Service-Oriented Architecture— What Is It, and How Do We Get One?
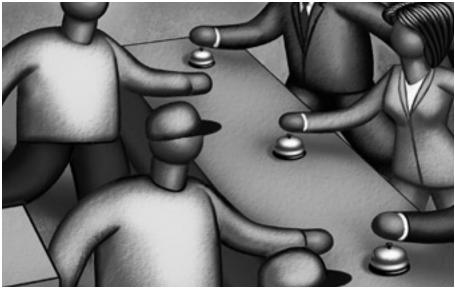
SOA can provide a secure, reliable, and agile architecture that benefits the entire institution

By **Jim Phelps** and **Brian Busby**

Everyone involved in information technology (IT) at higher education institutions around the world has likely heard of service-oriented architecture (SOA). Most of us have been contacted by vendors selling SOA solutions. But what is SOA, how does it work, and how might our schools benefit from it?

The Division of Information Technology (DoIT) at the University of Wisconsin–Madison (UW–Madison) spends a significant portion of its budget on integration and interfaces. We are fast approaching a point at which there are so many interfaces and customizations to our applications that we will not be able to upgrade the systems or make changes. DoIT began considering transitioning to an SOA model because of the growing burden of managing multiple interfaces and communication protocols among complex enterprise systems.

The university's Common Systems Interoperability Architecture Working Group (CSIAWG) was tasked with finding a common integration architecture. CSIAWG, sponsored by the UW System (the overarching organization of all of the University of Wisconsin campuses), drew from various campuses and major application owners. We spent much of the past 18 months talking to people about SOA. The conversations included presentations for vendors and other campuses and various group meetings. Ultimately, CSIAWG recommended that SOA should be the future integration architecture, offering a way out of integration gridlock.

SOA Defined

Simply put, SOA presents well-defined business functions as services, which are made available to multiple applications through standard protocols. Using SOA, institutions can integrate business functions into new and interesting applications. SOA represents a fundamental shift in the way applications are built, requiring a rethinking of business processes and the role applications play in the enterprise.

SOA is built on reusable, shared, networked services, with each service a business function. It is an architecture that seamlessly connects separate technology systems through Web services—reusable software components that use a standardized messaging system—built within an Internet-based platform. It allows different kinds of systems and platforms to communicate with each other in a common language, without custom interfaces. What makes SOA valuable is its ability to reuse business functions in different combinations. With SOA, the application lives "above" these services as an orchestrated business process.

SOA also provides complete transparency of the process. For example, the library, dining hall, and computer labs might use the same service (business function) from the registrar to answer student enrollment questions (such as, Is this student in Biology 101?). With SOA, these consumers would receive answers instead of raw student data.

How Does It Work?

To understand SOA, consider the architecture commonly found on col-

lege campuses: the data-oriented system. In a data-oriented system, campus departments share information by sending it in batch-file transfers from one department to another. For example, the library might need to check individuals who wish to borrow materials to find out if they are full-time students. To do so, the library would query the student records department, which, in the case of UW–Madison, would send all 45,000 student records in a batch. The library would then load the batch into its system's database, simply to locate the data it needs for a small subset of those students.

Research firm Eduventures notes that university departments can easily misinterpret data sent by batch transfers because departments use different business rules or different versions of data to suit their unique needs. Recreational facilities, for example, might define a current student as one who is enrolled and has paid all college fees and is therefore authorized to use campus facilities. The library, on the other hand, might define a current student—one eligible to check out books—as one who is enrolled but has not necessarily paid all fees. Such differences can create obvious data-accuracy problems and minimize the usefulness of information.

The fact that the batch-file method includes all possibly needed data increases the security risk and places a large burden on already strained computing resources. Batch-file transfers inherent can be cumbersome, time-consuming processes prone to errors. As departments transfer batch files across campus, different "versions of truth" begin to circulate. In contrast, SOA is based on a service-level agreement between a service owner (source) and the service consumer. The transaction between the two is tracked at all points.

What SOA Can Do for You

SOA provides a "single source of truth" for all information, thereby reducing misinterpretation. Instead of transferring the university's entire student-record roster upon the request for a single student's information, the

registrar can send only the needed information. This fundamental shift—from shipping data to providing services—has many potential benefits including agility in deploying applications, increased data security, and improved transparency. In addition, because SOA uses Web services to send information directly to the requesting department, a transparent trail of the information's route is created. This is not the case with a data-oriented system, which replicates large batch files from one system or server to another, a method that provides little visibility within the file transfers.

SOA allows university departments to speed their decision cycles by providing accurate information in real time. Through a centralized architecture, the university can establish uniform, institution-wide business rules to ensure that departments make decisions based on accurate information. For example, if a student shows up to use a recreational facility, staff can enter the student's name and ID number and know immediately if the student is eligible for access to the facility. The answer would come from a uniform, institution-wide standard service.

By sending only the information required and increasing the visibility of the information trail, SOA improves the security and privacy of university information, which can include bank and credit card account numbers, Social Security numbers, and other sensitive human resource and financial information.

Is SOA the Way to Go?

Completely overhauling a technology infrastructure is daunting. Evaluating the architecture and providing proof points on *why* a technology infrastructure needs restructuring can be equally so.

UW–Madison has myriad enterprise and custom applications, ranging from legacy mainframe applications to modern Web applications. Each has multiple integration points using multiple integration technologies. Industry research shows that up to 50 percent of large IT enterprise budgets is spent on integration and interfaces, and UW–Madison's

numbers align with this statistic. To drive down future IT costs and enable us to quickly adapt to new needs, we needed to look at how we could connect all these systems in a way that made sense.

In addition, vendors are increasingly moving toward SOA and SOA technologies, and a number of higher education institutions are in discussions with each other and the Mellon Foundation to develop a foundation for deploying SOA. These groups are working on varied activities including a user-interface layer that will support service composition and customization closer to the end user, workflow, and a research platform for the analysis and data mining of text and rich media objects.

It became clear to us that we needed to move toward SOA to prepare for future applications from our vendors and our peers. By making services generic and common to all business processes, a university can improve the security and privacy of its transmissions while protecting its future technology investments and driving down the costs of replicating data and maintaining multiple interfaces.

Migration Strategy and Process: Where to Begin?

After choosing an SOA approach, our next step is to identify the components that would compose our SOA platform. This investigation includes several proof-of-concept (POC) projects with various vendor and open source products. We are working with Oracle's SOA Suite, Sun's Composite Application Builder, and LogicBlaze, an open source solution from the Apache foundation. Our goal is to establish a reference architecture and standard methodology for future integration.

Oracle has made a commitment to SOA with its open standards–based platform called Oracle Fusion Middleware. This platform provides support for the development, deployment, and management of an SOA. Oracle is deploying the Fusion Middleware platform with all of its current application lines. For example, Oracle's PeopleSoft Enterprise applications today position every

major integration point—more than 200 enterprise services and 2000 service operations—as standard Web services that we could access to help build our service-oriented infrastructure.

Although developing and organizing a major migration strategy is always challenging, we view the disruption as an opportunity to review and improve internal processes and assets. In many ways, implementing a new system is like remodeling a house. Typical questions might be, Where can you upgrade or make things easier, or Why not put in new plumbing and electrical systems since you have already ripped down the walls? Other questions with which we grappled and that other universities are likely to have include:

- What are our business processes, and how can they be improved?
- How do we move data from place to place within our current architecture? What is the flow?
- How should we represent data? Are there industry standards that we can use?
- Where are our pain points? What is currently a very difficult business process?
- How can we streamline our business processes, information and data flows, and new application development?
- Where is there high risk of data exposure?

Right now, UW–Madison is in the first phase of a multiyear plan to re-architect our enterprise applications. At each stage of every upgrade or change, we document business processes and how best to orchestrate such a major change. Because this is a huge undertaking, we identified only a few points from which we will begin the migration. Other universities considering an SOA approach should identify small steps toward the larger goal to ensure success.

Nontechnical Challenges

Migrating to an SOA presents a variety of technical challenges, but IT teams will be well served to consider how to address another major challenge—resistance to change. IT team members need to develop a marketing strategy to "sell" the new architecture to university constituents. They must become savvy in presenting the needed overhaul to other campus leaders in order to create an environment in which everyone is working together throughout the implementation of the architecture. IT teams need to be able to communicate the benefits of SOA in nontechnical business terms to win over the administration and university departments. For day-to-day system users, SOA will mean real-time access to data (no more delays), a single source of truth (no more redundant, inaccurate data sets), and enhanced security.

Universities also need to consider how they will set up a new budgetary model to support SOA. At UW–Madison, each application traditionally has had its own maintenance and support costs, and the "owner" department is responsible for paying them. Because SOA makes component applications available as Web services across the entire university system, a gray area is created regarding who pays for these services. We will need to move to an enterprise-centric budgeting model, one in which we pool budgets to support SOA development costs.

What's Next?

To further our cause of educating administrators, we plan to create an SOA leadership group to establish policies. Other universities working to take a similar route should examine the various functions in governance within their institutions' systems and use these groups as a model. At UW–Madison, we are setting up an Integration Competency Center (ICC) that will consist of experts in SOA and business leaders who make organizational decisions. Some questions we are in the process of answering about the center include:

- What is the correct composition of the ICC at various stages of maturity of the SOA? Who should the participants be at each stage?
- What are our governance issues? What is the appropriate framework that we would use to address these issues?
- What technology do we need? What is the appropriate order for implementation?

- What definition for business information do we have? Are there existing standards we should use? How do we adopt standards in a highly decentralized organization?

A successful migration to SOA requires that the developers, managers, and business process experts understand the concepts and the impacts of SOA. They all must understand the goals, difficulties, and gains.

We are continuing to evaluate SOA solutions, and we are currently working on a roadmap for deploying these technologies. We have started taking advantage of Web services available through PeopleSoft's Integration Broker and are now deploying Web services in our identity management suite and in other places around campus. We are also engaging the campus at large on a process for adoption of Web Services and SOA. We have published an SOA Desired State document, which details our goals and the benefits that we hope to see. We are working on a suite of strategies for reaching this desired state that involves governance, funding, and technical infrastructure deployment.

Technical leaders at UW–Madison believe that we could have the technology for an SOA in place in a couple of years. Yet we also understand that we are on a multiyear implementation schedule with deep cultural impacts that will be challenging to carry out. All of this work—including the culture change, the policy/governance plans, and technical deployment—will enable UW–Madison to build business process–driven applications. We see SOA as the way forward with enterprise workflow. We also see SOA as the solution to the constant demand for data and reporting. We know that SOA is only part of a larger solution, but we believe that SOA will help us be more agile and more creative in the solutions that we deliver to our students, faculty, and staff. *e*

At the University of Wisconsin–Madison, Jim Phelps (phelps@doit.wisc.edu) is Senior IT Architect and Brian Busby (blbusby@wisc .edu) is Assistant Director of Application Development and Integration, Division of IT.