represents about 70% reuse.

The BES Format Modules are about 5% new software with the remaining 95% reused from the Server3 'Data Handlers' (in fact the source code distributions for the format modules still build the Server data handlers too).

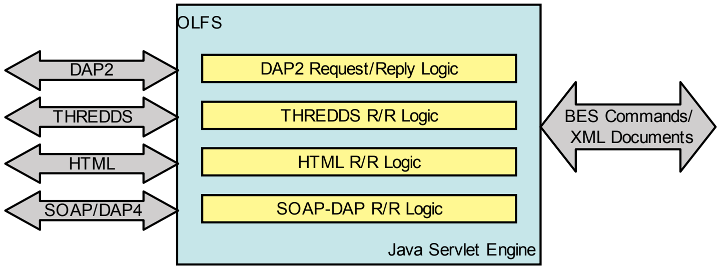
The PPT modules used by both the BES and OLFS are inherited from the ESG II project

The OLFS

The OPeNDAP Lightweight Front end Server (OLFS) is the first of the two processes which make up Server4. The OLFS is implemented in the Java programming language using the Java Servlet (J2EE) technology. The functions carried out by the OLFS are limited to examining the incoming request, passing that request to another process or servlet and packaging the result so that it is a correctly-formatted response. The OLFS does not need to interpret the content of the client requests or BES responses, only how to encapsulated them in different types of MIME, SOAP, or XML documents.

In addition to support for OPeNDAP's Data Access Protocol, version 2 (DAP2), the OLFS provides support for:

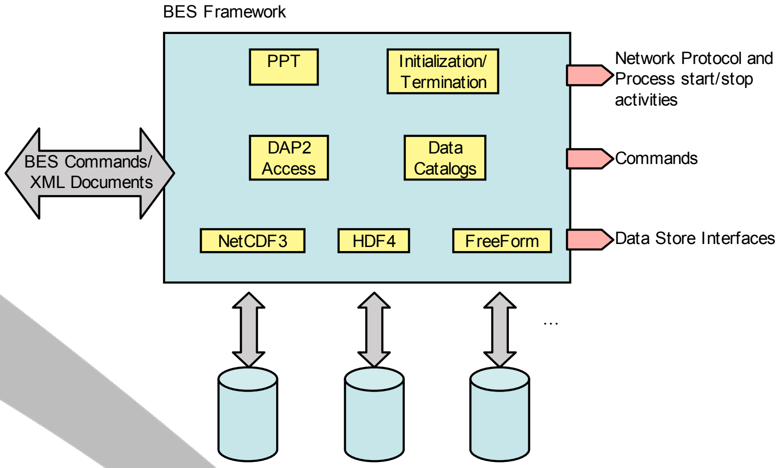
THREDDS data catalogs
HTML directory pages
A prototype SOAP interface for DAP (aka DAP4)



We choose to implement the OLFS using Java Servlets because those provide a good balance between:

Increased performance
Ease of installation
Security features
Code reuse

The BES



The BES is a framework for developing high performance data servers developed at the High Altitude Observatory of NCAR/ESSL. The BES’s essential feature is that virtually all of its functionality is implemented through *plugin* modules loaded at run-time. This provides a way to combine software components developed for different projects into a new data server.

The BES supports three main types of plugin modules:

Network protocol and initialization/termination modules
Command modules
Format modules

As delivered for use with the OLFS, the BES includes a PPT network protocol module, DAP and data store catalog command modules, and a number of format modules.

The different types of modules are defined by C++ classes which describe the interface that a particular module must implement. Specializations of these classes are compiled to shared-object libraries and loaded by the BES at run-time using information in the BES configuration file.

The BES has more advanced memory management than the Server3 software, which is needed for large data requests.

The BES includes support for SSL authentication, although this is not currently accessible through the OLFS.

OPeNDAP Staff & Resources

Peter Fox: President; p.fox@opendap.org
James Gallagher: j.gallagher@opendap.org
Daniel Holloway: d.holloway@opendap.org
Nathan Potter: n.potter@opendap.org
John Chamberlain: j.chamberlain@opendap.org

•Main web site: www.opendap.org
•SCM: http://scm.opendap.org:8090/trac
•Developer's Wiki: http://wiki.opendap.org
•Email list: opendap@unidata.ucar.edu
•Tech Questions: opendap-tech@undata.ucar.edu
•US Mail:
165 Dean Knauss Dr.
Naragansett, RI 02882
Voice: (401) 286-1304

Technical and Issue Interests

OPeNDAP's primary interest is in the development of software to facilitate sharing highly structured data typically found in scientific or similar subject areas.

OPeNDAP is a Not-For-Profit corporation, Incorporated in Rhode Island