An Architecture for Integrating WWW and CORBA

Pei-Lei Tu, Donald Ferguson
IBM Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, New York 10598, USA
914-784-7209 or 914-784-7159 (phone) 914-784-6201 (fax)
peilei, dfferg@watson.ibm.com

Daniel T. Chang
Database Technology Institute
IBM Corporation, C362/L95
555 Bailey Avenue
San Jose, CA 95150, USA
408-463-2319 (phone) 408-463-3834 (fax)
dtchang@vnet.ibm.com

The integration of WWW and CORBA is a ideal solution for deploying CORBA's distributed object technology to the wide application base of WWW and for providing WWW a scalable object server with robustness and fault-tolerance for mission-critical business applications. There are emerging activities at both OMG and W3C to define an open standard for integrating the two technologies. At the same time, there are several new commercial products or research projects which intend to support the integration by adopting one of the following three major approaches. The first approach integrates existing CORBA client-server structure into WWW by extending the Web server function as CORBA clients. Therefore, the Web server can provide the CORBA-compliant services if its CGI program is also a CORBA client. This approach requires the least modification to existing WWW or CORBA technology. Available products or projects include NeXT's WebObjects and CERC's Web*. The second approach is to add interoperability of HTTP with other communication protocols such as IIOP and DCE. This approach relies on some gateway programs to bridge the client and the server. ANSA's distributed objects and W3Objects are two examples. The third approach combines clients and servers from either WWW or CORBA with Java-enabled design. Existing client-server structures will be enhanced while the server becomes an integrated WWW and CORBA server and the client has both WWW and CORBA functions. This approach is adopted by many recently announced products including SunSoft's Java Object Environment(JOE), Iona's Orbix for Java, and PostModern's Black Widow.

Even though there are growing activities for promoting the integration of WWW and CORBA, there is little in-depth analysis of the differences in their individual client-server structures. Such an analysis could provide a more seamless integration of WWW and CORBA. We can start this analysis from the perspective of a three-tiered client-server structure of each technology, which hopefully can lead to a more seamless integration of WWW and CORBA. The focus of our investigation into the architecture design of integrating WWW and CORBA is on providing a seamless solution for three-tiered client-server computing consisting of presentation objects, business objects and data objects. The solution will include a collection of key components which can at present be implemented with existing technology and later migrate to newly adopted standard or technology. Some of the key components include:

- Generation of dynamic HTMLs from presentation objects defined in IDL. WWW stresses the implementation for presentation (user interface) in HTML while CORBA client can accept any programming implementation for presentation as long as the presentation object's interface is defined in IDL. This component is to seamlessly bridge such a difference between WWW and CORBA.

- Interpretation of meta-HTML or macrofiles for user interface layout, error-handling, integrity checking, etc., under the adoption of distributed object design.
- Typecode convertion between components with different type systems, such as HTML and CORBA objects, or HTML and Java objects.
- Extensible DII. The integrated clients would need a more dynamic invocation interface than current CORBA clients as the object's server could be a mixture of local or remote servers as some methods will be dispatched remotely and some will be executed locally if downloaded as Java applets.
- Interfaces between the transient or persistent objects of business and data tiers, presentation and business tiers, and presentation and data tiers. One major design issue is the the statelesss communication of HTTP and repetitive spawning of CGI processes while CORBA allows other effective communication protocols and reuse of server resources. Such a difference would impact the content and extent of stored state information and any phased approach to migrate to new standards, such as HTTP/NG.
- CORBA-compliant Web clients.
- CORBA-compliant Web servers.