Triple Space Computing: Adding Semantics to Space-Based Computing

Johannes Riemer[1], Francisco Martin-Recuerda[2], Ying Ding[3], Martin Murth[1], Brahmananda Sapkota[4], Reto Krummenacher[2], Omair Shafiq[2], Dieter Fensel[3], and Eva Kühn[1]

[1] Institute of Computer Languages, Space-Based Computing Group, Vienna University of Technology, Vienna, Austria
{mm, jr, eva}@complang.tuwien.ac.at
[2] Digital Enterprise Research Institute, University of Innsbruck, Innsbruck, Austria
{francisco.martin-recuerda, reto.krummenacher, omair.shafiq}@deri.org
[3] Electronic WebService GmbH, Innsbruck, Austria
[4] Digital Enterprise Research Institute, National University of Ireland, Galway, Galway, Ireland
{brahmananda.sapkota}@deri.org

Abstract. Triple Space Computing (TSC) is a very simple and powerful paradigm that inherits the communication model from Tuple Space Computing and projects it in the context of the Semantic Web. In this paper, we propose Triple Space Computing as a new communication and coordination framework for Semantic Web and Semantic Web Services. We identify the value added by TSC and propose the overall architecture of TSC and the interactions among different components.

1 Introduction

Triple Space Computing [1] is a powerful paradigm that inherits the communication model from Tuple Space Computing and projects it in the context of the Semantic Web. Instead of sending messages forward and backward among participants, like most of today's web service-based applications do, triple-based applications just use a simple communication based on reading and writing RDF triples [2] in shared persistent and semantically described information spaces. Triple Space Computing as a new paradigm for coordination and communication compliant with the design principles of the Web, thus provides a major building block for the Semantic Web and for interoperation of Semantic Web Services.

The current communication paradigm of Web Services is message-oriented. SOAP as a communication technology for XML implies messaging, WSDL defines messages that a Web Service exchanges with its user, and literally all ongoing research efforts around Semantic Web Services rely on these technologies. Triple Space Computing (TSC) [1] aims to promote a promising alternative to message-based communication technologies by adding semantics to Tuple Space computing [3]. TSC is based on the evolution and integration of several well-known technologies: Tuple Space Computing, Shared Object Space [4], Semantic Web and in particular RDF [2]. Tuple Space

R. Mizoguchi, Z. Shi, and F. Giunchiglia (Eds.): ASWC 2006, LNCS 4185, pp. 300 – 306, 2006.

computing was invented by David Gelernter in the mid-80s at Yale University. Initially presented as a partial language design, Linda was then recognized as a novel communication model on its own and is now referred to as a *coordination language* for parallel and distributed programming. Coordination provides the infrastructure for establishing communication and synchronization between activities and for spawning new activities. There are many instantiations or implementations of the Linda model, embedding Linda in a concrete host language. Examples include C-Linda, Fortran-Linda and Shared-Prolog. Linda allows defining executions of activities or processes orthogonal to the computation language, i.e. Linda does not care about, how processes do the computation, but only *how* these processes are created. The Linda model is a *memory* model. The Linda memory is called *Tuple Space* and consists of logical tuples. There are two kinds of tuples. Data tuples are passive and contain static data. Process tuples or "live tuples" are active and represent processes under execution. Processes exchange data by writing and reading data tuples to and from the Tuple Space.

However, [1] reports some shortcomings of the current Tuple Space models. They lack any means of name spaces, semantics, unique identifiers and structure in describing the information content of the tuples. TSC takes the communication model of Tuple Space Computing, wherein communication partners write the information to be interchanged into a common space and thus do not have to send messages between each other; TSC enhances this with the semantics required for Semantic Web enabled technologies.

The prototype development is based on Corso (Coordinated Shared Objects) system [4]. Corso is a platform for the coordination of distributed applications in heterogeneous IT environments that realizes a data space for shared objects. Corso offers maximum scalability and flexibility by allowing applications to communicate with one another via common distributed persistent „spaces of objects“. For testing and validating the TSC technology with special attention to the support for Semantic Web Services, we will integrate this system in the Semantic Web Service Environment WSMX[1], which is the reference implementation of the Web Service Modeling Ontology WSMO[2]. Thereby, the TSC technology will be aligned with emerging technologies for Semantic Web Services. By providing the basis for a new communication technology for the Semantic Web, TSC will provide a significant contribution to international research and development efforts around the Semantic Web and Semantic Web Services.

In this paper, we report some of the progresses. We present the overall architecture of TSC, which is mainly focusing on the TSC data and operation model and the introduction of different components involved in a Triple Space environment and the connections and interaction among these components. Finally we mention some potential future works.

2 TSC Architecture

Like the Web, TSC originally intended to build a Triple Space Computing infrastructure based on the abstract model called REST (Representational State Transfer) [6].

[1] www.wsmx.org
[2] www.wsmo.org