# Triple Space Computing for Semantic Web Services

Omair Shafiq
Digital Enterprise Research Institute (DERI),
University of Innsbruck (UIBK)
6020 Innsbruck, Austria.
omair.shafiq@deri.org

**Abstract.** This thesis will address how to enable Triple Space Computing as a communication paradigm for Semantic Web Services. Currently, Semantic Web Services are following a message based communication paradigm. Triple Space Computing is envisioned as communication and coordination paradigm for Semantic Web Services which is an extension of tuple space computing to support RDF and then use it for communication based on the principle of persistent publication and read of data. Web Service Modeling Ontology (WSMO) is our conceptual model for Semantic Web Services. Web Service Execution Environment (WSMX) is one of the reference implementations of the WSMO conceptual model. The paper presents an overview of technical insights about integration of WSMX with Triple Space Computing and proposes that how WSMX can use Triple Space computing for its communication and coordination in terms of dynamic components management, external communication management, resource management and coordination of different interconnected WSMXs.

## 1 Introduction

Triple Space Computing (TSC) [1] has been proposed that defines the technologies and settings needed to develop a new paradigm for Web service communication that complies with the basic principles of the Web, i.e. stateless communication of resources, persistent publication of resources, unique identification of resources and non-destructive read access to resources [5]. The Triple Space Computing further adds compatibility with Web design principles, thus overcoming the deficiencies of message-based communication. This thesis addresses

In order to overcome drawbacks in existing communication paradigm of Semantic Web Services, integration of Triple Space Computing becomes a necessity. It will help Semantic Web Services to conform to the principles of Web by allowing communication based on persistent publication and read of semantic data in form of RDF triples over Triple Space. It will allow the reuse of information while communicating as information is published persistently. Asynchronous communication will allow SWS to work in distributed environments like Web. It will help in logging the results of time consuming processes so that it can be reused where required.

## 2 A roadmap for Triple Space Computing in Semantic Web Services

This thesis will integrate the Triple Space Computing with WSMX [2] by analyzing that how and where exactly the two technologies fit together. The integration has been proposed [4] as three major entry points which are (1) enabling components management in WSMX using Triple Space Computing, (2) External communication grounding in WSMX using Triple Space Computing and (3) Resource Management in WSMX using Triple Space Computing. This integration will be used further to enable the communication of different

inter-connected WSMX and then to build an application scenario to show the viability. Each of the integration entry points have been described in subsections below.

WSMX has a management component [3] that manages the over all execution by enabling coordination of different components based on some execution scenario [5] specified by user in Goal. In this way there is a clear separation between business and management logic in WSMX. The individual components have clearly defined interfaces and have component implementation well separated with communication issues. Each component in WSMX have wrapper to handle the communication issues. The WSMX manager and individual components wrappers are needed to be interfaced with Triple Space in order to enable the WSMX manager to manage the components over Triple Space. The communication between manager and wrappers of the components will be carried out by publishing and subscribing the data as a set of RDF triples over triple space. The wrappers of components that handle communication will be interfaced with Triple Space middleware.

WSMX acts as a semantic middleware between users and real world web services [3]. Currently, due to existence of message oriented communication paradigm, users communicate with WSMX and WSMX communicate with Web Services synchronously. The external communication manager of WSMX is needed to provide a support to communicate over Triple Space. The interfaces for sending and receiving external messages by WSMX are needed provide a grounding support to alternatively communicate over Triple Space. This needs to be resolved by addressing several issues, i.e. invoker component in WSMX is needed to support Web Services Description Language (WSDL) and Simple Object Access Protocol (SOAP) communication binding over Triple Space. The Entry point interfaces will be interfaced with Triple Space middleware in order to provide the glue between existing Web Services standards and Triple Space Computing.

WSMX contains different repositories to store ontologies, goals, mediators and web services descriptions as WSML based files [3]. The internal repositories of WSMX are needed to be made optional and enable to store the WSML based data as set of RDF named graphs in Triple Space Storage. This is mainly concerned with transforming the existing representation of data in form of WSML into RDF representation. The repository interfaces are needed to be interfaced with Triple Space middleware.

After enabling WSMX with Triple Space Computing, the next step will be to enable the communication and coordination of different WSMXs over Triple Space, i.e. forming a cluster of different interconnected WSMX nodes to support distributed service discovery, selection, composition, mediation, invocation etc. The management component in WSMX is will be enhanced to coordinate with WSMX managers in other WSMXs over Triple Space to form a cluster.

## References


1. D. Fensel, Triple-space computing: Semantic Web Services based on persistent publication of informatio: In Proceedings of the IFIP International Conference on Intelligence in Communication Systems, INTELLCOMM 2004, Bangkok, Thailand, November 23-26, 2004.
2. C. Bussler et al, Web Service Execution Environment (WSMX), W3C Member Submis-sion, June 2005. Available at http://www.w3.org/Submission/WSMX
3. M. Zaremba, M. Moran, T. Haselwanter, WSMX Architecture, D13.4v0.2 WSMX Working Draft.
4. O. Shafiq, R. Krummenacher, F. Martin-Recuerda, Y. Ding, D. Fensel, "Triple Space Computing middleware for Semantic Web Services", The MWS Workshop at 10th IEEE International Enterprise Computing Conference (EDOC 2006), 16-20 October 2006, Hong Kong.
5. T. Haselwanter, Maciej Zaremba and Michal Zaremba. Enabling Components Management and Dynamic Execution Semantic in WSMX. WSMO Implementation Workshop 2005 (WIW 2005), 6-7 June, Innsbruck, Austria.