

ICAC 2005 Tutorials

TUTORIAL 1: SEMANTIC WEB SERVICES FOR AUTONOMIC COMPUTING: A CONCEPTUAL MODEL, LANGUAGE, AND EXECUTION ENVIRONMENT

PRESENTERS:

- Prof. Dr. Christoph Bussler, Digital Enterprise Research Institute
- Dipl-eng. Dumitru Roman, Digital Enterprise Research Institute
- Mr. Matthew Moran, Digital Enterprise Research Institute

ABSTRACT:

Despite being based on widely accepted standards, Web service technology has not realized its promise for internet-based integration between organizations. A significant stumbling block in this regard is the absence of a formal description of the meaning of the data exchanged by Web services. By implication, automatic discovery and automatic composition of Web services becomes unrealistic because of the potentially different interpretations of service descriptions and data exchanged at their interfaces. Semantic Web Services combines Semantic Web and Web service technologies to overcome these issues, and make services on the Web more “intelligent”. Based on semantic description frameworks, intelligent mechanisms are applied for automated discovery, composition, execution, and management of Web services. The tutorial explains how the application of semantics to Web services can overcome the deficiencies of the current Web Services technology stack of SOAP, WSDL, and UDDI as a platform for integration, and how semantics in Web services can make them more “intelligent”, thus allowing automatic tasks (e.g. discovery, selection, composition, mediation, execution, monitoring, etc.) to be performed with respect to Web services. The Web Service Modeling Ontology (<http://www.wmo.org>) provides the central theme for the tutorial. Moving from the conceptual model underlying WSMO, the tutorial goes on to explain the syntax and semantics of the languages used to express WSMO descriptions, before presenting and demonstrating a reference implementation. The tutorial finishes with a hands-on session where attendees get a chance to create their own Semantic Web Service based applications.

TUTORIAL OUTLINE:

- **Introduction to Semantic Web Services**
 - vision of Semantic Web Services
 - current Web Services technologies and their limitations
 - challenges for Semantic Web Services
- **WSMO Conceptual Model and Specification**
 - overview of WSMO: mission and working groups
 - WSMO building blocks: design and specification
 - Specific aspects (design and specification):
 - Web Service Modeling Language WSML
 - Choreography, Orchestration, and Mediation in WSMO
 - Comparison to related work

- Comparison to related work

- **WSMO-enabled Systems for Semantic Web Services**
 - Web Service Execution Environment: the WSMO reference implementation
 - IRS III: a bridge system between OWL-S and WSMO
 - Semantic Web Fred: automated goal resolution by electronic representatives
 - **WSMO tools and Hands-On Session**
 - introduction of the WSMO tools to be used
 - demonstration of application to be constructed
 - overview of provided Web Services
 - assisted hands-on session.
-

TUTORIAL 2: AC TOOLKIT TUTORIAL

PRESENTER:

- Balan Subramanian, IBM
- Jim Cybrinski, IBM

ABSTRACT:

The IBM Autonomic Computing Toolkit is a collection of core AC technology components that include components for data collection, problem determination, automated problem resolution, solution installation and an integrated solutions console. The toolkit is an offering from the Autonomic Computing group at IBM and includes a number of scenarios and tools to aid in the creation of autonomic systems that provide self-healing and self-configuring capabilities.

The objective of this tutorial is to educate attendees on the IBM Autonomic Computing Toolkit which is a key autonomic computing offering from IBM and encourage the adoption of the same. The toolkit is specifically built for developers who have an interest in developing autonomic systems.

TUTORIAL OUTLINE:

1. **Introduction to the IBM Autonomic Computing Toolkit**
2. **Problem determination and self-healing technologies**
 - a. Problem Determination Scenario
 - b. Generic Log Adapter Installation and runtime
 - c. Autonomic Management Engine overview and installation
 - d. Custom scenario development
 - Writing rules for the Generic Log Adapter
 - Writing resource models for AME
 - e. Log and Trace Analyzer
 - f. Correlation with the Log Trace Analyzer
3. **Self-configuring technology**
 - a. Solution install overview
 - b. Solution install scenario
 - c. Installing components with the Solution Install runtime

4. **Integrated Solutions Console**
 - a. Overview
 - b. ISC installation
 - c. Developing ISC components
 - d. Log Trace Analyzer in the ISC
5. **Other technologies**
 - a. IBM Touchpoint Simulator
 - b. Policy Management for Autonomic Computing

TUTORIAL 3: REINFORCEMENT LEARNING: A USERS' GUIDE

PRESENTER:

- Professor Bill Smart, Washington University in St. Louis

ABSTRACT:

Reinforcement Learning (RL) is a machine learning paradigm in which control strategies for agents are learned from direct experience of the world. It has been shown to be highly effective in areas as diverse as backgammon playing, elevator sequencing, and mobile robot control. Recently, some researchers have begun to incorporate RL techniques into Autonomic Computing (AC) applications, with good success.

This tutorial will introduce RL, with a heavy emphasis on how to use the algorithms and techniques in practical applications. The underlying theory of RL will be given, but the main goal of the tutorial is to build up an intuition of what RL is, what it is not, and how it can be successfully applied to AC problems. For those interested in a deeper coverage of the topics, references to the background material will be supplied in the tutorial notes.

TUTORIAL OUTLINE:

1. **Basic reinforcement learning.**

Covers the basic algorithms and techniques from RL, with examples relevant to the AC audience. Enough of the underlying mathematics will be given to allow understanding, but the main intent is to give a survey of the basic RL machinery, and an intuition about when it can be successfully applied to a problem.
 2. **Extensions to the basic algorithms.**

The basic RL techniques make several assumptions (discrete states, actions and time, and full observability of the states). This section deals with extensions that relax these conditions, so that the techniques are more relevant to real problems. Again, the intention is to provide a brief mathematical introduction and then build intuitions about why each of the extensions is useful, and what the associated problems are. The current state-of-the-art and open problems in each area will also be briefly discussed.
 3. **Current application of RL techniques to autonomic computing applications.**

A brief survey of AC applications from the current literature that use RL techniques. These techniques will be discussed, and linked back to the material covered in the previous two sections.
-