SOA
Myth or Reality??

Jaqui Lynch
Mainline Information Systems
Email – jaqui.lynch@mainline.com

Session S04
Agenda

• Terminology and definitions
• What is SOA and why bother?
• What SOA is not
• Web Services and SOA
• SOA and ITIL
• Summary
• References

Acronym Soup

SOA JAVA
DCE .NET
RPC ESB
CORBA LDAP
MQ DCOM
SOAP ITIL
ITSM XML ...

And the winner for this week is

.......... SOA
Terminology and Definitions

- **SOA (from the OASIS definition of SOA)**
  - Service Oriented Architecture (SOA) is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains.

- **Other definitions**
  - A framework or architecture that allows IT assets to be organized into discrete components called services. Services can then be made available to applications to use.
  - Basically a collection of services that communicate with each other.
  - Not just for technology – it is often used as a business mapping methodology.
  - IBM SOMA = Service oriented modelling and architecture 2004

- **Service**
  - Any repeatable business task – e.g., check customer credit; open new account.
  - In IT this means a well defined self-contained independent function or entity.
  - In business terms it would be a service such as “open an account” or “buy music”

- **Loose coupling**
  - Try to reduce artificial dependencies.
  - Services maintain an awareness of each other but minimize dependencies.

- **Service Types**
  - Stateless
    - Completely standalone independent requests, very scalable.
  - Stateful
    - Needed to establish sessions between user and provider.
  - Idempotent Requests
    - Duplicate requests on failure – improves reliability.

- **Service Contract**
  - Agreement on how services will communicate.
  - Defined in service description documents.
Terminology and Definitions

• Service
  – Service Encapsulation
  – Service abstraction
    • Hide service logic from the outside world
  – Service reusability
    • Break logic into services that encourage reuse
  – Service composability
    • Composites of services
  – Service autonomy
    • Services control their own environment
  – Service optimization
    • High quality is better than low quality
  – Service discoverability
    • Should be setup so that discovery mechanisms can find them

Concepts of SOA

• Principles behind SOA
  – Reuse, granularity
  – Interoperability
  – Components, modularity
  – Industry and common standards
  – Services
    • Includes identification, provisioning, delivery, monitoring, etc

• Not all Web services fit into an SOA
  – e.g., SOAP-RPC does not as it is not interoperable by nature

• Not all SOAs use Web services technology
  – e.g., guaranteed messaging using message-oriented middleware (MOM) technology

• Services may be implemented using a variety of techniques
  – COM/.NET components, integration server adapters, ...
Enterprise Service Bus (ESB) – IBM version

Source: IBM

Enterprise Service Bus

- Standards based middleware infrastructure that allows the integration and linkage of services
- It does not implement SOA but provides the features that allow SOA to be implemented
- Abstraction layer on top of some kind of messaging system
- Requestor talks to a well known API
- Bus uses messaging technologies to route and deliver the messages and return the responses
Enterprise Service Bus – My Version

1. Buy music
2. Deliver music
3. Play new music

Why SOA?

- Simplification
- Reduce artificial dependencies
- Separates services from hardware, software and programming languages
- Links computational resources and promotes their reuse
- Business and technology agility
  - Cost effective and faster responses to changing market needs
  - Mergers and acquisitions are simplified
- Simplify connection to and usage of legacy assets
- Unrelenting user Requirements
  - On demand computing
  - Virtualization
SOA Reference Architecture

The enabling foundation for our product strategy

Built on a common framework for: Tooling, Runtime, Messaging, and Infrastructure

SOURCE: IBM

SOA Lifecycle

- Model
  - Gather requirements
  - Model and simulate design
- Assemble
  - Discover
  - Construct and test
  - Compose
- Deploy
  - Integrate people and processes
  - Manage and integrate information
- Manage
  - Manage applications and services
  - Manage security and compliance
  - Monitor business metrics
- And loop back to Model
- Key components
  - Interfaces
  - Messages
  - Extensibility
  - Service discovery
What SOA brings to the table

• Architecture that enables reuse and integration of services through applications based on standard protocols such as DCOM, CORBA, Web, etc
• Removal of silo based mentality
• MAJOR cultural change
• Language, software and hardware independent communications for services
• Interface definition hides the technical info so the application just calls the service
• AIM – IT and business responsiveness/agility

Why SOA now and why on P?

• We have the technology
  – Virtualization
  – Shared ethernet
  – Shared SCSI
  – Micropartitioning
  – Software such as MQ and Provisioning
  – Workload partitions
  – LPAR migration
  – Robust security
• And we have the performance
Shift To A Service-Oriented Architecture

- Function oriented
- Build to last
- Prolonged development cycles

- Coordination oriented
- Build to change
- Incrementally built and deployed

- Application silos
- Tightly coupled
- Object oriented
- Known implementation

- Enterprise solutions
- Loosely coupled
- Message oriented
- Abstraction

Source: Microsoft (Modified)

Web Services and SOA

- Interfaces must be based on protocols such as http, ftp, smtp, etc and should use XML messages where possible
- XML
  - Markup language for describing data in messages in a document format
- HTTP OR HTTPS
  - Request/response protocol used on the web
- SOAP
  - Protocol for exchanging XML based messages normally using HTTP
- XACML
  - Markup language for expressing access control rules and policies
- UDDI
  - Universal description discovery and integration
  - XML based registry to publish service descriptions and enable their discovery
- WSDL
  - Web services description language
  - XML based service description that includes interfaces, bindings, and message formats
- Web 2.0
  - Refers primarily to the new web sites that are setup for collaboration and sharing
  - Blogs, wikis, web syndication, etc
- Mashups
  - Web applications that combine content from multiple sources into one coherent view
What SOA is NOT

• A product
• A universal panacea
• Replacement for planning, funding and education
• A fix for poor management or bad attitudes
• Free or cheap
• New technologies do not normally fail due to the technology – typically it is due to lack of organizational commitment, unreasonable timeframes or expectations or lack of planning
• A cultural change is needed to be successful – for some reason people resist change

What is ITIL?

• ITIL (the IT Infrastructure Library) has been widely adopted worldwide as the framework or infrastructure for enabling IT Service management.
• The goal, as with SOA is business agility and quality of service
• Supports and is supported by the British Standards Institution's standard for IT service Management (BS15000).
• Consists of 7 books that provide a set of best practices, drawn from the public and private sectors internationally.
  1. Business perspective
  2. Service support
  3. Service delivery
  4. Security management
  5. Application management
  6. ICT Infrastructure Management
  7. Planning to implement IT service management
ITIL Books

IT vs SOA vs ITIL

<table>
<thead>
<tr>
<th>IT</th>
<th>SOA</th>
<th>ITIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology/function</td>
<td>Business/Coordination</td>
<td>Process</td>
</tr>
<tr>
<td>Reactive</td>
<td>Proactive</td>
<td>Proactive</td>
</tr>
<tr>
<td>Silos</td>
<td>Integrated, enterprise</td>
<td>Integrated, enterprise</td>
</tr>
<tr>
<td>Ad Hoc</td>
<td>Repeatable, reuse</td>
<td>Repeatable, reuse</td>
</tr>
<tr>
<td>IT focus</td>
<td>Service focus</td>
<td>Service focus</td>
</tr>
</tbody>
</table>
Why ITIL

- Organize large, heterogeneous IT Systems
- Instill best practices to avoid costly errors
- Cut costs
- Improve Service
- Better Asset Utilization
  - Virtualization is key here
- Helps you to focus on critical business processes

- Sound similar to SOA??????
- Some of it is

SOA/ITIL Common Ground

- Both focus on services
- Both are proactive and stress proper planning
- Both target an integrated enterprise
- Both focus on repeatable components of work and on reuse
- Both remove the silo or business unit focus
- Both are frameworks or architectures
- Neither is a product
- Both are about best practices

- SOA is targeted at service delivery and support areas so it could be viewed as a subset of ITIL service delivery and support
- The combination of SOA and ITIL moves organizations to be more service focused.
Summary

• SOA is here to stay for some time

• SOA allows you to integrate your services, business units and technologies by adopting a methodology that enables this

• IBM’s SOA has a major commitment to SOA as a strategic focus area

• System p virtualization and system design provide the necessary performance, availability, scalability, security and management so the benefits of SOA can be fully realized – this is a key to success

• SOA and ITIL do not compete – they actually complement each other

References

• Merging SOA and ITIL, Kunal Mittal

• CDBI Report, Business flexibility through SOA, David Sprott

• SOA Practitioners Guide

• http://www.service-architecture.com/web-services/articles/service-oriented_architecture_soa_definition.html


• Reap the benefits of SOA – Jaqui Lynch (IBM Systems Magazine)
Questions