# 1.1.1 - Governance

## Contents of this Page

- **Summary**
  - Description of the profile
  - Capabilities
- **Requirements traceability**
- **complianceDiscovery**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **discovery**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **governanceService**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **identity**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **managementInformation**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **managementNotification**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **managementProcesses**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **metadata**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **metrics**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **metricsDiscovery**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **monitor**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **provenance**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **store**
  - Description
  - Requirements addressed
  - Overview of possible operations
- **workflowModel**
  - Description
  - Requirements addressed
  - Overview of possible operations

## Summary

*Description of the profile*
Service Oriented Architecture is an architectural paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. Consequently, it is important that organizations that plan to engage in service interactions adopt governance policies and procedures sufficient to ensure that there is standardization across both internal and external organizational boundaries to promote the effective creation and use of SOA-based services.

A workflow capability is deemed required to manage curation workflows and governance workflows.

**Governance** specializes capabilities architecturally implied by its associated concepts of Artifact, Governance, Management, Metrics, Policy. The implied architectural capabilities are described in the following paragraphs.

**Artifact** An artifact is a managed resource within the Semantic Infrastructure.

An artifact is associated with the following capabilities:

- descriptions to enable the artifact to be visible, where the description includes a unique identifier for the artifact and a sufficient, and preferably a machine processible, representation of the meaning of terms used to describe the artifact, its functions, and its effects;
- one or more discovery mechanisms that enable searching for artifacts that best meet the search criteria specified by the service participant; where the discovery mechanism will have access to the individual artifact descriptions, possibly through some repository mechanism;
- accessible storage of artifacts and artifact descriptions, so service participants can access, examine, and use the artifacts as defined.

**Governance** Service Oriented Architecture is an architectural paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. Consequently, it is important that organizations that plan to engage in service interactions adopt governance policies and procedures sufficient to ensure that there is standardization across both internal and external organizational boundaries to promote the effective creation and use of SOA-based services.

SOA governance requires numerous architectural capabilities on the Semantic Infrastructure:

* Governance is expressed through policies and assumes multiple use of focused policy modules that can be employed across many common circumstances This is elaborated in the inherited Policy profile.

* Governance requires that the participants understand the intent of governance, the structures created to define and implement governance, and the processes to be followed to make governance operational. This is provided by capabilities specialized from the inherited Management Profile.

* Governance policies are made operational through rules and regulations. This is provided by the following capabilities, most of which are specializations of the inherited Artifact Profile:

  - descriptions to enable the rules and regulations to be visible, where the description includes a unique identifier and a sufficient, and preferably a machine process-able, representation of the meaning of terms used to describe the rules and regulations;
  - one or more discovery mechanisms that enable searching for rules and regulations that may apply to situations corresponding to the search criteria specified by the service participant; where the discovery mechanism will have access to the individual descriptions of rules and regulations, possibly through some repository mechanism;
  - accessible storage of rules and regulations and their respective descriptions, so service participants can understand and prepare for compliance, as defined.

* SOA services to access automated implementations of the Governance Processes.

* Governance implies management to define and enforce rules and regulations. This is elaborated in the inherited Management profile.

* Governance relies on metrics to define and measure compliance. This is elaborated in the inherited Metric profile.

**Management** Governance implies management to define and enforce rules and regulations.

Management is provided by the following capabilities:

- an information collection site, such as a Web page or portal, where management information is stored and from which the information is always available for access;
- a mechanism to inform participants of significant management events, such as changes in rules or regulations;
- accessible storage of the specifics of processes followed by management.

**Metrics** Artifact Descriptions include references to metrics which describe the operational characteristics of the subjects being described

Architectural implications of metrics on the Semantic Infrastructure are reflected in the following capabilities:

- access to platform infrastructure monitoring and reporting capabilities
- access to metrics information generated or accessible by related services
- mechanisms to catalog and enable discovery of which metrics are available for a described artifact and information on how these metrics can be accessed;
- mechanisms to catalog and enable discovery of compliance records associated with policies, contracts, and constraints that are based on these metrics.

**Policy** Artifact Descriptions include references to policies defining conditions of use and optionally contracts representing agreement on policies and other conditions.

Architectural implications of policy on the Semantic Infrastructure are reflected in the following capabilities:
• descriptions to enable the policy modules to be visible, where the description includes a unique identifier for the policy and a sufficient, and preferably a machine processible, representation of the meaning of terms used to describe the policy, its functions, and its effects;
• one or more discovery mechanisms that enable searching for policies that best meet the search criteria specified by the service participant; where the discovery mechanism will have access to the individual policy descriptions, possibly through some repository mechanism;
• accessible storage of policies and policy descriptions, so service participants can access, examine, and use the policies as defined.

Policy capabilities are specialization of Artifact capabilities.

Capabilities

- complianceDiscovery
- discovery
- governanceService
- identity
- managementInformation
- managementNotification
- managementProcesses
- metadata
- metrics
- metricsDiscovery
- monitor
- provenance
- store
- workflowModel

Requirements traceability

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Source</th>
<th>Capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provide support for Curation Workflow Activities</td>
<td>Gap Analysis::Workflow:::000 - Provide support for Curation Workflow Activities</td>
<td>workflowModel</td>
</tr>
<tr>
<td>Provide a means to visualize the models that have been loaded into the KR. The UML Model Browser displays the granular details of a model in list format, but there isn’t away to visualize and navigate the whole model. The grid portal provides this feature from the metadata, there should be a way to do this when viewing models in the KR. This would enhance a potential new user’s ability to determine if they can use all or part of a model.</td>
<td>Gap Analysis::Workflow:::015 - Authoring tools should automate the construction of workflows</td>
<td>workflowModel</td>
</tr>
<tr>
<td>Provide a means to visualize the models that have been loaded into the KR. The UML Model Browser displays the granular details of a model in list format, but there isn’t away to visualize and navigate the whole model. The grid portal provides this feature from the metadata, there should be a way to do this when viewing models in the KR. This would enhance a potential new user’s ability to determine if they can use all or part of a model.</td>
<td>Gap Analysis::Workflow:::016 - Semantics describing workflows will be needed</td>
<td>workflowModel</td>
</tr>
<tr>
<td>Use business process workflow tool to automate the process of model submissions( and other related data) to the KR.</td>
<td>Gap Analysis::Workflow:::041 - Automate model submission process</td>
<td>workflowModel</td>
</tr>
<tr>
<td>Provide workflow component modularization and customization</td>
<td>Gap Analysis::Workflow:::135.3 - Modularization and Customization</td>
<td>workflowModel</td>
</tr>
<tr>
<td>Application developers need to be able to easily use the infrastructure to create workflows without having to figure out how to integrate the data or services, the infrastructure should step them through it and just ask for things that are not already available</td>
<td>Gap Analysis::Workflow:::155 - Create workflows</td>
<td>workflowModel</td>
</tr>
<tr>
<td>Behavioral dynamic models capture the behavior of services. Behavior of services provides an unambiguous definition of the service constraints, capabilities, dependencies and interactions. The metadata and grammar required to realize service behavior is called behavioral semantics. Behavioral semantics provide a mechanism for better service discovery and enforcing the constraints at design and runtime.</td>
<td>Semantic Infrastructure Requirements::Artifact Management::Behavioral Models</td>
<td>workflowModel</td>
</tr>
</tbody>
</table>
Service Oriented Architecture is an architectural paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. Consequently, it is important that organizations that plan to engage in service interactions adopt governance policies and procedures sufficient to ensure that there is standardization across both internal and external organizational boundaries to promote the effective creation and use of SOA-based services. SOA governance requires numerous architectural capabilities on the Semantic Infrastructure: Governance is expressed through policies and assumes multiple use of focused policy modules that can be employed across many common circumstances. This is elaborated in the inherited Policy profile. Governance requires that the participants understand the intent of governance, the structures created to define and implement governance, and the processes to be followed to make governance operational. This is provided by capabilities specialized from the inherited Management Profile. Governance policies are made operational through rules and regulations. This is provided by the following capabilities, most of which are specializations of the inherited Artifact Profile:** descriptions to enable the rules and regulations to be visible, where the description includes a unique identifier and a sufficient, and preferably a machine-processable, representation of the meaning of terms used to describe the rules and regulations; * one or more discovery mechanisms that enable searching for rules and regulations that may apply to situations corresponding to the search criteria specified by the service participant; where the discovery mechanism will have access to the individual descriptions of rules and regulations, possibly through some repository mechanism; * accessible storage of rules and regulations and their respective descriptions, so service participants can understand and prepare for compliance, as defined. * SOA services to access automated implementations of the Governance Processes. Governance relies on metrics to define and measure compliance. This is elaborated in the inherited Metric profile.

A service description is an artifact, usually document-based, that defines or references the information needed to use, deploy, manage and otherwise control a service. This includes not only the information and behavior models associated with a service to define the service interface but also includes information needed to decide whether the service is appropriate for the current needs of the service consumer. Thus, the service description will also include information such as service reachability, service functionality, and the policies and contracts associated with a service. A service description artifact may be a single document or it may be an interlinked set of documents. Architectural implications of service description on the Semantic Infrastructure are reflected in the following functional decomposition: * Description will change over time and its contents will reflect changing needs and context. This is elaborated in the inherited Change profile. * Description makes use of defined semantics, where the semantics may be used for categorization or providing other property and value information for description classes. This is elaborated in the inherited Semantic Model profile. * Descriptions include reference to policies defining conditions of use and optionally contracts representing agreement on policies and other conditions. This is elaborated in the inherited Policy profile. * Descriptions include references to metrics which describe the operational characteristics of the subjects being described. This is elaborated in the inherited Metrics profile. * Descriptions of the interactions are important for enabling auditability and repeatability, thereby establishing a context for results and support for understanding observed change in performance or results. This is elaborated in the inherited Interaction profile. * Descriptions may capture very focused information subsets or can be an aggregate of numerous component descriptions. Service description is an example of a likely aggregate for which manual maintenance of all aspects would not be feasible. This is elaborated in the inherited Composition profile. * Descriptions provide up-to-date information on what a resource is, the conditions for interacting with the resource, and the results of such interactions. As such, the description is the source of vital information in establishing willingness to interact with a resource, reachability to make interaction possible, and compliance with relevant conditions of use. This is elaborated in the inherited Interoperability profile.

This includes predefined templates, workflows, and governance policies for governing the service lifecycle as well as an approval and review process for service specifications and the ability to promote services through the stages of the service lifecycle. Service orchestration and choreography allows both application developers and non-developers to discover service "building blocks" that can be composed dynamically to provide business capabilities. Special cases include the orchestration of multiple services for a distributed query, or for a transactional workflow. Service orchestration and choreography will leverage static and behavioral semantics from the Semantic Infrastructure. The Semantic Infrastructure provides the behavioral semantics required for dynamic composition of services or generation of distributed queries. This includes runtime contract discovery and negotiation to determine composability of services based on service capabilities and constraints. Another use case is dynamic retrieval and enforcement of the policies that are in effect for a service interaction in the areas of logging, data transformation, or routing. This information can be used either during the design of the orchestration or during the execution of the defined flow. Link to use case satisfied from caGRID 2.0 Roadmap: Federated query over the TCGA data and other data sets is performed using a service orchestration.
One of the key requirements for participants interacting with each other in the context of a SOA is achieving visibility: before services can interoperate, the participants have to be visible to each other using whatever means are appropriate. The Reference Model analyzes visibility in terms of awareness, willingness, and reachability. Visibility in a SOA ecosystem has the following architectural implications on mechanisms providing support for awareness, willingness, and reachability. Mechanisms providing support for awareness will likely have the following minimum capabilities: * creation of Description, preferably conforming to a standard Description format and structure; * publishing of Description directly to a consumer or through a third party mediator; * discovery of Description, preferably conforming to a standard for Description discovery; * notification of Description updates or notification of the addition of new and relevant Descriptions; * classification of Description elements according to standardized classification schemes. In a SOA ecosystem with complex social structures, awareness may be provided for specific communities of interest. The architectural mechanisms for providing awareness to communities of interest will require support for: * policies that allow dynamic formation of communities of interest; * trust that awareness can be provided for and only for specific communities of interest, the bases of which is typically built on keying and encryption technology. The architectural mechanisms for determining willingness to interact will require support for: * verification of identity and credentials of the provider and/or consumer; * access to and understanding of description; * inspection of functionality and capabilities; * inspection of policies and/or contracts. The architectural mechanisms for establishing reachability will require support for: * the location or address of an endpoint; * verification and use of a service interface by means of a communication protocol; * determination of presence with an endpoint which may only be determined at the point interaction but may be further aided by the use of a presence protocol for which the endpoints actively participate.
identity

Description

Descriptions which include a unique identifier for the artifact.

Requirements addressed

- Governance Model
- Service Description Model

Overview of possible operations

managementInformation

Description

An information collection site, such as a Web page or portal, where management information is stored and from which the information is always available for access.

Requirements addressed

- Governance Model

Overview of possible operations

managementNotification

Description

A mechanism to inform participants of significant management events, such as changes in rules or regulations.

Requirements addressed

- Governance Model

Overview of possible operations

managementProcesses

Description

Accessible storage of the specifics of processes followed by management.

Requirements addressed

- Governance Model

Overview of possible operations

metadata

Description

A representation of the meaning of terms used to describe the artifact, its functions, and its effects.

Requirements addressed

- Governance Model
- Service Description Model

Overview of possible operations
metrics

Description
Access to metrics information generated or accessible by related services

Requirements addressed
- Service Description Model
- Governance Model

Overview of possible operations

metricsDiscovery

Description
Mechanisms to catalog and enable discovery of which metrics are available for a described artifact and information on how these metrics can be accessed.

Requirements addressed
- Service Description Model

Overview of possible operations

monitor

Description
Access to platform infrastructure monitoring and reporting capabilities.

Requirements addressed
- Service Description Model
- Governance Model

Overview of possible operations

provenance

Description
While the Resource identity provides the means to know which subject and subject description are being considered, Provenance as related to the Description class provides information that reflects on the quality or usability of the subject. Provenance specifically identifies the entity (human, defined role, organization, ...) that assumes responsibility for the resource being described and tracks historic information that establishes a context for understanding what the resource provides and how it has changed over time. Responsibilities may be directly assumed by the Stakeholder who owns a Resource or the Owner may designate Responsible Parties for the various aspects of maintaining the resource and provisioning it for use by others. There may be more than one entity identified under Responsible Parties; for example, one entity may be responsible for code maintenance while another is responsible for provisioning of the executable code. The historical aspects may also have multiple entries, such as when and how data was collected and when and how it was subsequently processed, and as with other elements of description, may provide links to other assets maintained by the Resource owner.

Requirements addressed

Overview of possible operations

store

Description
Accessible storage of artifacts and artifact descriptions, so service participants can access, examine, and use the artifacts as defined.

Requirements addressed
Service Description Model
Governance Model

Overview of possible operations

workflowModel

Description

Workflow Model maintenance

Application developers need to be able to easily use the infrastructure to create workflows without having to figure out how to integrate the data
or services, the infrastructure should step them through it and just ask for things that are not already available

Authoring tools should automate the construction of workflows.

Provide workflow component modularization and customization

Use business process workflow tool to automate the process of model submissions (and other related data) to the KR.

Provide support for Curation Workflow Activities

Semantics describing workflows will be needed.

This includes predefined templates, workflows, and governance policies for governing the service lifecycle as well as an approval and review
process for service specifications and the ability to promote services through the stages of the service lifecycle.

Service orchestration and choreography allows both application developers and non-developers to discover service "building blocks" that can be
composed dynamically to provide business capabilities. Special cases include the orchestration of multiple services for a distributed query, or for
a transactional workflow. Service orchestration and choreography will leverage static and behavioral semantics from the Semantic Infrastructure
2.0.

The Semantic Infrastructure provides the behavioral semantics required for dynamic composibility of services or generation of distributed queries.
This includes runtime contract discovery and negotiation to determine composibility of services based on service capabilities and constraints.

Another use case is dynamic retrieval and enforcement of the policies that are in effect for a service interaction in the areas of logging,
validation, data transformation, or routing. This information can be used either during the design of the orchestration or during the execution of
the defined flow.

Requirements addressed

- 016 - Semantics describing workflows will be needed
- 155 - Create workflows
- 015 - Authoring tools should automate the construction of workflows
- 041 - Automate model submission process
- 000 - Provide support for Curation Workflow Activities
- Service Governance and workflows
- Behavioral Models
- Service Orchestration and Choreography
- 135.3 - Modularization and Customization

Overview of possible operations

To be provided.