



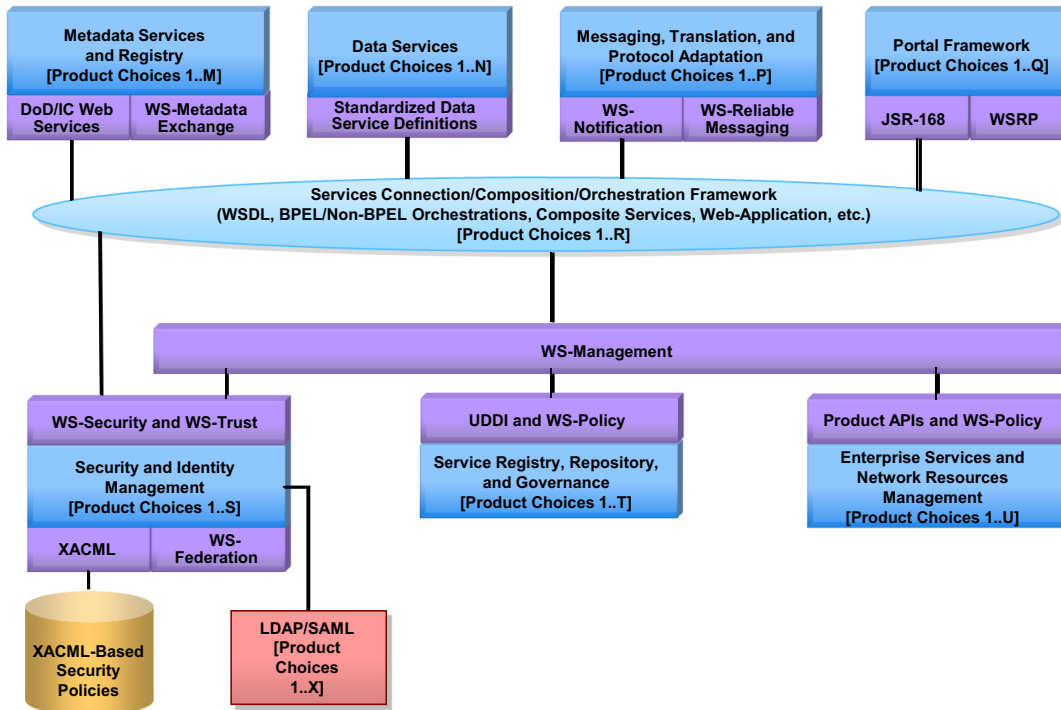
Choose Your Next SOA Implementation Step Carefully

In our interactions with Government Agencies and Fortune 2000 companies, two central messages are becoming clear. First, many organizations began accepting a vendor marketing message in approximately the 2003 timeframe. This marketing message focused on how implementing an Enterprise Service Bus (ESB) was the path to achieving successful Service-Oriented Architecture (SOA) implementations. In some cases, the marketing message went as far as to say SOA = ESB. While an ESB certainly provides useful capabilities for a successful SOA implementation, it doesn't provide all required capabilities. Some ESB products are also based on a vendor proprietary implementation strategy that makes it difficult for the ESB to interoperate with other vendor SOA infrastructure products. Second, in an environment where mergers and acquisitions (from both a business and product vendor perspective) are commonplace, multi-vendor product interoperability is an important aspect of IT strategies and budgets. Few companies can afford to replace major portions of their IT infrastructure because one division uses different vendor SOA products than another division. These budget pressures are placing an increasing emphasis on multi-vendor product interoperability and federation of SOA capabilities across the enterprise.

For organizations that have implemented an ESB but no other SOA infrastructure capabilities or have made large investments in a multi-vendor environment, the essential question is "what is my next investment to increase our overall SOA capability?" Answering this question wisely is very important because a wrong decision could be very difficult and expensive to correct later. So what are the parameters that allow an organization to choose their next step wisely? The best way to answer this is to start with a reference architecture. Simply stated, a reference architecture is a template solution for an architecture that consists of a set of common (or commodity) capabilities and their interfaces. The key phrase in this definition is common capabilities and their interfaces. For a SOA infrastructure, example common capabilities are: metadata services, data services, security, service registry/repository, enterprise resource governance and management, messaging, data/protocol adaptation, and orchestration. Regarding the interfaces, there are essentially three ways to approach them.

- 1. They can be defined by product vendors. However, this typically leads to unique definitions and a lack of interoperability with the interfaces provided by other vendor products.**
- 2. They can be defined by your organization. However, this approach also typically leads to unique definitions and a lack of interoperability with the interfaces provided by other products.**
- 3. They can be defined by open standards bodies such as OASIS or W3C. Many of these standards exist today, and they are commonly referred to as the WS-* standards. Essentially, these standards allow for the specification of a highly reusable, web services based, and composable reference architecture that promotes platform independence, data sharing, and multi-vendor product interoperability. In other words, interoperability and federation are treated as a composition pattern rather than as an off-the-shelf vendor product solution.**

Open Standards offer a wise next-step strategy. The diagram below is a visualization of an example open standards based SOA Reference Architecture using standards from OASIS, W3C, DMTF, and others. Note that other reference architecture diagrams are also possible based on your organization's needs.



Open Standards Based Reference Architecture Diagram

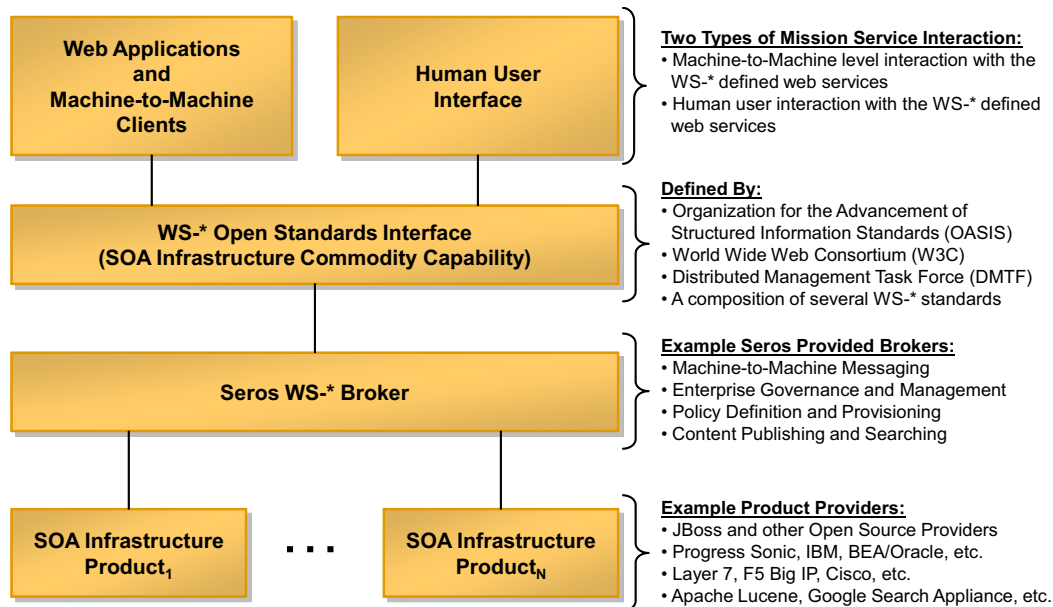
The boxes in the diagram are representative of the common (or commodity) SOA Infrastructure capabilities required by a highly robust, enterprise SOA infrastructure implementation. By using the WS-* open standards, (e.g., WS-Metadata Exchange, WS-Security, WS-Management, etc.) an open standards based reference architecture is achieved that provides interoperability in a multi-vendor environment. Thus, a high degree of reuse is obtained and vendor lock-in is prevented. This characteristic is very important to enterprise SOA implementations for six reasons:

1. It allows for the definition of enterprise level design patterns that specify how to perform standardized integration and implementation of multiple vendors and/or open source products. In the ideal case, the design pattern utilizes both the WS-* based machine-to-machine interfaces as well as JSR-168/WSRP compliant portlets. Thus, both machine-to-machine and human user interfaces are provided. Additionally, whenever possible, the portlets perform all interactions with the backend vendor and/or open source products via the WS-* web services. In many cases, the JSR-168/WSRP portlets also provide access to user interfaces provided by the vendor and/or open source products. This type of design pattern approach is especially important when a heterogeneous set of multi-vendor products is in use across multiple groups/divisions within an organization.
2. It provides the proper standards and web services mechanisms to enable reuse and federation across multi-vendor and multiple SOA implementations.
3. It provides the proper standards and web services mechanisms to incrementally add the next SOA infrastructure capability without creating a vendor lock-in situation.
4. It provides web services for data exchange utilizing data formats based on open standards such as XML.
5. It provides a web services based approach to both design and runtime governance using WS-Management. This is especially important for the runtime governance because both web services and other network resources need to be monitored and managed in a consistent manner.
6. It provides a solid basis for the remaining system engineering considerations that need to be accomplished as the implementation architecture is created. These considerations include: network traffic monitoring, routing, and management; load balancing and network bandwidth; enterprise level governance and management; data consistency, data caching, and/or data replication between

distributed nodes; error logging and management; event logging and management; and data services and storage.

Seros believes, as do a growing number of industry experts, that an open standards based reference architecture is the foundation for enabling an agile SOA environment that meets the needs of the dynamically changing business environment. Many Fortune 2000 companies along with Government Agencies are building such a blueprint with either internal or consulting resources. This approach allows your organization to focus on providing SOA infrastructure as commodity capabilities accessible via standardized interfaces across your enterprise. In this way, your next SOA step as well as specific vendor and/or open source product choices can be made much more rationally. These choices will now align with your organizational business and system engineering objectives that could include driving down the overall cost of IT infrastructure. In fact, you will be “choosing wisely” because you have taken large steps to avoid specific vendor/product lock-in while remaining flexible to changes in the vendor/open source market place, technology, and underlying software platforms. Ultimately, you will have made the wise decisions that are critically important to obtaining the business leverage and return on investment a SOA foundation provides.

Seros has already implemented many of the capabilities shown in the Open Standards Based Reference Architecture Diagram with a set of vendor and open source products. A simple representation of the Seros solution approach and design pattern is shown in the diagram below. This diagram visualizes the concepts described above and a highly cost effective approach for achieving the desired results. The Seros solutions can be provided at a fraction of the cost and time your organization will invest to create this same capability. Give us a call at 703-841-5977 to request a demonstration or obtain more information.



Seros Solution Approach Diagram

To learn more about Seros, please visit us on the Web: www.seros.com.

For more information, please contact:

Dr. Paul Bailor
 14900 Conference Center Dr., Suite 400
 Chantilly, VA 20151
 703.480.2451
paul.bailor@seros.com