Service-Oriented Architecture (SOA)

During the past few years, a confluence of IT customers' pressing technical and school requirements and the availability of new technical capabilities has set the stage for the next generation of computing. The IT industry labels this generation as one of "dynamic computing," and broadly characterizes it as an era in which IT resources become more automated, more agile and—most important—more synchronized with the processes and objectives that the resources support. Dynamic computing and the agility it promises, in turn, depend on various building-block tools and technologies. None of these enablers is more important or wide-reaching than that of service-oriented architecture (SOA).

In SOA deployments, software applications and school processes are presented as collections of standards-based components (or "services") that each perform a discrete function. The self-contained services are loosely coupled and designed to be called into use in a message-based, event-driven fashion, rather than being tightly coupled and hard coded to function in a synchronous, guaranteed-available mode (although SOA services can be configured to execute synchronously, if desired). The services themselves can either be built from scratch, or be created by wrapping existing education logic and exposing it via standard interfaces. Service orientation has existed in IT environments for many years, but only in proprietary forms until recently. The emergence of key standards, particularly the XML-based collection of Web services specifications, has catalyzed the creation of common SOA environments that can easily interact with one another. Standardized SOA, in turn is generating excitement among IT and school executives because it holds the potential to solve a range of technical and education problems that currently bedevil most school systems. Ultimately, SOA offers users a means to escape from the inflexible and incompatible collections of IT systems and applications that most schools have accumulated over many years. The difficulty and cost of getting such systems and software to interoperate, not to mention the challenges of modifying them to adapt to changing education requirements, have overwhelmed IT budgets and held back innovation. SOA lowers the interoperability hurdles and converts monolithic and static systems into modular and flexible components. As a result, education IT groups can focus on supporting the ever-changing education needs of its school system, rather than spending all its time and money just keeping a cobbled-together computing environment limping along.

The major benefits of implementing SOA are the following:

- Improve flexibility and adaptability by being able to rapidly create and modify applications and processes in response to changing education requirements.
- Improve system interoperability and reduced integration costs with internal and external organizations by adopting standards-based services interfaces.
- Improve IT productivity and reduced development time and costs through the creation and reuse of modular service applications.
- Establish IT as a critical contributor to education processes and academic performance.
- Better match cost of applications to actual use and business value.
- Extend the lifetime of legacy applications by exposing them to modular service interfaces.
- Tap standards-based services components and applications offered by third-party providers.

Despite SOA's promise, many school systems remain uncertain about the relevance of SOA to their specific needs, or unclear about how to evolve to this new form of computing in an intelligent and manageable fashion. The good news is that the core SOA concepts are relatively easy to grasp, as is the technology's potential for easing their most pressing pain points. Furthermore, users can move to SOA incrementally, and can often start by extending and enhancing the value of their existing systems and software. Not surprisingly, SOA has become a primary theme for all of the leading infrastructure and applications software vendors, and increasingly serves as a unifying architecture under which they align their products/services and go-to-market messaging.

SOA Leverages Application Standardization and Education Events

Inevitably, SOA's very name causes people to focus on the "services" aspect of the technology. And, indeed, much of SOA's power derives from its ability to leverage standards-based functional services, calling them when needed on an individual basis, or aggregating them to create composite applications or multi-stage education processes. The building-block services can be stored and reused, and can also be updated or replaced without affecting the functionality or integrity of other independent services. In this

latter regard, the services model offers big advantages over large monolithic applications, in which modifications to some portions of the application can have unintended and unpredictable effects on the rest of the application to which it's tightly bundled. Beyond the services aspect, however, SOA also requires another fundamental technology if it is to realize its full potential: *event-driven computing*.

Ultimately, the key goal of most SOA implementations will be to automate as much processing as possible and desirable, and to provide critical and actionable information to education participants when they must play a role in a workflow process. Realizing these goals will require the computing infrastructure itself to recognize meaningful events and respond to them appropriately—either by automatically initiating new services and processes or by notifying users of the events, putting them into context and, often, suggesting the best courses of action. This broad capability is sometimes referred to as an *event-driven architecture*, which goes hand in hand with a *services-oriented architecture*. For the purposes of this paper, we generally use the overarching "SOA" label to encompass both the service oriented and event-driven elements of this trend. Although SOA is typically positioned within the realm of middleware technology and other infrastructure applications, the architecture inherently encompasses the application software tier as well. SOA's effect on both middleware and applications has facilitated a trend underway for several years—the migration of certain functionality from the application tier to the infrastructure tier. In the past, many school enterprise applications have been largely self-contained environments that included their own proprietary security models, analytic and reporting tools, data stores, school processes, and so forth. For some time, though, we've seen the movement of much of this horizontal functionality into the middleware tier, where it can be instantiated in a standard fashion and applied to all the education logic and processes flowing within the application/services tier.

Escalating IT and Education Demands Drive SOA Adoption

As recently as 2004, there was still significant debate about the benefits that SOA could deliver and, more important, about the willingness of schools to migrate to this new architecture. However, it has been demonstrated that SOA is well on its way to broad adoption. The schools already moving down the SOA path are doing so for many different reasons that are specific to their own situations. Unifying all of the discrete SOA drivers, however, is a fundamental need that schools have to improve IT flexibility and adaptability in order to better support key education requirements. Beneath this overarching objective are a range of common challenges, many tied to the internal complexity and rigidity of the typical education IT environment. This complexity and inflexibility drives up development, integration and maintenance costs and limits IT's ability to rapidly respond to changing education opportunities and threats. Collectively, schools have spent many billions of dollars to integrate their assortment of proprietary platforms and applications, and also to maintain and extend monolithic applications that often reside in their own, walled-off functional silos. IT departments have had to provide interoperability among different data formats, different application interfaces, different message routing systems, different security systems, and a long list of other technical disconnects. It was possible, if expensive, to custom code point-to-point integrations of all these elements when working within the bounded realm of a single school's internal IT infrastructure. In the age of the Internet, it quickly became impractical to apply the custom approach when dealing with dozens of suppliers and partners or with hundreds, thousands or millions of Web-based users and their IT systems. The only practical solution to this escalating complexity was to create a standards-based layer (or layers) on top of the disparate infrastructure. This new layer—formed initially by individual Web services specifications and, increasingly, by comprehensive SOA-based initiatives—can serve as a common foundation for development, communications and business process flow.

First Steps for Schools Wanting to Realize SOA Benefits

As should be clear, SOA and other high-value technologies such as virtualization and policy-based management long ago passed the theoretical stage. Many school IT departments are well along in adding these capabilities to address both tactical IT and strategic education needs. They are also beginning to explore and adopt synergistic technologies such as event-driven architecture and education activity monitoring, which complement SOA and which will increasingly be considered to be necessary capabilities that reside under the overall SOA umbrella. Schools are being aided in their efforts to adopt the full range of SOA functionality by the growing availability of standards-based products, including those comprising the School Interoperability Framework (SIF). Still, schools need a starting point—or, more accurately, optional starting points, depending on their capabilities and requirements—to adopt SOA. Although it's tempting for IT managers to start by tackling some of the technical aspects of SOA, that typically isn't the best first step. Even before beginning to develop, expose and manage services as part of a SOA project, IT managers must first enlist the backing and participation of their school executives and department

heads in the process. In fact, SOA—whose entire purpose, after all, is to establish more flexible and agile IT resources to better support education activities—will often serve as the mechanism that catalyzes a more collaborative and effective relationship between a school's IT and education groups. SOA, in effect, requires breaking down the wall that often exists between the IT department and the school participants. For their part, schools must start to reorient their technology-centric perspectives, and also may have to cede some control to education strategists and users. These school users, in turn, will increasingly be able to rely on rapid IT support for addressing new education needs. Over time, school users will also be able to create and modify applications and business processes by composing services themselves, rather than asking IT departments to code them. The sharing and reuse of common services across a school enterprise will also require different school units to become less territorial, so that broad education standards and services can be established and used consistently. Once IT and school leaders jointly commit to pursuing a SOA strategy to improve their education agility, the first step is to wrapper and expose existing application functionality as Web-services-enabled services that users and/or applications can access via standard interfaces like ProActive School Services. In addition to SOA-enabling their internal applications, users can increasingly also tap their familiar packaged applications via the same standard interfaces. That's because ProActive School and other application vendors utilize functional service modules that can function and be accessed as independent services. This adoption of Web services standards within the major application suites will make it easier for schools to integrate elements of the packaged applications with internal software applications that have been exposed as Web services. Web services support also allows vendors with different application products in their portfolio to easily build bridges between them, as ProActive School has done, for example, in providing standards-based interoperability between its SASI and ProActive School's Services.

When schools first begin to create and deploy Web-services-based solutions, they often aren't too concerned with issues such as services security and manageability. As the number of services grows, however, schools soon need mechanisms to store and register the services, to ensure their availability, to limit their exposure only to authorized users or applications, to monitor their use and their performance, and to modify them over time as necessary. They also need to begin to establish and apply rules to govern the identification of school-relevant events and the appropriate responses to them. As a result, even schools that start with bounded, point-to-point solutions can benefit greatly by quickly putting in place the registry, education activity monitoring and management mechanisms that will be required as SOA throughout their organizations. IT departments who have begun SOA deployments commonly cite reductions in development costs and increases in development productivity that they attribute to the loosely coupled SOA model. Meanwhile, schools that implement SOA environments internally can more easily leverage the growing collection of external services offered by software-as-a-service providers and other third parties. Most current generation SOA deployments are too new or too limited in scope to have realized the sweeping advantages that can come from creating a modular, highly adaptable IT infrastructure school-wide. The ongoing industry shift toward SOA and dynamic computing may not be as quick to arrive or as glitzy as the rise of the commercial Internet, but its impact could prove as far reaching. As SOA deployments proliferate, schools, their communities and their users will realize wide ranging benefits due to the flexibility of the services-oriented approach, and will increasingly view IT as having a critical and integral role in supporting their overall education strategy. Those schools that prove adept at riding the SOA wave will realize significant advantages over those who are resistant this computing trend. And vendors such as ProActive School, with its Suite of Education Services, are making it increasingly straightforward to begin the SOA journey.