

High Performance Web Services Architecture

Matisse DBMS, Solaris & Java Provide Foundation for Global Services

Background

HIV InSite is a web site developed by the University of California at San Francisco (UCSF), one of the world's leading health sciences institutions, to provide comprehensive up-to-date information on HIV and AIDS treatment, prevention and policy. Launched in March 1997, it began as a source of basic information about HIV and AIDS. Since then, it has been expanded into a global information resource that includes news, conference information, links to other related web sites and an extensive collection of original material. HIV InSite is a non-profit resource that allows free, anonymous access to all available content.

Moving to a Next-Generation, Advanced Web Services Architecture

Since 2000, HIV InSite has been migrating to a web services architecture. The first step, which was recently completed, transformed the static, HTML-based (hypertext markup language) site into a dynamic, XML-based (extensible markup language) model and separated the web site's content from the presentation layer. The presentation layer is now governed entirely by XSL (extensible stylesheet language) and all documents are maintained as XML, which allows content to be delivered dynamically and changes to happen instantaneously. The new architecture dramatically reduces system administration tasks and costs – an important factor as the site continues to grow as a global resource.

The site is powered by Sun Microsystems servers running the Solaris operating environment and is implemented with Java. "Sun servers are the clear leader for web environments, since Sun invented a lot of new web techniques that weren't available on other platforms," says Norman Jefferies III senior

software engineer at HIV InSite. "Solaris runs continuously and is considerably less expensive to operate in the long term than Windows. When you're running a 24x7 global operation like ours, you need the kind of performance that Sun can deliver."

"Java gives HIV InSite a proven, stable and reliable programming environment," Jefferies explains. "With Sun's open systems and multi-vendor approach, J2EE is ideal for us."

HIV InSite is also building a number of new database-dependent applications. The first classifies antiviral medications and shows their interactions with other drug combinations. The data is multidimensional and changes based on each drug combination. HIV InSite is improving health professionals' access to statistics and other reference information from the Center for Disease Control and to PubMed from the National Institute of Health, and it will allow them to create sessions they can save and come back to.

The Next Step to Web Services: An All-Object Environment

To complete HIV InSite's move to a web services architecture and achieve the site managers' goal of providing on-demand delivery of diverse, complex content and data, they required a more powerful database solution than the one currently in place and wanted to deploy an end-to-end object-based environment.

Object databases were at the top of Mr. Jefferies' list, but he was concerned that they weren't ready for prime time; based on the problems he'd had using first-generation object databases on other projects in the mid-90s. He didn't want to use a relational or object-relational database because of what he called

"the daunting task" of translating Java objects into relational tables and back again, and the heavy administration requirements. XML databases weren't an option, either, as he knew he wanted to store more than just XML objects.

"My success is largely based on the tools I have," Jefferies explains. "If you're working in an object-based development environment, like Java or C++, to have to save information in a relational format is a huge amount of work."

"On the other hand, XML is a good format, but not for data storage. With object databases, there are no restrictions on the different types of objects you can handle, and you can work within an all-object environment from the programming language to the database. But, we found that the early object databases lacked the performance and reliability needed to provide a consistent level of service."

Matisse: Best Database for Web Services and Complex Data

HIV InSite selected Matisse as the best database for its new web services architecture, complex data requirements and 24x7, high-performance services to the global health community.

Matisse enhances the performance and capabilities of web services and other network applications. The versioning engine allows HIV InSite to add new information and objects in real time without any downtime. Matisse's object modeling and support for hierarchies and complex relationships made it the best choice for the new drug and health databases HIV InSite is building. The built-in scalability, zero administration and high performance to meet continually fluctuating usage demands ensures that HIV InSite is available 24x7.

"With Matisse, I have a truly zero-administration database," says Jefferies. "We're a nonprofit, so we don't have a lot of people here. The fewer tasks I have for a system administrator, the easier it is on our resources."

"It's clear that Matisse's engineers did their homework, because there's nothing missing from Matisse," Jefferies continues. "It's a database any object developer would want to use. And Matisse Software has been extremely responsive to all my requests. You can't say that about too many companies."

©2003 Matisse Software Inc. All rights reserved.