

**Run-time  
Service Oriented Architecture (SOA)**

**V 0.1**

July 2005

# Table of Contents

<b>1.0 INTRODUCTION .....</b>	<b>1</b>
<b>2.0 PRINCIPLES.....</b>	<b>1</b>
<b>3.0 FERA REFERENCE ARCHITECTURE .....</b>	<b>2</b>
<b>4.0 SOA RUN-TIME ARCHITECTURE.....</b>	<b>4</b>
4.1 FEDERATES .....	5
4.2 INTERFACES.....	5
4.2.1 <i>Portal</i> .....	5
4.2.2 <i>Gateway</i> .....	5
4.3 SOA FEDERATION .....	7
4.3.1 <i>Federation Server</i> .....	7
4.3.2 <i>Agent Framework</i> .....	9
4.3.3 <i>Collaborative Process (CP) Flow Controller</i> .....	9
4.3.4 <i>Event Manager</i> .....	10
4.3.5 <i>Activity Manager</i> .....	11
4.3.6 <i>Decision Manager</i> .....	11
4.3.7 <i>Built-in Services</i> .....	12
<b>5.0 SOA STANDARD CONVERGENCE .....</b>	<b>12</b>
<b>6.0 SOA DESIGN AND DEPLOYMENT .....</b>	<b>13</b>
<b>7.0 REFERENCES .....</b>	<b>15</b>

## 1.0 Introduction

Technology-driven collaborations among organizations or organizational units in a single organization face many challenges. The most critical challenges include

- Isolated proprietary systems
- Lack of commonly accepted open standard for business collaborations

To date, there has not been a single standard capable of resolving these and other related issues. However, within the last few years of very intensive developments of standard components mostly by ebXML and Web Services, we came to the point when we can say that a standard-based architectural solution that supports collaborative processes of any type is available now. That solution is presented in this document in the form of the loosely coupled Service Oriented Architecture (SOA) based on Federated Enterprise Reference Architecture (FERA) [FERA].

The FERA-based SOA Information Model (IM) [SOAIM] and SOA Collaboration Semantics (CS) [SOACS] developed by Semantion, are key SOA architectural components that integrate all other SOA components into the SOA run-time architecture.

SOA-based systems do not require traditional programming language coding. A complete SOA run-time specification in a XML form is generated from the collaborative process model by explicitly defining services, their inputs, their outputs and their static and dynamic choreography. This specification is captured in open standard XML-based deployment documents that support collaborative process execution on SOA.

## 2.0 Principles

Three main principles of the SOA architecture are:

- Completeness
- Open standard-based
- Standard convergence

SOA completeness is supported by SOA IM and SOA CS. SOA CS with SOA IM provides full semantics support for the overall integration of the SOA architecture.

SOA CS formally defines all necessary interfaces with methods and semantics required for the collaboration data (SOA IM) manipulations and interactions between the SOA architectural components.

SOA IM can be stored in a standard registry like OASIS ebXML Registry [ebRIM,ebRS] or OASIS UDDI [UDDI] and used to provide informational support for both context and content related to a business process. The SOA IM is presented in a form of an open standard-based XML document referred to as the Collaborative Process Information Document (CPID) [SOAIM] that can be either created manually or generated from a business process definition in a visual modeling tool.

The SOA components are based on open standards with the exception of the agent framework and business rules. There are no adequate agent framework or business rules standards available today that are conforming to all SOA requirements and one of the ebSOA TC goals will be to initiate and support the development of agent framework and business rules standards as well. All other SOA components are standard-based.

Standard convergence is a very important aspect of SOA. In this sense, convergence is the collection of standard-based components from different standard groups (ebXML, Web Services, etc.) with SOA IM and SOA CS providing their integration into SOA.

### **3.0 FERA Reference Architecture**

The main components of the FERA reference architecture (*Figure 1*) are: federates, interfaces and SOA Federation. Each participant involved in the collaboration is called a federate. Federates can be systems or people. People access the SOA Federation through the Portal interface using personal computers (desktop, laptop) or handheld wireless devices. Systems access the SOA Federation through the Gateway interface. The Gateway interface provides complete support for external (public) collaborations and secure communications between federates and the SOA Federation.

The SOA Federation is a central block of the SOA. It includes

- Federation Server
- Agent Framework
- Collaborative Process (CP) flow component and
- Built-in Services

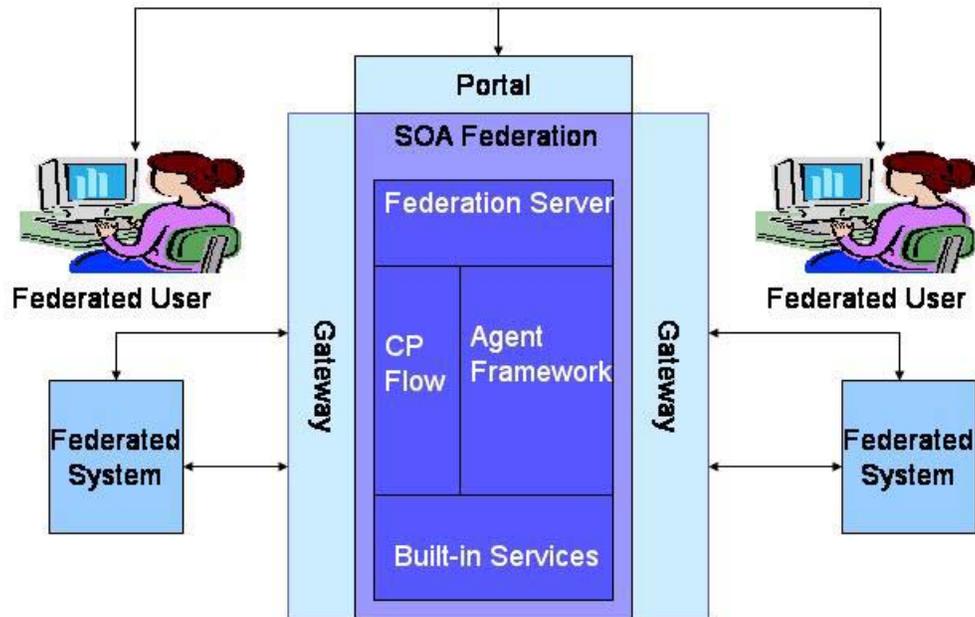


Figure 1: FERA Reference Architecture

The Federation Server is a bridge between the SOA external and internal worlds. The Agent Framework is a pool of agents that perform activities and make decisions during the execution of collaborative processes whenever they are assigned the role that can perform the activity or decision. The CP flow component manages collaborative process flows included in a collaborative process. Each collaborative process can have one or more collaborative process flows. A collaborative process flow includes activities, decisions, references to their inputs and outputs and events that are generated during the execution of the process either by the confirmation of an input/output receipt or by the validation of a rule evaluating specific metrics within the process. The SOA IM collaborative elements are explained in detail in the [SOAIM]. Built-in Services are third-party tools for the collaborative data analysis, reporting, etc., that can be used internally when they are invoked directly through the Federation Server interface or externally when they are invoked through either the Gateway or the Portal.

## 4.0 SOA Run-time Architecture

This section contains a detailed explanation of the SOA run-time architecture (*Figure 2*) based on the FERA reference architecture described in Section 3.

The key components of SOA are:

- Federates
- Interfaces
- SOA Federation

As explained in Section 3, federates are participants involved in the collaboration. Federates can be people and/or systems. Interfaces enable people and systems to access the SOA Federation and vice versa. The SOA Federation supports SOA integration and manages the context and the content of collaborative processes.

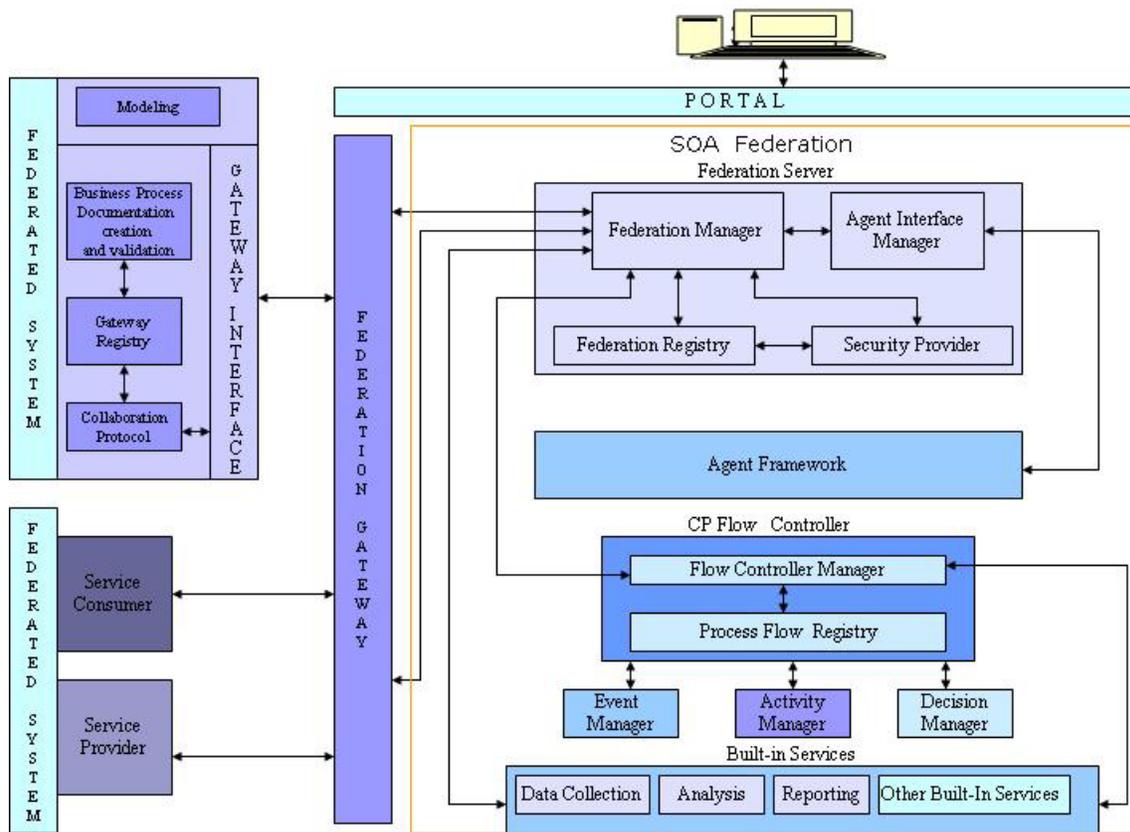


Figure 2: SOA Run-time Architecture

## 4.1 Federates

Federates are participants that perform activities and makes decisions included in the collaborative processes. They can be people and/or systems.

Systems can be simple as a basic web service or they can be complex SOA systems with many federates and a sophisticated SOA Federation. Both standard-based and proprietary systems are supported as long as they are able to communicate via SOA communication protocols through the Gateway interface. SOA supported communication protocols are explained in more detail in next section.

Standard-based systems are based on any open standard (i.e., Web Services, ebXML, etc.) or a combination of open standards.

Proprietary systems include any application solutions that are in existence today (i.e., Enterprise Resource Planning (ERP), Supply Chain Management (SCM), Customer Relationship Management (CRM), Product Life-Cycle Management (PLM), etc.)

## 4.2 Interfaces

Interfaces support communication with the SOA Federation. Two types of interfaces are defined:

- Portal
- Gateway

### 4.2.1 Portal

The Portal interface enables people to communicate with SOA Federation via either personal computers and/or handheld wireless devices.

The Portal interface specification is documented in [SOACS] document. This document specifies all portal forms needed for submissions and retrievals of information. This information includes meta-data and content for federate profiles, security policies, collaborative process specifications, business artifacts, etc. Additionally, the Portal interface supported protocols are listed in [SOACS].

### 4.2.2 Gateway

The Gateway interface provides support for external (public) collaborations and secure communications between federates and the SOA Federation.

The Gateway supports two XML-based communication protocols: ebXML Messaging [ebMS] and SOAP [SOAP] with or without WS-Security [WSS] and/or WS-Reliability [WSR].

Gateway collaboration support is provided by ebXML Business Process (BP) [ebBPSS], Web Services Business Process Execution Language (WSBPEL) [WSBPEL] and WS-Choreography [WSCHOR]. Two types of collaborative engines are supported:

- ebXML Collaborative Engine
- Web Services Collaborative Engine

ebXML Collaborative Engine is based on ebXML BP with ebXML Collaboration Protocol Profile and Agreement [ ebCPPA] standard specifications. Web Services Collaborative Engine is based on WSBPEL, WS-Choreography and Web Services Description Language (WSDL) [WSDL].

Beside references to the above standards supporting external collaborations and protocols, the SOA Gateway specification also includes definitions of Gateway message types needed to support communication between the Gateway Collaborative Engine and the SOA Federation. There are two generic message types defined:

- Administrative
- Collaborative

Administrative message types belong to the SOA Federation Server administrative tasks. These tasks include: submission and editing of SOA participant's profile information, definition and creation of all collaborative entity types, definition of collaborative documents, etc. Collaborative message types support SOA Federation Server collaborative operations (i.e., submissions of collaborative activities' outputs to the Federation Server, security and encryption checkups, Collaborative Engine request submission for an activity execution on a federate, etc.).

The SOA Gateway supports two types of web services: deterministic and non-deterministic. Deterministic web services are web services which participation in a collaborative process is known during the modeling time. Non-deterministic web services are web services which participation in a collaborative process is known only in the run-time during the collaborative process execution. Plug-in services is another name for non-deterministic web services. The logic for sourcing of plug-in services can be expressed through a combination of metrics and rules in the SOA IM and their actual sourcing in the execution can be determined by either an agent or by a human following the process choreography.

## 4.3 SOA Federation

The SOA Federation is a central block of the SOA architecture with the following components whose interfaces and methods are defined in the *SOA Collaboration Semantics* specification:

- Federation Server
- Agent Framework
- Collaborative Process (CP) Flow Controller
- Event Manager
- Activity Manager
- Decision Manager
- Built-in Services

The SOA Federation plays the role of the orchestration server. It takes control over the web services involved in the collaborative process and coordinates entire collaborative process execution.

### 4.3.1 Federation Server

Collaborative contents and contexts are defined and stored on the Federation Server. The Federation Server is a bridge between the external world and the SOA Federation. The Federation Server:

- Coordinates all collaborative activities during the collaborative process execution.
- Stores and version-controls collaborative participants (federates) profiles, the gateway profile, security profiles, business process specifications, collaborative documents, business artifacts and web services information.
- Stores and version controls meta-schemas defining collaborative process flow definitions.
- Communicates with the Gateway and the Portal.

The main components of the Federation Server are:

- The Federation Manager
- The Agent Interface Manager
- The Federation Registry
- The Security Provider

#### 4.3.1.1 Federation Manager

The Federation Manager is a central coordinator between federates and the SOA Federation. The Federation Manager

- Manages all requests and responses from and to the federates.
- Uses Security Provider services to perform all needed security authentications and authorizations.
- Manages and queries meta-data and content stored in the Federation Registry.
- Communicates with and provides necessary information for the Agent Interface Manager that manages the interface with the Agent Framework.

#### **4.3.1.2 Agent Interface Manager**

The Agent Interface Manager manages the interface with the Agent Framework.

The Agent Interface Manager services include:

- Agent delegation
- Agent monitoring and escalation
- Agent invocation (binding)

Agent delegation creates an agent interface (Agent entity in SOA IM) that contains details about agent and protocol required for communication with it, and associates agent with a specific activity or decision.

Agent monitoring and escalation service monitors local SOA Federation agents and performs escalations when agents are not available.

The agent invocation (binding) service uses agent interface to invoke an agent needed either to perform an activity or make a decision.

#### **4.3.1.3 Federation Registry**

The Federation Registry stores, version-controls, queries and maintains collaborative meta-data and content that include collaborative participants (federates) profiles, gateway profile, security profiles, business process specifications, collaborative documents, business artifacts and web services information.

Either ebXML Registry or UDDI can be used as a Federation Registry.

#### **4.3.1.4 Security Provider**

The Security Provider has three managers

- The CP Flow Security Manager
- The Federation Identity Manager

- The Federation Security Policy Manager

The CP Flow Security Manager manages and coordinates the Federation Identity Manager and the Federation Security Policy Manager. It directly communicates with the Federation Manager.

The Federation Identity Manager authenticates a user/system accessing the Federation Server. The Federation Security Policy Manager performs authorization checkups based on a security policy.

The core functionality of the Federation Identity Manager is based on Security Assertion Markup Language (SAML) [SAML] while the core functionality of the Federation Security Policy Manager is based on eXtensible Access Control Markup Language (XACML) [XACML].

The overall Security Provider semantics is based on SOA IM and SOA CS. The Security Provider is a perfect example of how SOA IM and SOA CS can be used not only for the collaborative process support in SOA but for the SOA architectural infrastructure as well.

### **4.3.2 Agent Framework**

The Agent Framework is a set of services that provide capabilities that are used to design, implement and administer agents. In the SOA Federation, Web Services are used to publish and deploy agent interfaces needed to send requests and receive responses from agents that perform activities and make decisions.

Agents can be running locally on the SOA Federation or remotely on SOA federates. Remote agents can be either fixed (located on the fixed location) or mobile. Local SOA Federation agents are supported by the native (local) Agent Framework (*Figure 2*). Federate (remote) agents are supported by federate Agent Frameworks running on SOA federates.

There is no an Agent Framework standard available today that has large acceptance. Because of this, the Agent Framework is in need of both an Agent Framework standard and an associated business rules standard since business rules are one of the key components of Agent Framework implementations. One of the SOA TC goals will be to initiate and support the development of the Agent Framework and business rules standards.

### **4.3.3 Collaborative Process (CP) Flow Controller**

The CP Flow Controller includes

- The Flow Controller Manager and
- The Process Flow Registry

Each collaborative process executed on a SOA system is a set of collaborative process flows and other collaborative entities that provide support for modeling collaborative processes. The Flow Controller Manager manages collaborative process flows. A collaborative process flow is a set of correlated events, activities and decisions that represent collaborations between roles. Detailed descriptions of all these entities and their attributes are in [SOAIM].

In FERA-based SOA, an activity or a decision can be performed by: an agent or a person or a person using a system with inputs received from the SOA Federation. Agents perform activities and decisions that can be fully automated and whose logic can be delegated to an agent or sourced by an agent in run-time based on rules and metrics.

The Flow Controller Manager manages collaborative process flows and availability of inputs, outputs, criteria and choices for collaborative processes' activities and decisions. For example, as soon as a reference to an input becomes available, the Flow Controller Manager will retrieve all activities, decisions and events which the input is related to. Based on that information, the Flow Controller Manager will send a message to the Activity Manager, or the Decisions Manager, or the Event Manager to start the activity or decision or fire the event's associated trigger.

At any time, a collaborative process flow is at a specific stage controlled by the Flow Controller Manager.

The Process Flow Registry manages all collaborative process flow informational entities: CollaborativeProcessFlow, Activity, Decision, Event, InputOutput, etc.. These entities, explained in [SOAIM], can also be stored and managed in the Federation Registry in which case the implementation of the Process Flow Registry is not needed.

#### **4.3.4 Event Manager**

The Event Manager manages events. An event is a collaborative element that represents that something happens during the CP flow. Each event has a trigger that creates the event and one or more actions that are the consequences of the event. Events can be organized into clusters or combined to form compound events. They progress through stages in the life cycle whereby each stage change has a meaning to the participants. Events can take place in the SOA Federation context or in the systems that are federated. Other collaborative elements (e.g., users, agents, systems) can subscribe to or publish events.

The Event Manager creates event instances when trigger conditions are met and controls the stages of events based on their trigger and action confirmation availability,

### **4.3.5 Activity Manager**

An activity is a task or an operation performed by either a SOA federate or a local SOA Federation agent.

The Activity Manager manages execution of activities communicating with the Flow Controller Manager and the Process Flow Registry.

When all required inputs for an activity become available, the Flow Controller Manager notifies the Activity Manager. The Activity Manager manages the execution of the activity based on the activity related information in the Process Flow Registry. The activity execution generates outputs that become inputs for another activity or criteria for a decision.

The Activity Manager also updates metrics if an activity generates metric information. A metric contains a quantifiable value that belongs to a specific argument that can be referenced in business rules that are used to dynamically control collaborative flow during the collaborative process execution. Metrics are also used to measure collaborative processes and generate their patterns.

### **4.3.6 Decision Manager**

Similar to an activity, a decision is performed (made) by either a SOA federate or a local SOA Federation agent.

The Decision Manager manages executions of decisions communicating with the Flow Controller Manager and the Process Flow Registry.

When all required criteria for a decision become available, the Flow Controller Manager notifies the Decision Manager. The Decision Manager manages the execution of the decision based on the decision related information in the Process Flow Registry. When a decision is made, a choice is generated and it becomes an input for another activity or a criteria for another decision.

The Decision Manager also updates metrics if a decision generates metric information.

### 4.3.7 Built-in Services

Built-in Services provide functionalities that enable the SOA Federation, SOA participants and SOA administrators to perform different types of tasks. Some of the typical SOA tasks include:

- Monitoring
- System and collaboration reporting
- Impact analysis
- Data analysis
- Data quality analysis
- Collaborative pattern analysis

These functionalities are not covered by any SOA specifications. They are just listed with a basic set of functions required to enable them.

## 5.0 SOA Standard Convergence

Most of the run-time SOA components are based on already available standards. SOA specification enables standard convergence by supporting already approved standard components which integration into the SOA architecture is provided by the FERA-based SOA Information Model [SOARIM] and the SOA Collaboration Semantics [SOACS].

*Figure 3* presents a list of standard components that support SOA.

The following is an initial list of standard specifications used in SOA:

- Data formatting (OASIS UBL, OAGIS, RosettaNet, etc.)
- Collaborative process information model and semantics (SOA IM, SOA CS)
- Messaging (W3C SOAP, OASIS ebXML)
- Life cycle management, query management and maintenance of meta-data, business process specifications, collaborative documents and business artifacts (OASIS ebXML Registry, OASIS UDDI)
- Collaborative processes (OASIS WSBPEL, W3C WS-Choreography, OASIS ebXML BP)
- Collaboration protocol profiles and agreements (W3C WSDL, OASIS ebXML CPPA)
- XML Documents content assembly and verification (OASIS CAM)
- Security (OASIS SAML, OASIS XACML, OASIS WS-Security)
- Portal interface (W3C HTML, JSP, W3C HTTP, W3C HTTPS, SSL, W3C XSL, W3C XSLT, etc.)

This list will be extended and modified over time to include all available standards supporting SOA.

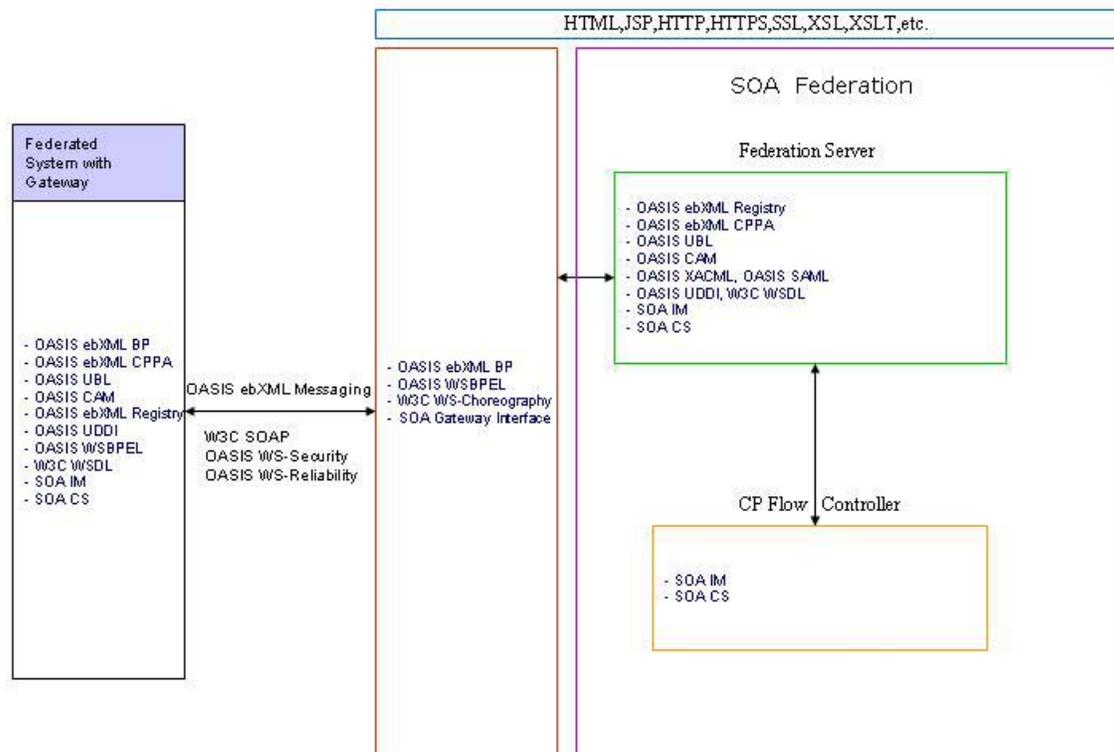


Figure 3: SOA Standard Convergence

The Web Services related standards should be used in accordance with the Web Services Interoperability (WS-I) [WSI] profiles developed by the Web Services Interoperability Organization.

## 6.0 SOA Design and Deployment

SOA design and deployment include the following steps:

- SOA business process modeling
- Detailed modeling of collaborative processes
- Deployment of SOA documents

These steps are presented in *Figure 4 - SOA Design* and *Figure 5 – SOA Deployment*.

According to business requirements, SOA business process modeling defines:

- Collaborative process and its collaborative process flows
- Collaborative process roles
- SOA participants (federates) with assigned roles
- Collaborative documents
- Collaborative scenarios

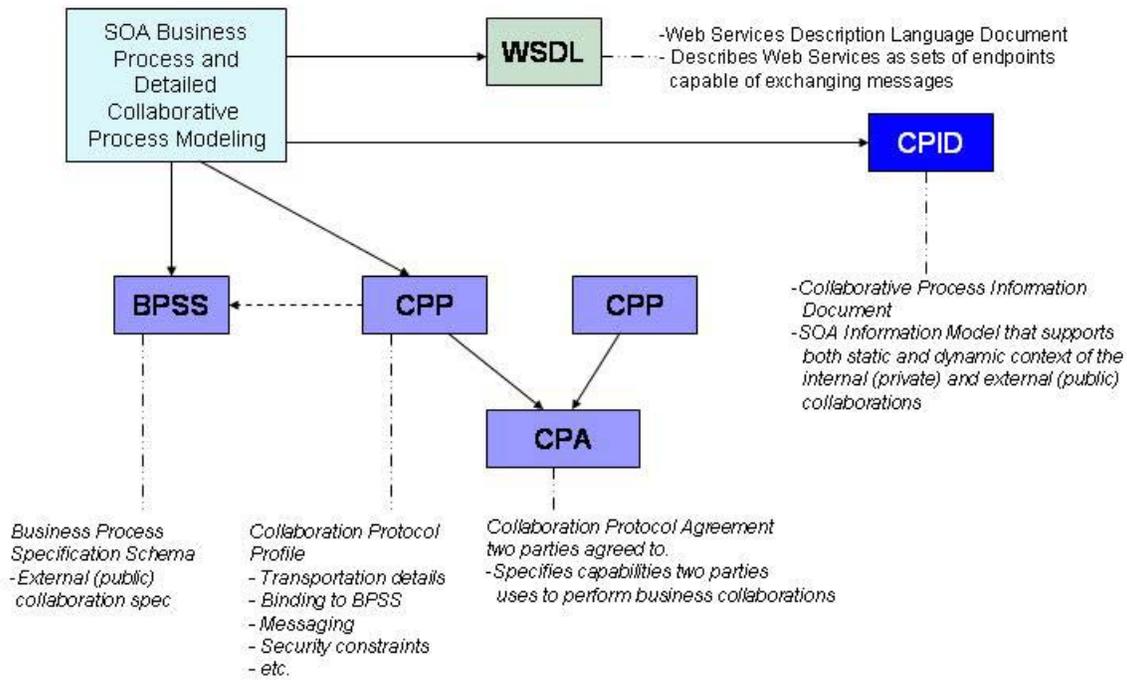


Figure 4: SOA Design

Each SOA supported business process, also known as a collaborative process, implements some kind of collaboration. A collaborative process can have one or more collaborative process flows that include activities, decisions and events taking place during the collaboration. Collaborative process roles perform activities and decisions. Collaborative process roles are assigned to SOA participants. A participant can be a user, a system or an agent.

There are two types of documents used in SOA: collaborative documents and SOA documents. Collaborative documents, for example invoices, purchase orders, store orders, etc., are documents that contain information received or generated during the execution of the collaborative processes. SOA documents are documents that contain information needed to model collaborative processes and support their execution in SOA. SOA

documents examples are SOA Collaborative Process Information Document (CPID), Web Services Description Language (WSDL), ebXML Collaboration Protocol Profile and Agreement (CPPA), etc.

When SOA business process modeling is completed, the next step is the detailed collaborative process modeling that models and defines all details related to activities, decisions and events in collaborative process flows. This step generates:

- The rest of collaborative document formats not defined in the SOA business process modeling step
- SOA documents (CPID, WSDL, CPPA, etc.)

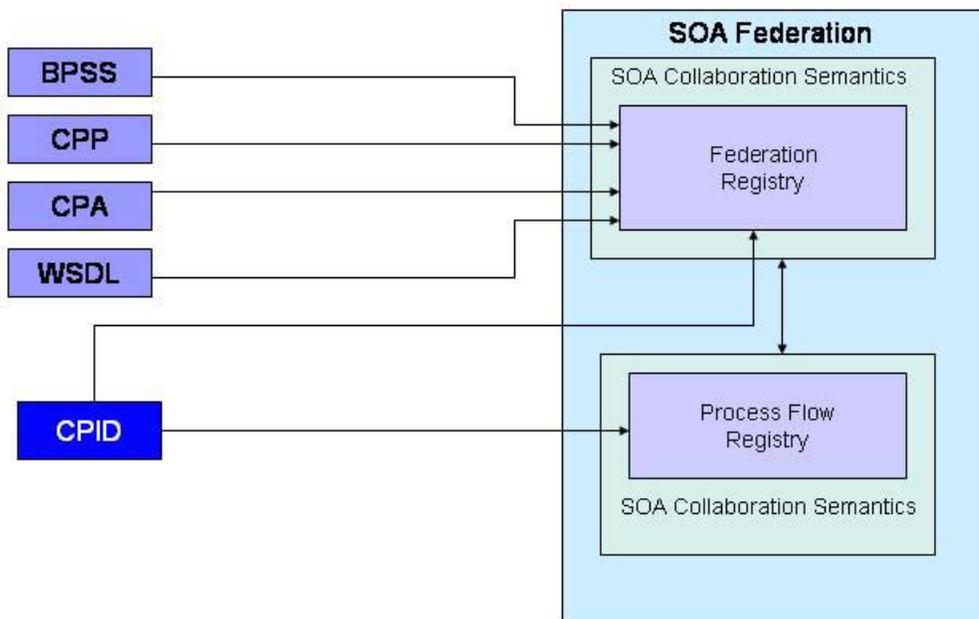


Figure 5: SOA Deployment

The final step is deployment of SOA documents in SOA Federation.

## 7.0 References

[ebBPSS] ebXML Business Process Specification Schema  
<http://www.ebxml.org/specs>

- [ebCPPA] ebXML Collaboration Protocol Profile and Agreement Specification (CPPA)  
<http://www.oasis-open.org/specs>
- [ebMS] ebXML Message Service Specification  
<http://www.oasis-open.org/specs>
- [ebRIM] ebXML Registry Information Model Specification  
<http://www.oasis-open.org/specs>
- [ebRS] ebXML Registry Services Specification  
<http://www.oasis-open.org/specs>
- [FERA] Closing the Process/Technology Gap, FERA: Federated Enterprise Reference Architecture: A Framework for Loosely Coupled Business Process Integration  
[http://www.cpd-associates.com/index.cfm?content=subpage&file=include\\_RPPage.cfm&ID=72404138&DOC=177813046](http://www.cpd-associates.com/index.cfm?content=subpage&file=include_RPPage.cfm&ID=72404138&DOC=177813046)
- [SAML] Security Assertion Markup Language (SAML)  
<http://www.oasis-open.org/specs>
- [SOACS] Service Oriented Architecture Collaboration Semantics (SOA CS) Specification  
[http://www.semantion.com/specs/soa/SOA\\_CS\\_V0.1.doc](http://www.semantion.com/specs/soa/SOA_CS_V0.1.doc)
- [SOAIM] Service Oriented Architecture Information Model (SOA IM) Specification  
[http://www.semantion.com/specs/soa/SOA\\_IM\\_V0.1.doc](http://www.semantion.com/specs/soa/SOA_IM_V0.1.doc)
- [SOAP] W3C Simple Object Access Protocol (SOAP)  
<http://www.w3.org/TR/SOAP>
- [UDDI] Universal Description, Discovery and Integration (UDDI)  
<http://www.oasis-open.org/specs>
- [WSBPEL] Web Services Business Execution Language (WSBPEL)  
[http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wsbpel](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel)
- [WSCHOR] WS-Choreography  
<http://www.w3.org/2002/ws/chor>
- [WSDL] Web Services Description Language (WSDL)  
<http://www.w3.org/TR/wsdl>
- [WSI] Web Services Interoperability (WS-I) Profiles  
<http://www.ws-i.org/deliverables/index.aspx>
- [WSR] Web Services Reliability (WS-R)

<http://www.oasis-open.org/specs>

[WSS] Web Services Security (WSS)

<http://www.oasis-open.org/specs>

[XACML] eXtensible Access Control Markup Language (XACML)

<http://www.oasis-open.org/specs>