Semantic Service Oriented Architecture:
White Paper Overview

Sam Chance
chances@saic.com

Executive Summary

Today's intelligence analysts work around the clock to gather, compile, and analyze vast amounts of data to identify and mitigate potential threats. They engage hostile actors across a spectrum from near-real time activities to strategic, long-term trends. Because of the critical nature of their efforts, there exists a need for analysts to collaborate effectively and employ all the tools at their disposal. Unfortunately, many tools they require fail to interoperate seamlessly – forcing analysts to spend valuable time attempting to perform non-analytical tasks, such as data manipulation. This situation, combined with challenges in sharing information across agencies, and demands for actionable intelligence often results in conjecture offered to decision makers. Meanwhile, our adversaries, more organic and agile, enjoy a smaller decision cycle. As long as this condition persists, we suffer a disadvantage. SSOA will provide a solution to allow analysts to focus centrally on analytical tasks with a viable approach to enable effective capability sharing and collaboration among disparate organizations.

Operational Summary

Operationally, SSOA will present a dramatic effect on workflows analysts execute. SSOA will transform analysts' current tradecrafts from human-centric, manually intensive processes to automated, adaptive and intelligent processes. Analysts will pose objectives to the enterprise and intelligent software will automatically assemble workflows to accomplish analysts' goals. Analysts will themselves be resources, or services, on the network, and will collaborate and share business functions dynamically. Analysts' time will center on higher order, cognitive reasoning tasks, yielding more productive analysis; thus outputting actionable intelligence and enabling superior decisions.

Technical Summary

SSOA will instantiate a robust decentralized architecture based on proven distributed computing models and standard protocols as appropriate. SSOA will allow for secure capabilities (i.e. services) sharing across organizations – be it inter-departmental, -divisional or, as intended, across DoD agencies themselves.

SSOA will provide rigorous descriptions of services and workflows allowing the automated detection of new services of interest and recommend substitute services when preferred services become unavailable. For example, if an analyst has to monitor the elements contributing to an event, such as the launch of a new vehicle, then the information provided to that analyst can dynamically adjust based on the assessment of the factors contributing to that event.

The underlying SSOA architecture will monitor and manage the health status of every service. Self-healing, self-organizing, and self-synchronizing, the system prevents "stale" references to services or "page not found" errors. SSOA implements spontaneous networking, dynamic service discovery and automatic failover functions.

SSOA features a highly scalable decentralized and distributed federated architecture; therefore, outages or failures in parts of the network do not necessarily impact others. If a portion of the federation is down, applications may continue to execute unhindered.

The SSOA infrastructure is founded on an organic and semantically enriched model. Empowering analysts to create ad hoc, "situation based' applications is best realized by a hierarchical model, patterned after complex adaptive systems (CAS). Biological organisms observed in nature provide a tangible example of CAS. Pervasive machine interpretable descriptions of resources will greatly improve our ability to assemble software components into useful applications using automation.

Technical Underpinnings

SSOA is technically founded on three notions: (1) the principals of Service-Oriented Architecture (SOA), (2) Standards Based Design (SBD), and (3) semantics based computing. SSOA combines and implements these computer science concepts into a robust,

extendable architectural capable of enabling complex, powerful functions:

SOA enables heterogeneous, componentized, and distributed applications to work together seamlessly (where seamlessly indicates that participating services are made available using well known, standard interfaces; thus precluding implementation of multiple, ad hoc specifications). SOA is both a process and architectural mindset that focuses on organizing systems as reusable components, rather than fixed processes. SOA helps break up stovepipes and facilitates cross-domain sharing of enterprise resources.

Standards Based Design adds durability, rapid implementation of technologies, and platform independent application specification. Given that heterogeneity is pervasive and will remain, the logical conclusion is an approach that enables portability and interoperability now and in the future. SSOA seeks to implement and contribute to standards from major standards bodies such as the World Wide Web Consortium (W3C), International Standards Organization (ISO), Object Management Group (OMG), OSGi™)[1] and Service Component Architecture (SCA)[2]. OSGi™ defines a standard to change application components dynamically with no downtime, software lifecycle management, and interoperability of applications and services. SCA helps avoid "SOA Vendor Lock-in," specifies a language independent model for creation of services in an SOA, and a service assembly Model detailing the "wiring" of components. A workflow definition, such as UML with DoD domain mark-up (UPDM), provides a potential standard mechanism for expressing service orchestration, empowering analysts to create ad hoc, situational applications.

The current World Wide Web (Web) is largely human consumable, and optimized for presentation. In this [data] space humans exchange and interpret data. In the Semantic Web [information] space, content residing in documents, portions of documents, or other mediums is described by explicit relationships between domain entities/concepts creating machine interpretable content. The Semantic Web then connects machine interpretable content available from distributed, independent contributing communities forming the Web of understanding or the Semantic Web. Availability of semantic content allows efficient aggregation, from which machine interpretable knowledge and understanding can ultimately be constructed.[3] SSOA will leverage standards such as Resource Description Framework (RDF) and Web Ontology Language (OWL) to describe resources, including components, tracked entities, knowledge objects, services, processes – and personnel.

A Distributed Service Framework technology used in conjunction with OSGi, SCA and other technologies will be used as a basis for the SSOA's internal service remoting, scaling and self-healing capabilities. Underlying services fortified with the machine-interpretable descriptions (i.e. semantics) will enable a new generation of composite, adaptive applications heretofore unseen.

Phased Implementation

SSOA is envisaged as a three-phase implementation allowing progressively more sophisticated functionality, and reduced complexity confronting human analysts. Phase 1 will demonstrate analyst interaction with workflows and provide the basis for future automation. Phase 2 will provide described workflows, wherein services are described semantically and invoked based on their 'type' of service. Phase 3 will result in dynamic composition and execution of workflows based on user defined objectives.

Value Proposition

A scalable, decentralized distributed architecture, SSOA is powered by Commercial-off-the-Shelf and Open Source software solutions which combine component-based service representation and discovery, semantic service descriptions, and complex event processing to discover relevant information sources and to trigger queries and actions among those services.

SSOA brings significant value through its ability to:

- Enable service location independence by supporting spontaneously networked services
- Demonstrate the ability to share resources across disparate organizations dynamically
- Enable interoperability across services through a technology agnostic approach to implementation
- Provide information to analysts dynamically based on events
- Allow intelligent software processes to adapt workflows by invoking substitute services when primary services become unavailable
- Provide automated, state-based recovery from failures or network outages
- Push relevant services to end-users based on permissions and roles/contexts
- Facilitate analyst preference learning, and capture best practices by observing actions and providing predictive capabilities
- Employs industry strength security mechanisms, including those for distributed computing.

[1] At inception, OSGi was an acronym: Open Services Gateway initiative. It has since been deemed a term by the managing consortium See www.osgi.org.

[2] See www.osoa.org.

[3] Daconta, M., Obrst, L., Smith, K. (2003). The Semantic Web: A Guide to the Future of XML, Web Services and Knowledge Management. Indianapolis, Indiana: Wiley Publishing Inc.