



Industry Best Practices in Achieving Service Oriented Architecture (SOA)

A Report of the

Net-Centric Operations Industry Forum (NCOIF)
Data Sharing and Services Strategy Working Group

April 22, 2005



Association for Enterprise Integration
2111 Wilson Boulevard, Suite 400
Arlington, Virginia 22201

This report is a product of industry collaboration under the Net-Centric Operations Industry Forum (NCOIF). Distribution of this report is not limited or restricted. The NCOIF operates under the governance of AFEI. Participation in NCOIF working groups is open to those with legitimate interest in assisting the Defense Department with the development of net-centric policies and implementation strategies that will shape the business environment as the Department evolves and transforms itself.

For questions or comment please contact Best Lauer at AFEI (blauer@afei.org).

Acknowledgements

AFEI wishes to thank the following individuals for their efforts in completing this report:

- Mr. Greg Gardner, Oracle, Chairman of the NCOIF Data Sharing & Services Strategy Working Group
- Mrs Joan Baumstarck, EDS, Vice-Chairman of the NCOIF Data Sharing & Services Strategy Working Group
- Mrs. Patricia Segó, Unisys, Data Sharing & Services Strategy Working Group

AFEI also thanks the following organizations who contributed to this report:

Absolute Computer Tech	Lockheed Martin
BAE SYSTEMS	McDonald Bradley
Booz Allen Hamilton	Metamatrix
Battelle Memorial Institute	Microsoft
Boeing	Mitre
CACI	Northrop Grumman
CISCO	Oracle
Data Systems Analysts, Inc.	Raytheon
DNC	Reactivity
Eagan McAllister Associates	Rockwell-Collins
EDS	Sun Microsystems
EMSolutions	SIGABA
Forrester Research	SRA
IBM	Systinet
Institute for Defense Analysis	Titan
Intelligent Decisions Inc	Unisys
Graves Corner Group	Weblayers
Green Hills Software	Westbridge Technology

Contents

- Executive Summary 1
- 1. Introduction..... 2
 - 1.a NCOIF Background 2
 - 1.b Terminology..... 3
 - 1.c Net-Centric Operations and Warfare in DoD..... 4
 - 1.d Industry Views and Parallels of Net-Centric Operations..... 6
 - 1.e SOA, Web Services, and Standard Languages 7
 - 1.f Industry Views of SOA 9
 - 1.g Conclusions..... 10
- 2. Industry Best Practices in Achieving Service Oriented Architecture 12
 - 2.a Vision and Leadership..... 12
 - 2.b Policy and Security 14
 - 2.c Strategy and Roadmap Development..... 16
 - 2.d Acquisition and Governance 18
 - 2.e Implementation and Operations 21
- 3. Conclusion and Next Steps 23
- Appendix 1: SOA Best Business Practices List..... 24
- Appendix 2: Business Service Lifecycle Planner 26

Executive Summary

This document was developed under the Net-Centric Operations Industry Forum's (NCOIF or "the Forum") charter to provide industry advisory services to the Department of Defense (DoD), Chief Information Officer (CIO). It presents a list of industry best practices in achieving Service Oriented Architecture (SOA). The introduction lays the groundwork for the discussion by establishing terminology and definitions and relates this topic to the DoD transformation challenge of net-centricity. The best practices listed in the body of the document are the result of numerous industry inputs and analysis of public sources. The Forum will gladly share any of those source documents upon request, but framed this paper in terms of a vendor-neutral dialogue. The wealth of information on SOA coupled with the need to publish the Forum's findings within a reasonable timeframe suggest that this report is simply framing the starting point for continuing assessment. The rapidly developing nature of both technology and best practices provides fertile ground for continuing this effort in a regular, logically organized pattern. Moreover, the initial findings indicate much more work is need that will fall into the area of other working groups, such as Information Assurance, giving opportunity for collaboration and discovery of intersections and adjacent possibilities. In sum, this document represents the first iteration of a conversation, and is neither a complete nor exhaustive coverage of the evolving subject of SOA.

1. Introduction

1.a NCOIF Background

The charter of the Net-Centric Operations Industry Forum (NCOIF or “the Forum”) includes the following missions:

1. Support migration to open business model that supports full competition but enables horizontal integration of the resulting capabilities or systems, regardless of the provider of those systems.
2. Review and comment on industry-wide frameworks that will support horizontal integration of platforms and systems.
3. Provide an industry advisory service for the Department of Defense (DoD or “the Department,”) Chief Information Officer (CIO) regarding net-centric strategies, programs, acquisitions, implementations and services.
4. Provide industry-wide critiques and analyses in response to government stakeholders.
5. Provide a forum for industry discussion and collaboration on evolving enterprise services models.

Members of the Forum met with DoD Networks and Information Integration (NII) representatives in January 2005 to define priorities for the execution of these missions in support of the Department. The results of the meeting include the following priorities:

- Recommend acquisition models that DoD could use to acquire services and for industry to provide services.
- Explore the role of information technology (IT) integrators and vendors in a Service Oriented Architecture (SOA) environment.
- Provide industry input on best commercial practices, service environment business models, internal industry practices, and applicability of those practices and models to DoD.
- Address the interest, risk, liabilities, advantages and disadvantages of industry operation of Global Information Grid (GIG) Enterprise Services (GIG ES).
- Review lessons learned from managed service efforts and industry business cases.

Noting that Gartner estimates that by 2008, more than 60 percent of enterprises will use SOA as a “guiding principle” when creating mission-critical applications and processes,¹ the Forum concluded that initial inputs to DoD would focus on what the NCOIF found to be the best practices for SOA.

In an attempt to focus the paper on best practices, the Forum has limited the technical documentation to basic explanations of SOA and services, such as web services. Thus, this document does not provide technical tables, illustrations, or results. Further, the Forum believes that topics such as adaptive protocol control, Internet Protocol Version 6 (IPv6), detachable

¹ Todd Datz, “What you need to know about Service Oriented Architectures,” CIO Magazine, 15 Jan. 2004 <<http://www.cio.com/archive/011504/soa.html>>.

software, provisioning capabilities, bandwidth reservation, trade studies, and corporate research are outside the scope of initial inputs to DoD. However, these topics are fertile areas for further explanation and elaboration in follow on documentation.

Before providing SOA best practices, this paper will first provide background information about net-centricity, network-centric operations, SOA, and web services. The document will begin by establishing clear terminology, as one challenge in the areas of transformation, net-centricity, web services, and SOA is the lack of a common lexicon. The terminology that this paper will provide will be based upon DoD definitions and concepts. Next, this paper will give the background of net-centricity and network-centric warfare in DoD, followed by industry views and parallels of network-centric operations. Further, this document will provide industry-standard definitions of SOA and web services, including benefits and issues of SOA, as well as provide industry views.

Following the background information, this paper will address best practices in achieving SOA in the Department.

1.b Terminology

The Department defines net-centricity, network-centric warfare, and Service Oriented Computing as described below:

1.b.1 Net-Centricity

Net-centricity is a robust, globally interconnected network environment (including infrastructure, systems, processes, and people) in which data is shared timely and seamlessly among users, applications, and platforms. Net-centricity enables substantially improved military situational awareness and significantly shortened decision making cycles. Net-centric capabilities enable network-centric operations (NCO) and network-centric warfare (NCW).²

1.b.2 Network-Centric Warfare

Network-centric warfare (NCW) is an information superiority-enabled concept of operations that generates increased combat power by networking sensors, decision makers, and shooters to achieve shared awareness, increased speed of command, higher tempo of operations, greater lethality, increased survivability, and a degree of self-synchronization. In essence, NCW translates information superiority into combat power by effectively linking knowledgeable entities in the battle space.³

1.b.3 Service Oriented Computing

The latest draft revision of the Department's Net-Centric Operations and Warfare Reference Model (NCOW RM) provides the following detailed elaboration of Service Oriented Computing:⁴

² "Data Sharing in a Net-Centric Department of Defense," Department of Defense Directive (DoDD), Number 8320.2, 2 Dec. 2004 <http://www.fas.org/irp/doddir/dod/d8320_2.pdf>.

³ "Data Sharing in a Net-Centric Department of Defense," Department of Defense Directive (DoDD), Number 8320.2, 2 Dec. 2004 <http://www.fas.org/irp/doddir/dod/d8320_2.pdf>.

⁴ Net-Centric Operations and Warfare Reference Model (NCOW RM), v1.1, Coordination Draft, Appendix D, Target Technical View (TTV).

1.b.3.a. Information Technologies

The Service Oriented Computing category includes current web services technologies as well as service oriented architecture concepts such as loose coupling interactions between providers and consumers that enable functional changes to one without impacting the other. Coarse-grained exchanges of information to improve performance are also included in this category. Extensions to SOA address business processing, semantic web, workflow management, content management, peer-to-peer service, and portal services.

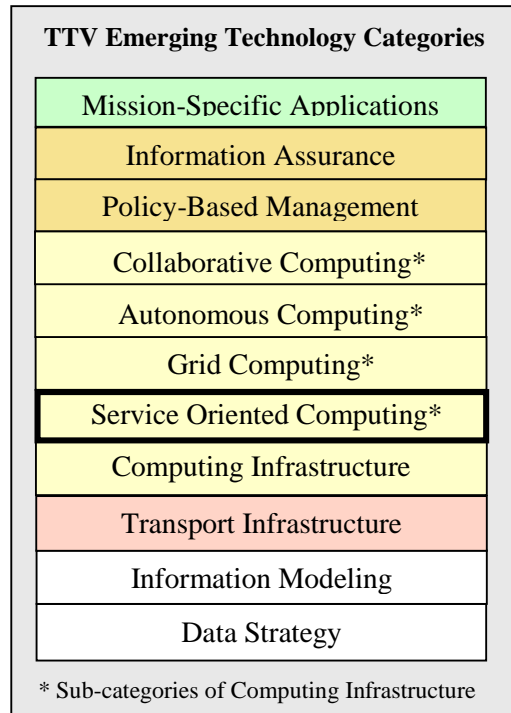


Figure 1: Service Oriented Computing

1.b.3.b Service Oriented Architecture

Service Oriented Architecture (SOA) is an approach to enterprise software applications that considers software resources as services available and discoverable on a network. Such services provide functionality to the enterprise while hiding the underlying implementation details. Providers of these services must be able to publish information about them in a service registry, where service consumers can look up the services they need and then retrieve the information they need about those services to bind to them. This “publish-find-bind” triangle forms the core of SOA. SOA addresses the complexity, inflexibility, and brittleness issues of existing approaches to integration.

Figure 1 shows the emerging technologies of the NCOW RM Target Technical View. Service Oriented Computing is one of the four components of the NCOW RM computing infrastructure.

1.c Net-Centric Operations and Warfare in DoD⁵

A series of books on NCW, written by Dr. David S. Alberts and others and published by DoD’s Command and Control Research Program (CCRP), assisted in the derivation of the concept of net-centricity. Titles of Alberts’ works include: *Network Centric Warfare*, *Understanding Information Age Warfare*, and *Power to the Edge*. These books provide a detailed description and explanation of the concept of net-centricity as applied to military operations.

Many of Alberts’ books are written from a warfighter perspective and use the term NCW, but the concept for net-centricity is applicable to all DoD mission areas (Warfighter, Business, Intelligence, and Enterprise Information Environment (EIE) Management). Familiarity with net-centric concepts is essential to understanding the description of the GIG’s target EIE. Net-

⁵ In the Department of Defense (DoD), Office of the Assistant Secretary of Defense for Network and Information Integration (ASD (NII))’s Global Information Grid (GIG) Architecture v2, architecture developers depicted Net-Centric Operations and Warfare (NCOW) at multiple levels of war and decision making. They began their work with a review of the evolving concepts of net-centricity, net-centric operations, and net-centric warfare. A detailed description of NCOW appears in: Net-Centric Operations and Warfare Reference Model (NCOW RM), v1.1, Coordination Draft, Appendix D, Target Technical View (TTV).

centricity is a concept that has been applied to describe the future operational environment of the GIG. This future model is called: Net-Centric Operations and Warfare.

1.c.1 Net-Centricity and the NCOW RM

The NCOW RM embodies net-centricity, enables a shared perspective of the target EIE, and assists decision makers in arriving at decisions that promote enterprise-wide unity of effort in achieving net-centric operational capabilities. Moreover, the NCOW RM produces a well-established, common taxonomy and vocabulary that is essential for DoD as it moves forward in developing, describing, and discussing net-centric operations and warfare.

The NCOW RM is focused on achieving net-centricity, which is attained through the transformation to a net-centric environment. This transformation requires the satisfaction of four key features: reach, richness, agility, and assurance.

- **Reach** is operationally defined in terms of space-time where distance is not a factor. However, reach recognizes that the integration of spatially disconnected capabilities costs time (i.e., there is a minimum delivery time). Time is the dominant limitation in success.
- **Richness** is operationally defined in terms of the total set of expertise, information, and/or capabilities that can be brought to bear, within a unit of time, to affect a decision or an action subsequent to a decision. Richness contributes to reducing the margin of uncertainty in a decision or action.
- **Agility** is operationally defined in terms of the number of effective adaptations that can be accomplished per unit of time. Thus, highly agile capabilities are those that can anticipate or react and successfully adapt to changes in the environment faster than less agile capabilities.
- **Assurance** is operationally defined in terms of achieving expected levels of operational and systems performance within a specified context, including an adversarial force in a specified timeframe. Adversarial force (i.e., counters to assurance) is measured in terms of work-factors (time to accomplish a condition or effect) and probabilities (likelihood of occurrence). Assurance should:
 - Provide the capability to deter an adversarial force.
 - Prevent an adversarial force from succeeding within a specified time and/or detect an adversarial force in time to provide mitigating responses to counter such a force application.
 - Provide the capability to recover in a timely fashion from an adversarial force, given that the application of such a force has succeeded to some degree.

Additionally, assurance can be directly related to the time-value of mission operations. That is, the time-value related to a mission may be assessed through the following types of questions:

- Can the mission succeed within the resources/unit time expected?
- Can mission performers respond to operational and systems failures and still succeed within some time boundary?

- Can operational or system resources be reconstituted, upon catastrophic failure, in time to still enable mission success?

1.c.2 Net-Centricity and the Target EIE

Alberts, et al, define net-centric in an operational warfighting sense; the NCOW RM adds specificity for its own purposes. The NCOW RM defines net-centricity for the GIG's target EIE as requiring the realization of a robust, globally interconnected EIE (including infrastructure, systems, processes, people, and data). Users are empowered to better protect assets; more effectively exploit information; more efficiently use resources; and unify military forces by supporting extended, collaborative communities to focus on the mission.

The realization of net-centricity continues to evolve as more and more organizational entities, users, processes, functional capabilities, and data become interconnected. Net-centricity gains its power through the ability to leverage and re-use data, services, and processes across functions, domains, and organizations. Efficiency, without loss of effectiveness, in the establishment, modification, and use of data, emerges within this concept. Interdependent operations can also be accomplished faster and with greater efficiency, and, in general, greater effectiveness. Finally, the distribution and tempo of decisions and resulting actions across functions can increase.

1.d Industry Views and Parallels of Net-Centric Operations

1.d.1 Industry Views

Underlying Task/Post/Process/Use (TPPU) and smart pull paradigms⁶ is the notion of information "discoverability" on the network or the ability to dynamically bind to or access services that make the information available to users and applications outside the traditional system boundary of the systems that collected or produced the information. Discoverability allows detection of and access to data services without regard to the user's or using application's local execution environment. (The likelihood is small that every user and system in DoD will be executing on a common platform environment.) DoD is part of multiple extended, virtual enterprises such as coalition operations, alliances, logistics supply chains, homeland security, and the like, all of which may need to operate in a net-centric fashion with some DoD-specific systems.

Another way of looking at the concept of net-centricity is that it inherently crosses traditional enterprise boundaries and operational contexts: Net-centricity exposes internal information models and services to entities previously viewed as external to system boundaries. As a result, the focus has shifted from attempting to create common execution environments to attempting to reach agreement on the representation of the "battle space" or business environment containing the services and information. Agreement on the representation of an environment makes discoverability by users and other applications more achievable and less subject to misinterpretation. Evidence of these improvements is appearing in the e-business world where new business languages and information model standards spring up seemingly every week in order to leverage the network accessibility of services.

⁶ For detailed information about TPPU and smart pull paradigms, please refer to the following article: John Stenbit, "Horizontal Fusion: Enabling Net Centric Operations and Warfare," Crosstalk: The Journal of Defense Software Engineering, (Jan. 2004) 4.

Despite the ability to cross traditional boundaries, net-centricity has funneled the realization that different operational contexts or functional domains have inherent needs for representing the business environment in different ways and with different frames of reference. Thus, net-centricity creates a growing need for mediation services to help users and applications access information that might be captured and represented in a way with which they are not familiar. This is also the driver behind the notion of the “Semantic Web.” An important aspect of dissimilar information representation is the concept of naming authorities for entities that are represented in the services offered on the network by the various domains and Communities of Interest (COI). Another important aspect is identifying ways to map between these different identity attributes. As an example, a tracking number for some entity in a given service might correspond to a mission number in another service or an asset ID in yet another service dealing with that same entity (e.g., a particular aircraft).

Of course, security and privacy concerns still remain in the net-centric world. Indeed, security and privacy concerns have recently become more significant because data is potentially accessible to anyone on the network. To maintain security and privacy, a net-centric environment emphasizes data tagging with security labels and federated trust approaches for determining who has access to what information. Further, these methods will require service interfaces to be security aware to a degree not seen before, and these methods will require network-level security services that are just starting to emerge. In short, local user account management approaches or system boundary perimeter access controls are not reliable in a net-centric environment.

Lastly, the net-centric concept implies persistence and ubiquity of the network. Unlike many systems today, the network cannot be “turned off,” even for very short periods, without major operational impacts. This means that services on the network need to be designed from the outset to be evolutionary and aware of their own possible replacement. In many cases, multiple versions of the same service may need to exist simultaneously on the network to support gradual evolution of other applications and services that depend on those previous versions. Similarly, applications that use services need to incorporate more install time and run-time “discovery” code to “sense” what services and service versions are present on the network and to use decision logic regarding which services or versions to employ in a given situation. People already do this naturally, but even end users will benefit from user interface cues and discovery services that advise them of new services and service versions present on the network

1.d.2 Industry Parallels

Net-centric operations are, therefore, similar in many ways to the enterprise service environments found in industry. SOA underpins these environments. Net-centricity both enables and, when applied across enterprise boundaries, requires SOA concepts to be applied in order to achieve flexible and adaptive operational effectiveness.

1.e SOA, Web Services, and Standard Languages

1.e.1 Background

One of the methods of implementing SOA is through the use of web services. A typical web service is a software component designed to support interoperable machine-to-machine interactions over a network. It has an interface described in a standard format (the format is Web Services Description Language (WSDL)). Other systems interact with the service in a manner

prescribed by its description using Simple Object Access Protocol (SOAP) messages, typically conveyed using HyperText Markup Language (HTML) with an Extensible Markup Language (XML) serialization in conjunction with other web-related standards.

SOA does not require web services, and web services can be deployed without a consistent or universal SOA. Many experts believe, however, that building web services as part of a larger SOA strategy, while certainly not the only approach, is one of the most useful.⁷ Many organizations begin with web services and then transition to more effective and manageable “business services” with the application of policy articulation, business rules, and process control.

Implementing business processes is the ultimate goal of SOA. To achieve business processes, a programming language such as Java or C# can orchestrate the invocation of services and the business workflow. However, in order to achieve the greater goals of agility, visibility and interoperability, a standard language that supports the rapid orchestration of services should be employed. This standard language is Business Process Execution Language (BPEL). Among its features, BPEL includes high level constructs to enable synchronous and asynchronous service invocation, parallel activity flows, decision branching, exception management and compensation—a mechanism that enables the undoing of complex, long running, business transactions.

1.e.2 Benefits of SOA

SOA eases the integration of the heterogeneous IT environments found in many organizations through the use of standard protocols, such as web services. Building services with SOAP and WSDL, for example, not only smoothes internal integration processes, but it also facilitates information sharing across organizational boundaries. Further, SOA offers the possibility of taking legacy investments and making them all work smoothly (and more cheaply) together. In fact, SOA reduces development costs since large chunks of SOA-based components are reusable. Moreover, identifying and leveraging the capabilities of existing systems maximizes the value of IT investments while minimizing risks.

Another benefit of SOA is that it forces IT workers to think in terms of dynamic operational needs—not simply a technical implementation of static requirements—and, therefore, requires leaders to focus on the best ways to improve operations. By exposing and sharing information across once-siloed applications, organizations can extract more business performance data in real time, improving business intelligence and increasing responsiveness.

Finally, since the benefits of SOA facilitate easier integration and increased agility, implementing SOA can lead to a greater return on investment (ROI). Services, their data, and interfaces can be generic enough that they can work with an array of front-facing systems, and

⁷ Todd Datz, “What you need to know about Service Oriented Architectures,” CIO Magazine, 15 Jan. 2004 <<http://www.cio.com/archive/011504/soa.html>>.

new technologies can be easily incorporated into SOA, reducing risk and expense while speeding development of new applications.⁸

1.e.3 SOA Issues

SOA does present some challenges, especially when the scope is cross-enterprise and the performance requirements are high. SOA has inherently more run-time overhead and transactional latency than equivalent functionality implemented through tightly coupled architectures (although the latter are not immune from bad design). This is one reason for the dictum to make services coarse-grained. Additionally, a net-centric SOA can be a challenge to implement when applied to real time requirements or strict flow-control rules, and some operational performance domains may not be addressable with current network constraints and available technologies (e.g. weapon system guidance controls).

Although SOA can integrate heterogeneous environments, it does not address the issue of different information models operating in different enterprises or operational domains. SOA also does not address the related issue of different identity or naming standards and appropriate authorities for different operational contexts. SOA does, however, make access to such identity and naming services easier. These considerations are driving the current industry push for web services security services, e-business information exchange standards, and federated identity management services and standards.

1.f Industry Views of SOA

Simplistically, SOA is essentially a collection of services. These services communicate with each other, which involves either simple data passing or two or more services coordinating some activity.⁹ SOA is not new; indeed, it has been around for years. Although significant challenges still remain, the recent evolution of both open-standards based technology and implementation processes have enabled radical improvements in SOA capabilities.

Modern SOA dramatically eases integration in heterogeneous environments and provides a transformational enhancement in agility, visibility, consistency, and interoperability. Additionally, integration is achieved as a result of common characteristics within SOA.

Common characteristics of SOA within industry include:

1.f.1 Services have platform or implementation technology-independent interfaces.

Although one can build SOA based on specific platform and technology standards, the concept of net-centricity suggests that such an approach is limited by enterprise scope, platform or technology acquisition constraints as well as life-cycle cost considerations. From an acquisition perspective, DoD operates as multiple enterprises—working and communicating with other federal, commercial and coalition partners. Thus, a platform-neutral and execution environment-independent approach to service interfaces is key to achieving the net-centric vision.

⁸ Todd Datz, "What you need to know about Service Oriented Architectures," *CIO Magazine*, 15 Jan. 2004 <<http://www.cio.com/archive/011504/soa.html>>.

⁹ Doug Barry, "Service-Oriented Architecture (SOA) Definition," <http://www.service-architecture.com/web-services/articles/service-oriented_architecture_soa_definition.html>.

Furthermore, such environmental-independence makes net-centric services easier to reuse and change and guarantees dynamically interoperability.

1.f.2 Services are exposed using standards based interfaces.

Although multiple methods exist to make services network accessible in an execution environment-independent way, adopting commercial standards for specifying and accessing service interfaces is the best approach to achieving a cross-enterprise SOA. Unless a compelling need exists that is not addressed by current net-centric SOA service interface standards, one should not consider creating a new service interface standard.

1.f.3 Services are “Loosely Coupled.”

Open standards are essential, as are the advertisement of the service interfaces in well known and widely accessible “service registries” or discovery services. Access to these registries should be enabled via a set of services available on the network at the virtual enterprise, enterprise, or Community of Interest levels. Explicit versioning in the service interface facilitates loose coupling, which brings change resiliency. Services can change and be versioned while eliminating or at least managing the ripple effect on applications consuming these services.

1.f.4 Services are “Coarse Grained.”

Services focus on high-level business processes using standard interfaces. The coarseness of the grain appropriate to a given service depends to some degree on how that service will be most often used and the likely “network distance” between the service provider and the service consumer. Lower bandwidth and higher latency connections suggest coarser granularity while a high frequency of “single item” requests from diverse sources suggests a need for finer-grained service interfaces.

1.f.5 Services are Modular.

A service represents a discrete unit of business, application, or system functionality. The intent should be to transform basic services into more complex system capabilities to deliver more valuable functionality. By reassembling services into a new configuration, new business services can be created to support a different business objective.

Services that are too fine-grained or incomplete add configuration management and administrative burdens to the service user and to run-time overhead. Services that are too coarse-grained create difficulty in using only the functionality and business rules one needs. Coarse-grained services also create potential for unintentional coupling and unnecessary run-time overhead.

1.g Summary and Transition

In this introductory section, the Forum has provided background information about net-centricity, network-centric operations, SOA, and web services. It established a common lexicon, reviewed DoD and industry views and provided industry-standard definitions of SOA and web services, including benefits of and issues related to SOA. Clearly, there is no lack of information on the subject and much of it evolves almost daily.

In the next section, the Forum will focus on the evolving concepts regarding the management, architecting, systems engineering, governance, and employment strategies of technologies and solutions related to SOA. The Forum has refined as tightly as possible literally hundreds of inputs into a succinct description of best practices. While far from the last word on the subject, these provide a consensus view of the current situation, offer a general set of predictions for the near term future, and lead to a number of guiding concepts and advice.

2. Industry Best Practices in Achieving Service Oriented Architecture

Best practices suggest that successful SOA and web service implementations most often take place within the context of an organizational commitment to operate more efficiently and effectively. Thus, to ensure a successful application of SOA to the Department, the Forum will address SOA within DoD from an organizational perspective.

To address SOA in this manner, the Forum first identified and focused on the “organization” or “organizations” of interest for DoD. The Forum suggests that DoD (as an enterprise) becomes the organizing entity to implement SOA. However, an enterprise implementation of SOA strategies and web service deployments requires extraordinary levels of commitment and organizational “horsepower” to affect enterprise-level changes, consistency, and governance. DoD will be attempting to consolidate SOA, web service solutions, and consistent governance throughout what amounts to an enterprise of already established enterprises. By comparison, industry and individual companies have better control over their smaller individual enterprises and can better affect change and consistency. The Forum appreciates the magnitude of this challenge, but also strongly believes DoD should accept this challenge.

This review of industry best practices is divided into five parts:

- Vision and Leadership
- Policy and Security
- Strategy and Roadmap Development
- Governance and Acquisition
- Implementation and Operations.

Each section is examined in detail. Since most of the topics are broader than specific cases, only certain parts address specific industry cases.

2.a Vision and Leadership

The decision to implement SOA in an organization requires an extraordinary commitment from senior leadership. Senior leaders must articulate the vision for the effectiveness desired from a web-based approach to information sharing as well as the value of moving beyond simple process automation to the ability to rigorously answer key business questions in real time. More importantly, leaders must anticipate and aggressively attack cultural resistance to the availability and sharing of information throughout their enterprise, and promote the value that consolidation and self-service enablement brings. This requires clear, consistent evangelizing and messaging. Best practices in this area include:

2.a.1 Evangelize the benefits of net-centricity, SOA, web services, and transformation.

A common understanding of the transformation goals and high-level business processes, as well as the proper adjustment of personnel roles and responsibilities will facilitate the transformation. An education program is key, as is developing and implementing a marketing plan.

2.a.2 Think differently.

Net-centricity enables a joint and multinational environment with possible supportive interdependencies that leaders have not previously considered. The traditional development, deployment and governance approaches will not optimize the results of SOA. Leadership must start thinking in terms of business processes; they must start thinking about the information providers, brokers, and consumers, and they must think about networks as weapons systems. Leaders must also think more explicitly about the boundaries of their operational scope boundaries and the implicit assumptions they have made about them.

2.a.3 Actively manage the cultural, strategic, and tactical issues of a major paradigm shift.

For the technological transformation (tactical dimension) to succeed, the strategic focus needs to address the business, organization, people, processes and culture. Further, the focus needs to address how these resources are defined and used in the business model. Leadership must recognize that the technical issues are not the hard part; the real challenge is the socialization of the service vice system approach and the business case for delivering the services. The IT support organizations or vendors, in particular, must be focused around business services and processes.

The Queensland Department of Primary Industries needed to improve knowledge sharing and intranet content publishing productivity for its staff of 4,000. SOA not only significantly reduced the turnaround time for publishing information, but also *put the responsibility for published information in the business groups and regions*. Templates ensured that the end users gained required knowledge. (The Queensland Department of Primary Industries example is as much an example of a successful “Data Strategy” as it is SOA. The Data Strategy is a foundational feature of a net-centric SOA.)

2.a.4 Proactively address the cross domain and cross business area issues.

When an organization supports a business model that cuts across organization lines, vertically and horizontally, and when an organization provides for the orchestration of services in support of essential business functions, the organization maximizes the flexibility of SOA. In fact, optimizing resources across organizations and systems enhances collaboration and leverages existing IT investments. At the same time, one must recognize the inherent differences and diversity in operational contexts across domain and enterprise boundaries and develop solutions that accept and deal with these essential differences so that all stakeholders in the enterprise get the support they need.

2.a.5 Team with industry, across military services, and across executive agencies.

Consider the commercial, federal, and multinational interactions and dependencies that DoD could leverage to improve operational effectiveness. The commercial world is grappling with “globalization,” enabled to a considerable degree by the Internet. DoD faces the same dynamics in their operational domain.

2.a.6 Create and document a business case for SOA.

The current environment in both government and industry demands a close examination of investments and project justification. A thorough business case document can help an enterprise acquire funding or approval, reduce resistance, and execute strategy. Business cases can be

strategic or financial, but should include the business case objectives and summary, an examination of alternatives, the financial metrics, supporting arguments, a high-level project schedule and significant milestones, and a discussion of the assumptions and risks. It must also address hard and soft benefits, with both stated in the business context. Further, a business case should address the highest-priority mission goals, whether those are cost savings, competitive advantage, governance, compliance, user experience, or service offerings. The exercise of creating a concise business case will assist the organization to understand the strategic goals, prioritize benefits, socialize the project, gather requirements, predict costs, consider alternatives, and monitor progress. After approval of the business case, the project manager can use the document to keep the project on track (strategically) and within scope.¹⁰

2.b Policy and Security

Once leaders have made the decision to improve business and doctrinal/tactical processes using web services, best practices requires the careful development of an architecture for and taxonomy of those services. The chosen services need to align well within the range and scope of operational architectures that the enterprise envisions supporting. Further, leadership must make decisions about the general standards model(s) and ontology(ies) that will be implemented across the enterprise and within communities of interest. This addresses one of the key issues with SOA—ways to deal with the inherent diversity of representation of the battle space or business landscape in information systems on the network. In addition, leaders need to consider the acquisition model for building such services and incentivizing (or indemnifying) interdependence of systems and services. Finally, senior leaders must carefully determine the organization’s approach to security policies and risk mitigation, items that they then must craft into policy guidance. A blend of a modest amount of top-down direction in key areas, particularly security and acquisition policy, combined with a healthy dose of bottom-up creativity and initiative appears to be the most effective practice.

2.b.1 Establish technical standards.

Dun & Bradstreet (D&B) uses a web-based pipe to its global database of information and analysis on worldwide companies in order to deliver its D&B Global Access Toolkit to its customers. At the time D&B developed the Toolkit, many standards were still in the specification stage. Thus, they designed the Toolkit to be compliant with as many new and emerging standards as possible. Today, an industry partner watches the newest standards closely for Dun & Bradstreet. Another example of a company that is attentive to standards is TSYS Prepaid, a leading provider of prepaid card solutions. TSYS Prepaid recognizes that many standards are constantly evolving, and it, too, uses an industry partner to track standards development.

2.b.2 Establish portfolio management policies and policy/information standards and put them in a standards-based registry.

Policies facilitate the development of ontologies, naming guidelines and services, data standards, and taxonomies. These policies also set the framework for establishing authoritative data sources. When the Logan City Council website found its content management costs spiraling, it

¹⁰ Chris Hadded, “Building the Business Case for Service-Oriented Architecture Investment.” [Application Platform Strategies: Methodologies and Best Practices](#), V1, 30 Mar. 2005 (Midvale, Utah: Burton Group 2005).

resolved the issue of data ownership and authority. Its new web services solution allows groups and businesses to install and manage their own events calendars, eliminating the Council's need to collect, input, de-conflict, and publish the data. Allowing groups and businesses to manage their own events calendars promotes accuracy and data ownership while reducing the Council's infrastructure cost.

2.b.3 Establish application interoperability policy.

The Hartford Financial Services Group, one of the nation's largest investment and insurance companies, had a complex integration problem. Its continually changing versions of its quote request service had to integrate with the numerous and constantly changing carriers who responded to the quote request calls. Hartford used loosely coupled web services and Universal Description, Discovery, and Integration (UDDI) to solve the many-to-many versioning problem of linking its service consumers to the second tier quote providers.

2.b.4 Consider how to benefit from both top-down and bottom-up leadership.

A proactive establishment of control and incentive mechanisms requires strong top-down policy and governance. Further, a top-down approach facilitates security across organizational boundaries. However, user and system owner engagement—through bottom-up leadership—is critical. Both approaches offer important advantages.

2.b.5 Establish governance, security, reuse, compliance, risk management, and versioning policies.

An enterprise implementing SOA needs a proactive top-down policy that facilitates a cross-organizational approach, as most likely the environment is not under the control of a single organization or project. For example, Wells Fargo conducted an audit of the 15 internal IT services providers that support the bank (which is the fifth largest in the U.S.) and found over 700 web services in use, with many more in development. Wells Fargo managed this proliferation of services with a centralized mechanism for web service registration, discovery, and re-use.

2.b.6 Employ multiple security approaches.

TSYS Prepaid's high volumes of financial transactions demanded a robust security framework, including encryption and authentication, for its web services. T-Mobile, a wholly owned subsidiary of Deutsche Telekom, developed adequate security measures for transmitting personal and financial data for its 87 million European subscribers while meeting stringent regulatory requirements.

2.b.7 Ensure security is “baked into the solution.”

King County, Washington implemented an information sharing initiative with other police departments in the region. Because the shared case information is highly sensitive, security is a primary concern. The King County web service returns information through a role-based security model and maintains a full audit trail.

2.b.8 Address SOA-unique security considerations.

Senior leadership needs to consider whether services always execute on behalf of some user or user role or whether a service can act as an autonomous agent acting on behalf of the enterprise

or some community of interest. Additionally, since applications from across the organization access the service applications, identity management and security enforcement that can manage across boundaries is critical. Clearly, a security policy must be enforceable.

Finally, senior leaders need to consider both transport and message level security because multi-hop scenarios, including whole range or intermediaries, are common in SOA. (Some intermediary examples include routers, policy enforcers and business process coordinators.) Transport level security, such as https, is simple (it encrypts headers as well as message bodies), but it stops at the endpoint, whereas message level security allows headers to be decrypted for routing while keeping content secure and private. Message level security also enables message parts to be handled independently, which is key for SOA intermediaries to work—and to work securely.

2.b.9 Plan for disaster recovery, business continuance, and disaster management.

Like security, a plan for disaster recovery, business continuance, and disaster management defines functions and architecture features that are important components of the target-state SOA. In fact, network-centric operations demand business continuance and disaster recovery plans. The Department of Homeland Security's (DHS) Disaster Management Interoperability Services (DMIS) program demonstrates the agile, adaptive nature of SOA and its conduciveness to the concepts of disaster management and collaboration.

2.c Strategy and Roadmap Development

A strategy and implementation roadmap captures the details of the execution of a web-based information sharing and optimization structure. Included in the roadmap are the architectural, structural and definitional details specific to the enterprise, as well as security and risk management considerations. SOA best practices mandate that this key step, the roadmap, evolves concurrently with policy, acquisition and governance. Additionally, best practices suggest that the roadmap is often influenced more by the adoption of a variety of minor implementations, experiments, and demonstrations across the organization than by explicit leadership direction. Best practices in this area include:

2.c.1 Develop, document and publish your SOA strategy.

Leaders must identify business and IT imperatives, along with the targeted business outcomes and SOA metrics, during the early phase of the SOA strategy and planning process.

2.c.2 Plan for incremental transformation and deployment.

SOA is an iterative process requiring incremental transformation and deployment plans. For example, Goulburn Valley Water (GVW), one of Australia's largest regional water authorities, implemented an employee time and attendance system, saving \$70,000 on approximately 180 employees. The system eliminated most errors from the previous manual process and provided better customer service. After this initial success, GVW is now planning to automate other data collection processes, such as the monthly filing of vehicle odometer readings. The key point is that GVW started with a simple first step, automating the data collection process.

2.c.3 Align programs/projects to share services.

Senior leaders should employ supportive efforts, including negotiating common definitions across lines of business, developing shared services against project time and return on investment, and inserting a cross channel (organizational) view into ongoing projects. Leaders should also develop charge back incentives to increase the motivation and charge back mechanisms to ensure proper accounting. Further, leadership should develop and make available service level agreements (SLAs) to facilitate the discovery process.

2.c.4 Maintain a vision of shared services but move toward it opportunistically and incrementally.

The enterprise must add shared services when new requirements develop. Additionally, the enterprise should reduce redundancy by reducing dependency in a fine-grained manner, potentially function by function.

2.c.5 Design for connections, change, and control.

The enterprise should build a topology of services that reflects the business processes, not the systems, thereby giving the enterprise the ability to make changes. Senior leaders should think in terms of how the enterprise business model and SOA will operate in the context of both the business as well as within the IT infrastructure. Sabre Holdings' Travelocity line of business is an example of a business using changeable web services. Because airfare searches are compute-intensive services that calculate millions of schedule possibilities, availability, rules, routes, and fares, and because the data is continuously updated throughout the day, Sabre needed a dynamic, very horizontally scalable architecture and implemented an open-platform server farm with commodity hardware and software.

2.c.6 Create a common vocabulary.

Planning requires the creation of common vocabularies (i.e. taxonomy) to ensure proper understanding and consideration of how to manage, change and use taxonomies. A common vocabulary also facilitates the collaboration and sharing of information across different business areas. Further, as an enterprise defines a common vocabulary, the enterprise should also develop related vocabulary governance. With both, vocabulary mapping between business areas or communities of interest (for example, relating tracks to targets or installations to supply points or units to weapons systems) becomes much more manageable.

2.c.7 Recognize the importance of cross-enterprise architecture.

SOA will define the strategy for common shared services and provide guidelines for cross-channel and cross-enterprise services. The cross-enterprise architecture and adherence to this architecture facilitates the deployment of end-to-end business processes that cut across organizational boundaries and business partners. Moreover, the cross-channel (customer interface) perspective enables a single customer view of the different dealings within the company or organization. Senior leadership should establish a cross-enterprise board for the review and approval of common and cross-channel services and for the review of the full architecture.

2.c.8 Define and enforce application interoperability and business interoperability policies.

The enterprise should use common interoperability standards wherever available. Application

interoperability is typically based on SOAP and WSDL contracts. Policy interoperability should be based on UDDI and should leverage common policy mapping, support for Web Services Policy Framework (WS-Policy), Web Services Policy Attachment (WS-Policy Attachment), and taxonomies.

The enterprise should enforce policy at both design and run-time by ensuring services are not built or deployed that are not compliant and by using platforms when possible and intermediaries as necessary. An example of platform enforcement is using Web Services Security (WS-Security) implementation to enforce the authentication policy. An example of intermediary-based enforcement is using an intermediary to implement a policy that a platform cannot enforce.

Finally, run-time and design time policy enforcement must be synchronized.

2.c.9 Transform your IT development processes and policies.

Processes and policies should address new and existing programs. This will help programs deliver net-centric products.

2.d Acquisition and Governance

Acquisition and governance are two very different yet related processes. The Forum's acquisition analysis revealed that proven processes that work well for the acquisition of stand-alone systems are not sufficiently agile to keep up with the evolution of both technology and broadly accepted standards and processes for SOA. Instead, market-driven models that embrace frequent change, strong involvement with industry and standards bodies, and close ties with internal and external user communities are the most effective acquisition models for SOA.

Governance processes must be similarly adaptive and flexible; what the enterprise needs, what systems the enterprise will build, and how those systems will be built will be much different tomorrow than they are today. Of course, the enterprise must also stay within fiscal constraints. Organizations must have discipline and rigor in the enforcement of the architectures, standards, and policies they adopt for SOA because without rigorous governance, the Department will not realize SOA's potential benefits.

Gartner cautions, "Service-oriented architecture built opportunistically with the purpose of "getting it over with" as soon as possible, and at as low a cost as possible, will prove to be a disaster for enterprises' software infrastructures."¹¹ Accordingly, simplicity, interoperability based on open standards, scalability, and loosely coupled, modular services are keys to an effective governance process in the Department's dynamic environment. (*This initial survey has shown the Forum that government implementation of industry best practices in SOA acquisition and governance may require statutory and regulatory change. This finding requires further study.*) Best practices in this area include:

2.d.1 Incremental acquisition.

Bekins' HomeDirectUSA division—responsible for delivering consumer goods direct to its customers' homes—provides an example of an incremental web services acquisition. As the first

¹¹ Yefim Natis, "Users: Beware of Opportunistic Web Services Projects" 24 Jan. 2003 <<http://www.gartner.com>>.

step in its e-business transformation, Bekins implemented a Java-based application that allows customers to track orders. Next the company implemented an order and inventory management component. Finally, it brought in Tonnage Broadcast Exchange (TBE), which notifies Bekins' third-party vendors of shipping opportunities, giving them a chance to bid for orders in real time.

2.d.2 Use experiments, pilots, and collaborative demos.

The Forum notes that the Defense Information Systems Agency (DISA) and the Office of the Secretary of Defense (OSD) already use experiments and pilots, with Horizontal Fusion and Oktoberfest as excellent examples. Another example is the Commonwealth of Pennsylvania's execution of a pilot for its Pennsylvania Patient Safety Reporting System (PA-PSRS) that involved 22 separate facilities. Pennsylvania then used the feedback from the pilot to prepare the statewide rollout to 422 facilities.

2.d.3 Consider using enterprise modeling.

In many industries, businesses use modeling to identify their business processes and infrastructure as well as design their target architecture. In fact, some industry analysts refer to SOA as a digital model of the physical business processes. Determining what constitutes "the enterprise," defining the scope boundary dimensions and cross-boundary interactions, and exploring the model's limitations are key aspects of the modeling process.

2.d.4 Enforce policies.

Policies should not be left to documentation. Senior leadership should incorporate approval of funding for development and maintenance of web services as part of the enterprise governance cycle and processes, similar to the approval of other types of enterprise application development. Once a web service is used or subscribed to, a SLA mandates that the web service has responsibility for providing its services. Third-party consumers depend upon the service; correct funding and governance of items such as maintenance and operational support are imperative.

Policies are the foundation of governance analysis and auditing for compliance. A SOA compliance policy is a governance policy describing what internal and industry standards will be followed for all services, whether internal or from external providers. A security policy might also specify what security standards and credentialing processes will be enforced during services design and consumption. Further, the policy should bridge the gap between run and design time policy enforcement. Finally, a compliance policy should consider the issue of proper service usage and incentives for providing services as well as avoiding service "abuse," such as excessive and frivolous service requests.

2.d.5 Loosely coupled services requires detailed governance, management, and SLAs.

Loosely coupled technology demands a better framework for building and operating applications explicitly as public services, in terms of SLAs and policy. Governance defines and enforces the compliance rules for managing SOA business services. It also involves mapping corporate, business and IT policies to specific SOA business services and then ensuring the enforcement of the policies by the SOA infrastructure. As a result, a SOA governance model will dictate policies for services reuse, IT compliance, and security. A service use policy also describes the technical and business aspects of using services as well as provides a review process to ensure that existing services are adequately considered and evaluated prior to developing new ones.

2.d.6 Monitor, measure, and analyze the enterprise's SOA service network.

Publishing services in SOA requires appropriate definition of the procedures and policies before making web services available as business services. For example, senior leaders should determine who is allowed to publish a service to the registry, establish the release procedures, and determine the approval and certification process for designs, standards and security policies. These policies and procedures will provide a base for governance. Next, senior leaders should provide a governance process, including portfolio reviews and performance measures, to track IT transformation. The governance process should use the SOA Common Operating Environment (COE) dashboard to monitor execution.

For example, a Commonwealth of Pennsylvania program assures quality by applying processes such as automatic random samples and audits and alert features for new safety incident reports that match specified criteria. Service monitoring also makes possible the enforcement of service use and abuse policies as well as the monitoring of adherence SLAs. Further, service monitoring allows for the implementation of incentive programs for service use to compensate popular service providers for high service demand and use.

2.d.7 Promote Service Discovery and governance using a standards-based registry.

Service Discovery through a standards-based registry mechanism enables web service discovery, life-cycle management and governance. Additionally, Service Discovery provides the central registry for publishing and organizing web services for discovery both at design time and at run-time. Service Discovery is also core to SOA-based solutions—it promotes reusability, minimizes redundant efforts, enables loose coupling through service virtualization, and facilitates the implementation of applications through service composition and orchestration. Service registries, at the program level and at the portfolio or CIO level, provide a roadmap for future federation by providing the ability to promote web services as net-centric enterprise shared services as well as providing the discovery of other net-centric enterprise services. Service Discovery should be based on open standards, such as UDDI.

2.d.8 Consider run-time discovery where appropriate and where it provides business value.

Run-time discovery occurs when a SOA consumer accesses meta-data during use. Run-time discovery is not needed all the time, but it is necessary and effective in a variety of scenarios, such as: when load balancing is required, when a large number of consumers exist, when the enterprise experiences a high rate of change, and when the value of the information is particularly high. Additionally, run-time discovery supports net-centricity and facilitates information exchange in a fast moving (i.e., battlefield) environment. However, leaders must ensure that related policies are explicit and well thought-out.

2.d.9 Promote standards based process models, such as BPEL or Unified Modeling Language, for process model interoperability.

Aligning IT with business processes is the key value proposition of SOA. Process models are the typical articulation of this. The enterprise should treat BPEL, Unified Modeling Language (UML) and other process model artifacts as some of the most critical SOA design artifacts; these process models should be held subject to the same stringent governance requirements as other design artifacts. Further, the enterprise should create process models that the enterprise can share

and version to reflect changing needs. The enterprise should also consider developing new governance process models using BPEL and then monitoring those BPEL-defined business services and processes like any other component of the architecture. While BPEL is not sufficiently expressive for many process requirements, best practice users find scenarios where it can be applied and evolve from there.

2.e Implementation and Operations

This is where the “rubber meets the road” for SOA. Best practices reinforce that effective web services and SOA are implemented incrementally, but rapidly—building and testing each step and then formally “cutting in” the service and moving on to add the next. Hesitation and skepticism typically occurs as services and SOA are implemented, and, in many cases, employees and customers experience a slight dip in the quality of services before the quality recovers and rapidly improves. Leadership is key, as ongoing operations demonstrate the worth of web services and SOA—increased organizational effectiveness with radically improved access to information and collaboration, reduced costs with reusable assets, reduced personnel requirements, and improved customer satisfaction and employee morale. Best practices in this area include:

2.e.1 Implement incrementally following the delivery of business value (benefits).

An enterprise should implement “low-hanging fruit,” followed by the execution of back-end migrations, and then migrate applications to services interfaces. The enterprise should give priority to the incremental change that has the clearest, strongest business value, while recalling that some changes are high-impact because they enable other changes.

The Forum emphasizes the importance of a corporate registry for web services and recommends the following steps for web services implementation: 1) Implement a single (logical) corporate registry. 2) Incorporate funding approval for services as part of the governance process. 3) Have a central team manage the registry.

Here are two examples of organizations implementing SOA in simple, accessible circumstances: One IT firm implemented a content creation and web publishing system thereby automating a formerly manual process resulting in savings of \$180,000 a month and reduced submission time from four hours to 15 minutes. The Ministry of Revenue Quebec grew its on-line tax remittance model step by step. The current version is fairly well evolved and has generated significant time and cost savings.

2.e.2 Partnering and collaborative implementations work best.

Most successful commercial efforts take a collaborative approach to providing the customer’s solution, such as creating a partnership and sharing the risk. An example is the City of Anaheim’s Virtual Command Center, which connects the fire, police and other departments’ databases through a single web-based interface. This inter-departmental project maximizes taxpayer dollars and is the basis of an integrated emergency response capability.

2.e.3 Implementation is more important than theory.

Although a lab prototype or proof of principle can be valid risk mitigators on a path to an initial implementation, SOA implementation needs to do something that is operationally useful. eBay

and Amazon.com are both employing web services and SOA technology to leverage their software platforms.

eBay, an e-commerce leader on the Internet, conducts 40 percent of its business via their Web Services Platform.¹² They have built a SOA business model over a period of years centered on six core entities. Additionally, the company provides a developer's site with on-line documentation and sample code to enable third parties to tie applications via XML into the eBay platform. They also provide an architecture that supports all key web services development platforms. Finally, eBay provides a developmental sandbox for testing prior to taking new applications live.

Web services and SOA also drive the Amazon.com "Merchants" program. Amazon recognized that its loyal customers and unique technology were valuable to retailers—those who sought the resources and expertise to develop and sustain their own online retail presence, or those who saw Amazon as a major sales channel. Amazon began the Merchants program with the launch of the "Apparel & Accessories Store" in November 2002. The program, with a SOA based on standards-based web services, allows vast new product selections from third party, branded retailers to be placed on Amazon product pages.¹³

The U.S. Department of Justice (DOJ) has recently accepted the recommendation of its Global Infrastructure/Standards Working Group that the Department use SOA to achieve its vision that: "Any member of the justice community can access the information they need to do their job, at the time they need it, in a form that is useful, regardless of the location of the data." In particular, DOJ plans to securely scale information sharing across the more than the one million local, state and federal law enforcement, prosecution, courts and correction personnel.¹⁴

2.e.4 Pioneer! Do something!

Years ago, a leading systems integrator established a web service for weather data that provided real time, detailed weather data for displays, analysis, and forecasts. The systems integrator registered the service and made it available with no associated fee. Years later, when the service was taken off-line, the integrator was surprised to find that numerous organizations depended on that data. The exercise of developing and running the service served a specific internal company need, provided unanticipated lessons learned, and delivered value to external organizations. An enterprise should always be prepared to provide life-cycle support for pioneering services.

2.e.5 Ensure a robust publishing and discovery model to facilitate sharing and reuse.

The enterprise should establish the mechanisms (process, technology, governance, and incentives) that enable reuse across projects and streamline development of common services.

¹² Will Iverson, "Web Service in Action: Integrating with the eBay Marketplace," June 2004 <http://developer.ebay.com/devprogram/whitepaper_eBayWebServices2.pdf>.

¹³ "Systinet Powers Amazon Merchant Platform," <http://www.systinet.com/dl/cs_amazon.pdf>.

¹⁴ The Global Infrastructure Standards Working Group, "A Framework for Justice Information Sharing: Service-Oriented Architecture (SOA)" 28 Sept. 2004 <http://it.ojp.gov/documents/200409_Global_Infrastructure_Report.pdf>.

3. Conclusion and Next Steps

This review captures the key lessons learned and best practices from the Forum's initial industry survey. The tenets of effective SOA implementations are simplicity, open standards, modular design, and rapid, incremental implementation within a framework of organizational transformation. Clearly, a properly implemented SOA significantly improves operational effectiveness and delivers considerable return on investment. However, the principal lesson of this study is that SOA is simply a tool that must be implemented by engaged, attentive, and committed senior leaders who demand a culture of information sharing and improved organizational effectiveness.

Appendix 1 is a table of the SOA business practices presented in this document.

Appendix 2 is a Business Service Lifecycle Planner which provides a concise map for SOA implementation.

The Forum team is standing by to provide further refinement and detail as requested.

SIGNED: AFEI-sponsored Net-Centric Operations Industry Forum

Appendix 1: SOA Best Business Practices List

For quick reference purposes, the following is a high-level list of SOA and web services best practices presented in this document:

Vision and Leadership

- Evangelize the benefits of net-centricity, SOA, web services, and transformation.
- Think differently.
- Actively manage the cultural, strategic, and tactical issues of a major paradigm shift.
- Proactively address the cross domain and cross business area issues.
- Team with industry, across military services, and across executive agencies.
- Create and document a business case for SOA.

Policy and Security

- Establish technical standards.
- Establish portfolio management policies and policy/information standards and put them in a standards-based registry.
- Establish application interoperability policy.
- Consider how to benefit from both top-down *and* bottom-up leadership.
- Establish governance, security, reuse, compliance, risk management, and versioning policies.
- Employ multiple security approaches.
- Ensure security is “baked into the solution.”
- Address SOA-unique security considerations.
- Plan for disaster recovery, business continuance, and disaster management.

Strategy and Roadmap Development

- Develop, document and publish your SOA strategy.
- Plan for incremental transformation and deployment.
- Align programs/projects to share services.
- Maintain a vision of shared services but move toward it opportunistically and incrementally.
- Design for connections, change, and control.
- Create a common vocabulary.
- Recognize the importance of cross-enterprise architecture.
- Define and enforce application interoperability and business interoperability policies.
- Transform your IT development processes and policies.

Acquisition and Governance

- Incremental acquisition.
- Use experiments, pilots, and collaborative demos.
- Consider using enterprise modeling.
- Enforce policies.
- Loosely coupled services require detailed governance, management, and SLAs.
- Monitor, measure, and analyze the enterprise’s SOA service network.
- Promote Service Discovery and governance using a standards-based registry.
- Consider run-time discovery where appropriate and where it provides business value.

- Promote standards based process models, such as BPEL or Unified Modeling Language, for process model interoperability.

Implementation and Operations

- Implement incrementally, following the delivery of business value (benefits).
- Partnering and collaborative implementations work best.
- Implementation is more important than theory.
- Pioneer! Do something!
- Ensure a robust publishing and discovery model to facilitate sharing and reuse.

Appendix 2: Business Service Lifecycle Planner

This Business Service Lifecycle Planner provides a concise map for SOA implementation.¹⁵

Phase	Requirements / Activities	SOA Infrastructure	SOA Standards
Planning	<ul style="list-style-type: none"> • SOA governance & management • Policy enforcement • SOA metrics • Quality of service/reliability/latency policies • Security policies • Taxonomy design 	<ul style="list-style-type: none"> • Registry • Security • Management • Governance 	<ul style="list-style-type: none"> • Corporate standards & policies • WSDL • WSDM • WS-Policy
Enablement	<ul style="list-style-type: none"> • Business modeling • Corporate, business, IT governance • SOA infrastructure development • Web services development • Infrastructure development 	<ul style="list-style-type: none"> • SOAP — WS Server and runtimes • Application servers • Management and/or SOA network • Security proxies • WSDL, taxonomy, XML and other modeling tools • ESB, SOA networks, EAI/message brokers 	<ul style="list-style-type: none"> • BPEL4WS • WS-Addressing • WS-I Basic Profile • WS-Notification • WS-Eventing • WS-RM • WS-Security • XML, SOAP, WSDL
Publishing	<ul style="list-style-type: none"> • Business service approval • Certification process • Change management • Registration process & management • Categorize services and create taxonomies from service interface • Enrich service interfaces with policy-related metadata 	<ul style="list-style-type: none"> • Registry, taxonomy creation, content management • Policy design • Process design 	<ul style="list-style-type: none"> • UDDI

¹⁵ “A Practical Guide to SOA for IT Architects,” (Cambridge, MA: Systinet Corporation 2005) 10.

Industry Best Practices in Achieving Service Oriented Architecture

Phase	Requirements / Activities	SOA Infrastructure	SOA Standards
Discovery	<ul style="list-style-type: none"> Find and invoke business services Walk taxonomy trees Design time usage Runtime usage Configuration and change Operational management Introspect metadata 	<ul style="list-style-type: none"> Registry 	<ul style="list-style-type: none"> UDDI
Management And Security	<ul style="list-style-type: none"> Operate and manage business services Create, monitor and enforce SLAs and other policy Enforce security and identity Control service provider access Track and manage provider-consumer relationships Create parameters to monitor and provision monitoring tools Create and change taxonomies Create and change service providers 	<ul style="list-style-type: none"> Registry Identity server, hardware/ software firewalls Management proxies and instrumentation tools Alerting systems Discovery tools 	<ul style="list-style-type: none"> UDDI WS-DM WS-Security
Analysis	<ul style="list-style-type: none"> Analyze performance SOA metrics management SOA performance analysis 	<ul style="list-style-type: none"> Registry Management console Data mining/ analysis Visibility solutions 	<ul style="list-style-type: none"> UDDI WS-Policy assertion