OASIS Semantic Execution Environment TC:
Frequently Asked Questions Version 0.1

Note: This is an informal document and subject to change.

Q: What does the OASIS Semantic Execution Environment Committee do?

A: The OASIS SEE TC proposes that a standards-based execution environment for Semantic Web Services can provide a platform for service oriented architectures addressing the problems of data and behaviour interoperability between autonomous and heterogeneous business entities. The SEE TC will provide a reference architecture including component interfaces, semantic descriptions of the component interfaces and the components themselves, and the messaging mechanism to allow these components to communicate with each other. The OASIS SEE TC strongly recognizes the relationship of Semantic Web Services to Grid computing and aims to incorporate functionality required by Grid computing into the SEE architecture.

SEE infrastructure is based on the Service Oriented Architecture (SOA) paradigm and consists of a set of loosely coupled collaborating software components acting together in an open environment. We identify SOA to become the leading software paradigm for SWS infrastructure, however, SOA will not scale without signification mechanization of service discovery, service adaptation, negotiation, service composition, service invocation, and service monitoring; as well as data, protocol, and process mediation. SEE recognizes that SOA outside of tightly controlled environment (e.g., within the firewall) cannot succeed until/unless the semantics issue is addressed. In result of work of this committee, the set of common components coordinated through Common Service Layer will be identified and abstracted by their interfaces to facilitate their usage in the SEE infrastructure.

Q: What are benefits of open execution environment for Web services?

A: In order to support distributed heterogeneous applications built by different vendors, Web Services specifications have been developed. Existing Web Services cornerstone technologies provide the basic functionality for discovering (UDDI), describing interfaces (WSDL) and exchanging messages (SOAP) in heterogeneous, autonomous and distributed systems. But in practical terms, existing Web Services support operations which are limited to independent calls over a network, hard wired collaborations or predefined supply chains.

The approach to systems integration based on semantically enhanced Web Services is nowadays achievable through SWS. There are several efforts underway to develop a conceptual model, language and execution environments for SWS. The SEE TC aims to focus on providing an open execution environment for discovery, selection, mediation, invocation and interoperation of the SWS. The benefit of open execution environment for (Semantic) Web services is automation of tasks, which in current integration scenarios are typically carried by humans.

Q: Is a SEE part of a larger framework?

A: SEE is planned extendable (as a service oriented architecture) in order to allow further development and components additions. By its very design and purpose, it is meant to be part of a larger framework that aims to be a generic platform where SWS applications are supported for different business environments and processes. We are looking for a lightweight approach to coordinate various efforts in several European projects and some other initiatives that are all around the integration of semantic descriptions in service-oriented architectures. Since most of these initiatives use WSMO/WSML/WSMX we also decided to use it as a starting point for this TC.

Q: How work of SEE TC relates to Web Services Execution Environment (WSMX) effort?

A: The SEE TC work closely with WSMX group, having efforts coordinated as follows: the SEE TC takes care of the SEE architecture, APIs and execution semantics, while the WSMX working group focuses on SEE components development and implementation. In this way, conceptual and implementation levels are separated, allowing different implementation approaches for the same architecture, APIs and execution semantics.

Q: What is Web Services Modelling Ontology (WSMO) and how it relates to SEE?

A: Web Service Modeling Ontology (WSMO) is a framework for describing various aspects related to Semantic Web Services. It provides an ontology and language to capture four major aspects related to Semantic Web Services, namely:

Ontologies - provide the terminology used by other WSMO elements to describe the relevant aspects of the domains of discourse;

Web Services - are the computational entities providing access to services. They are described by their capabilities and their semantic means using terminology used in the ontologies;

Goals - represent user desires, for which appropriate Web services are discovered and executed. Goals are also refer to the terminology defined in the ontologies and specified using logical expressions;

Mediators - describe elements that overcome interoperability problems between different WSMO elements. These interoperability problems might stem from various mismatches, e.g. different terminology, communication protocols, etc.;

WSMO provides a formal foundation for the SEE. All aspects captured by WSMO are also reflected in components of the SEE system. WSMO aims to cover the complexity of

Semantic Web Services by their strong de-coupling, what gives a coherent view on the domain and facilities re-use of any of the aspects specified by WSMO.

Q: What is Web Services Modelling Language (WSML) and how it relates to SEE?

A: Web Service Modeling Language(WSML) provides a formal syntax and semantics for the WSMO. WSML is based on different logical formalisms, namely Description Logics, First-Order Logic and Logic Programming, which are useful for the modeling of Semantic Web services.

WSML consists of a number of variants based on these different logical formalisms, namely WSML-Core, WSML-DL, WSML-Rule and WSML-Full. These variants differ in their expressivity and syntax. Most complex capabilities are provided by WSML-Full which unifies the Description Logic and Logic Programming paradigms.

All WSML variants are described in terms of a normative human-readable syntax. Besides the human-readable syntax an XML and an RDF syntax is provided for exchange between machines. Furthermore, mapping between WSML ontologies and OWL are also provided.

In context of SEE, all aspects related to Semantic Web Services and their execution will be expressed in WSML. WSML allows specifying various properties using logical statements that can be dynamically processed by the computers. Thanks to the formal, logical representation computer reasoning can be applied in order to find out facts that are not explicitly stated. In principle, WSML provides a formal foundations for describing various aspects of Semantic Web Services what is further utilized and reflected in the execution environment.

Q: How does the SEE relate to Web services and other SOAs?

A: SEE is an execution environment for Semantic Web Services. Its scope is to enable a set of operations on Web Services from the most elementary ones as publishing, discovery and invocation to more advance ones as mediation, selection and orchestration. The semantics is added on top of the existing standards to better describe and disambiguate Web services as we know them today.

SEE is designed as a Service Oriented Architecture itself. The functionality mentioned above is offered by a set of modules and components seamlessly interconnected and managed in an architecture that follows the main principles of a SOA:

- Strong decoupling and strong mediation support
- Clear separation between Interface and Implementation
- Pear to Pear interactions (i.e. interactions between equal partners)

Q: How does SEE compare with related efforts elsewhere?

A: The OASIS SEE TC aims to continue the work of the WSMX working group, which took place in the context of several European Union projects such as DIP-Data, Information, and Process Integration with Semantic Web Services (http://dip.semanticweb.org), ASG - Adaptive Services Grid (http://asg-platform.org) or KW- Knowledge Web (http://knowledgeweb.semanticweb.org). Additionally, the TC members will be involved in future EU projects related with Semantic Web Services research area. This involvement has two purposes:

1. To ensure that the work developed by the SEE members is applicable in real use-case scenarios; this is done by interacting with different partners from both industrial and academic fields.
2. To be continuously informed of different approaches taken by different researchers, in order to combine efforts and avoid work duplication.

Q: Who will benefit from SEE?

A: There exist a good number of areas which have the potential to benefit from the solutions of the problems that SEE targets, because semantic interoperability is a widespread generic requirement in many of systems today. Among these, B2B is a prime application area.

Q: To what extend SEE will address issues of automatic service discovery, data and process mediation, negotiation, service composition, service invocation, service monitoring etc.?

A: SEE has no intention to address these issues directly. SEE TC focuses on outlining infrastructure for Semantic Web Services, interfaces of components and its execution semantics, while associated efforts (e.g. EU projects) can address above issues in details.

Q: Is a SEE implementable?

A: While SEE TC is not an implementation effort, we welcome groups working on architectures and especially on components for Semantic Web Services to align with SEE effort. WSMX working group, which was a direct initiator of SEE TC, will be transformed to implementation group aiming at providing reference implementation for SEE specifications.