SOA Software Architecture
- Adaptive and dynamic services
- More flexible, agile and reusable applications
- Services deployed to be shared across organizations
- Service software life cycle management approach?

SOA Software Lifecycle Model
- Sun – Conception, Inception, Elaboration, Construction, and Transition
- IBM (four iterative phases) – Model, Assemble, Deploy, and Manage
- webMethod – Plan, Design, Manage, Run, Use, and Change
- [1] - Design time, Deploy time, Run time, and Change time
- etc

Service Stakeholders and SOA Lifecycle Model [2]
- Service Provider (produce and publish services)
  - Design Time
    - Market needs/demands
    - Requirement engineering
    - Business modeling
    - Service design
    - Service development
    - Service testing
  - Run Time
    - Service provision: SLA: cost, availability, performance, etc
    - Service monitoring
  - Change Time
    - Service management
- Service Broker
  - Design Time
    - Registry selection
  - Run Time
    - Registry update
  - Change Time
    - Registry maintenance
- Application provider (Service consumer)
  - Design Time
    - Requirement engineering
    - Application design, implementation and model testing
  - Run Time
    - Service discovery
- Service orchestration/composition
- Service negotiation
- Service invocation
- Application testing
- Service monitoring
  - Change Time
    - Application maintenance

**IT Governance and SOA Governance**

- IT Governance
- A definition of SOA Governance [1]: “SOA governance is the creation, communication, enforcement, and adaptation of policies used to direct and control the creation and implementation of the life cycle of services.”
- Excerpts from [4]
  - SOA requires Governance from day one. Services are enterprise assets that will stay with a company for years to come. These assets need to be produced with enterprise quality and adhere to a well-defined set of standards and policies in order to ensure ongoing operations, reduced integration costs and complexities, and limit liabilities and security exposure.
  - Governance defines a new set of requirements. Implementing a Governance solution requires the definition of enterprise policies and establishment of strong auditing and conformance mechanisms to ensure that enterprise policies are being adhered to. Further, it requires analysis, tracking, and improvement of enterprise policies and architecture as the company’s SOA evolves.
  - Why SOA Governance?
    - The need for SOA Governance is business-oriented. In moving towards SOA, companies want to ensure continuity of business operations, manage security exposure, align technology implementation with business requirements, manage liabilities and dependencies, and reduce the cost of operations.
    - Several elements are required to achieve SOA Governance:
      - Enterprise SOA Policies…
      - Auditing and Conformance …
      - Management: Track, Review & Improve …
      - Integration …”

**Key Elements of SOA Governance**

- Policies: Management, Association, Enforcement, and Reporting
- Service Contract
- Service Lifecycle Management
- Metadata Management
  - Business info
  - Technical info
  - Governance info
Types of policies used in SOA [1]
- Messaging Security
- Access Control Policy
- Conformance to Enterprise Vocabulary and Schema
- Conformance to Technical Standards (WS-I, SWDL, WS-Security, WS-ReliableMessaging)
- Deployment process …
- Versioning Policies …
- Discovery Policy …
- Privacy Regulations …
- Quality of Service (QoS) …
- Reliability …
- Auditing and Reporting Requirements …
- Service Level Agreements (SLAs)

A Lifecycle Approach to SOA Governance [4]
SOA Governance Related Issues
- Compliance to standards or legislation, requiring audit trails of IT systems
- Budget (complicated, across multiple organization unit)
- Consequences of changing a service – hard to predict
- Ensuring quality of service is hard (design time, run time)
- Encapsulate business activities in services (attitude change from people)

Light Weight SOA Governance [5]
“Light weight SOA governance implies the following:
- Minimal documentation for service and service metadata.
- Light weigh procedures for service life cycle governance.
- Simple and lightweight tools and technical protocols to automate life cycle governance.
- Identification and addressing of only the key SOA governance concerns in a phased manner.
- Establishment of a Web 2.0 style service community evangelizing service usage and provisioning policies, scenarios and best practices. “

Excepts from [1]:
“SOA governance is the creation, communication, enforcement, and adaptation of policies used to direct and control the creation and implementation of the life cycle of services.

…
SOA management involves real-time control and collection of data about services.

…
SOA management process may directly affect governance.

…
The SOA governance life cycle is commonly referred to in four different phases – design time, deploy time, run time, and change time.
Design-Time Governance refers to the defining and controlling of enterprise services to be created in the enterprise, and the creation of policies used to direct and control the implementation of the enterprise service life cycle. In this phase, an SOA architect is responsible for defining and authoring policies for standard compliance, privacy requirements, access control, reliability, performance, messaging, and developing SLAs.

Deploy-Time Governance involves the process of testing and controlling compliance to enterprise policies in order for services to be deployed in an SOA. It involves deployment options and topologies, and adherence to policy should dictate whether or not a service can be deployed on a network.

Run-Time Governance refers to the process of enforcing the adherence to run-time service policies at run time. In addition to policy enforcement, this term is often used to include aspects of SOA management as it relates to these policies and to include real-time policy compliance monitoring, auditing, and measuring an collecting statistics.

Change-Time Governance involves managing services through the cycle of change. In a service life cycle, interfaces, service policies, and agreements may be modified many times. Change-time governance focuses on such issues as service versioning, depreciation, and run-time policy adoption. Governance tools can be used to achieve such strategies as adding service intermediaries to intercept messages and route them to the appropriate previous versions of services.

The following are common types of policies used in SOA governance:

- Messaging Security
- Access Control Policy
- Conformance to Enterprise Vocabulary and Schema
- Conformance to Technical Standards (WS-I, SWDL, WS-Security, WS-ReliableMessaging)
- Deployment process
- Versioning Policies
- Discovery Policy
- Privacy Regulations
- Quality of Service (QoS)
- Reliability
- Auditing and Reporting Requirements
- Service Level Agreements (SLAs)

References


