The Prospero Distributed Operating System

Department of Computer Science and Engineering
University of Washington

Personnel and Contact

B. Clifford Neuman
University of Southern California
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, California 90292

+1 (213) 822-1511
bcn@isi.edu

Project Summary

As systems grow, it becomes harder for the user to organize, identify, and locate the information and services that are of interest. The Virtual System Model is an approach to organizing large systems that makes it easier for users to keep track of and organize such objects. Prospero is a distributed operating system based on the Virtual System Model. It allows users to build their own "virtual systems" from the files and services that are available to them, making it easier to keep track of the objects of interest in the resulting name space. Tools are provided that allow a name space to be specified as a function of other name spaces. This relieves part of the burden of keeping name spaces up-to-date. A disadvantage of supporting multiple name spaces is the lack of name transparency: the same name may refer different objects when used by different people. Prospero addresses this problem by supporting closure.

Description

A *virtual system* defines a view of the world centered around the user. Those files and services of interest to the user have short names, while the names of objects that the user is less likely to access are much longer. Users can specify parts of their name space as a function of one or more other name spaces. This is accomplished using two tools: a *filter* is a user defined program, associated with a link, which changes the way one views objects seen through that link; a *union link* takes the contents of a linked subdirectory and makes them appear as if they are part of the directory containing the link.

Naming in Prospero is "object-centered". In *object-centered* naming, multiple name spaces are supported, and each object has an associated name space (called the closure). By supporting closure, it is always clear which name space is to be used when resolving a name.

Prospero runs as a network service. Prospero servers collectively implement a distributed directory service independent of the mechanism used to actually store the objects. In addition to directory information, the directory service stores information about individual objects. This information includes replication, closure, and access control information, a list of cross references, and a (possibly incomplete) list of back-links to the directories which name the object.

Client applications can run on any system, not just those running the server. No kernel changes are required. Several applications supporting the Prospero name space are provided and a library redefining the open system call allows existing applications to use the Prospero name space. Kernel support is planned.

Because it is intended to scale across administrative boundaries, support for heterogeneity was an important requirement in the design of Prospero. Prospero supports multiple access methods for the files it names. Among the supported methods are anonymous FTP, Sun's Network File System and the Andrew File System.

Access to individual objects in Prospero is mediated by existing access mechanisms and the security of such access is defined by those mechanisms. Access to read or write directory information is controlled by access control lists associated with directories, and with individual links within a directory. Prospero uses interchangeable authentication mechanisms and authentication information is tagged with the method used. The prototype uses an authentication type of "assertion"; a future release will incorporate Kerberos. Other authentication mechanisms may be added as needed.

Scalability was a key consideration in the design of Prospero. Prospero is intended to tie together pieces from numerous systems that are geographically distributed and which cross organizational boundaries. In addressing scale, Prospero is not just concerned with the demands placed on the system. The demands placed on the user are just as important. The user-centered view supported by Prospero is one way of limiting those demands.

Virtual systems may be set up by users or organizations. They may be created for individuals, projects, or any other purpose for which a separate view of the world might be useful. It is expected that organizations will maintain directories organizing objects in different ways, and that these directories can be incorporated into the virtual systems of those that need the information. Among these directories might be indices by author, project, subject, or any other key fields. Users can find objects by looking for them in the appropriate index, or by browsing through related virtual systems.

Status

Prospero is being used on more than 2,000 systems in 23 countries on four continents. The present implementation focuses on the file system and is being used to to organize references to information scattered across Internet archive sites (including many of the papers mentioned in this newsletter). A beta release is available, and those interested in obtaining the code should contact INFO-PROSPERO@ISI.EDU. When available, new pieces of the system will be announced on the INFO-PROSPERO mailing list. To be added to the list, send a request to INFO-PROSPERO-REQUEST@ISI.EDU.

Report Availability

Papers, reports, and documentation on Prospero are available by anonymous FTP from CS.WASHINGTON.EDU in the directory /pub/prospero/pfs/doc.

References

[1] B. C. Neuman, "The Virtual System Model: A scalable approach to organizing large systems (a thesis proposal)," Tech. Rep. 90-05-01, Department of Computer Science, University of Washington, May 1990.

[2] B. C. Neuman, "The need for closure in large distributed systems," *Operating Systems Review*, vol. 23, pp. 28–30, October 1989.

[3] B. C. Neuman, "Workstations and the Virtual System Model," in *Proceeding of the 2nd IEEE Workshop on Workstation Operating Systems*, pp. 91–95, September 1989. Also appears in the *Newsletter of the IEEE Technical Committee on Operating Systems*, Volume 3, Number 3, Fall 1989.

[4] B. C. Neuman, "The Virtual System Model for large distributed operating systems," Tech. Rep. 89-01-07, Department of Computer Science, University of Washington, April 1989.