

The 'Uberportal': An Answer to the Multiple Portal Problem

Multiple portals in large enterprises are usually deployed as stovepipes. Enterprises should build a portal architecture that allows interoperability, and should examine the "uberportal" as a viable architectural option.

Core Topic

Internet Platforms and Web Services:
Enterprise Portals

Key Issue

What are the best practices for deploying and managing enterprise portals, and how can enterprises achieve and measure return on investment from their portals?

Strategic Planning Assumptions

The strategic solution to the multiple portal problem will come with federated portal standards in 2005 (0.7 probability).

Vendors will deliver JSR168- and WSRP-compliant versions starting in 4Q03 (0.9 probability).

In the early days of enterprise portals, a single portal was commonly deployed to a particular target audience. For example, for a business-to-employee presence, the company selected one portal product and built one enterprise portal. If multiple audiences were addressed, it was not unusual to see a separate portal server instance deployed. This approach didn't cause significant problems because these audiences were typically managed separately and had different requirements.

Over the years, however, multiple portals for a single audience type have become more common. Multiple portals are now the rule, not the exception. We predicted this in early 2000 (see "Multiple Portals in Your Enterprise? Count on It!"), and it has become reality for many large enterprises, as well as for some small and midsize businesses.

Reasons for Multiple Portals

- Vendors of packaged applications frequently have their own portal products. These may be delivered via an original equipment manufacturer (OEM) of a platform-centric portal product (for example, i2 Technology's OEM license relationship with IBM for WebSphere Portal) or may be via its own portal technology (for example, SAP's Enterprise Portal). In some cases, the application vendor compels users to use its portal to access its application, by delivering functionality that is difficult or impossible to replicate in other portal products via the portlet model.
- In highly decentralized enterprises, different business units or regions may have chosen different portal products for their users.
- In some enterprises, disputes regarding technology platforms are in "full swing." The tactics of armed camps of .NET bigots

Gartner

against the armed camps of Java 2 Platform, Enterprise Edition (J2EE) bigots will drive multiple portals in an enterprise.

- Some enterprises lack appropriate governance over portal activities, which allows multiple, rogue portal efforts.

Enterprise IS departments want to minimize the number of portals to reduce acquisition, maintenance and support costs. However, in many enterprises, the businesses have forced their IS departments to support multiple portals — a trend the IS department should embrace rather than fight. Enterprises should build a portal architecture that allows for multiple portals, and seek portal products that are based on open architectures and on standards to allow for easier integration.

Enterprises that need to manage multiple portals struggle with the lack of interoperability standards. The emergence of portlet standards — Web Services for Remote Portlets (WSRP) and Java Specification Request 168 (JSR168) — in 2H03 will do little to provide the level of interoperability required. Until a more-complete set of interoperability standards emerges to allow seamless portal federation, enterprises must take a tactical approach. A common tactical model for portal interoperability is the "uberportal."

The Uberportal Concept

German for "over," uber implies a high level, overarching portal. The uberportal is a high-level, horizontal portal framework on top of one or more horizontal or vertical portals. The uberportal is essentially the entry point, or home portal, in a multiportal deployment. It will be the place where users spend most of their time and aggregate most of their content and applications.

One doesn't buy an uberportal, one builds it. The best portal products for the uberportal are open architected, service-oriented and standards-based.

There are five keys points of interoperability between the portals in a multiportal deployment:

- Directory
- Security
- Personalization data
- Metadata
- Portlets

Portlet interoperability will be accomplished via the adoption of WSRP and JSR168. Vendors will deliver JSR168- and WSRP-compliant versions starting in 4Q03 (0.9 probability). However, the other points of interoperability are problematic, at best. If the enterprise is lucky, it may have a rationalized directory infrastructure. Many enterprises have multiple directory strategies, most of which are disconnected. Common security across portals typically is not feasible because portal vendors employ their own proprietary "permissioning" models. Security standards, such as WS-Security and the results of the Liberty Alliance, should provide help; but today, a single sign-on (SSO) tool is generally as good as it gets. Personalization models are also proprietary, and no standards (or standards efforts) exist. Metadata standards (for example, Dublin Core) are incomplete and largely not adopted by portal product vendors.

A Minimalist Uberportal

Although the optimum uberportal addresses all five points of interoperability, some of these points are too difficult or costly to implement.

This has led to the concept of the "poor man's" uberportal, a minimalist implementation. However, given the cost and time required to establish a robust uberportal, the poor man's uberportal has become the dominant model for uberportals.

A poor man's uberportal features include:

- Rationalized directory services (common or replicated directory)
- SSO to all related portals
- Log on points user to personalized start page in the uberportal
- Access to lower-level portals is via hyperlinks or tabs on their start page
- When the user clicks on the link to the lower-level portal, his or her personalized start page in the lower-level portal is rendered, typically in a separate window
- When users finish their tasks in the lower-level portal, they close that window and return to the uberportal

Models for an Uberportal Architecture

When designing an uberportal, you can use one of three models to access an application in the lower-level portal. All three models can be used simultaneously in the same portal implementation.