CALL FOR PAPERS

International Workshop on Component and Service Interoperability (WCSI-10)

One-Day Workshop
Málaga (Spain)
June 29, 2010

Held in conjunction with
Tools 2010 Federated Conferences
(June 28 – July 2, 2010)

Organizers

C. Canal, U. Málaga
canal@lcc.uma.es
www.lcc.uma.es/~canal

G. Salaün, Grenoble INP-INRIA-LIG
Gwen.Salaun@inria.fr
www.inrialpes.fr/vasy/people/Gwen.Salaun

J. Cámara, INRIA
Javier.Camara-Moreno@inrialpes.fr

Important Dates

Submission:	April 14, 2010
Notification:	May 14, 2010
Camera-ready:	May 31, 2010
Workshop:	June 29, 2010

Submission

PDF file; up to 12 pages in the EPTCS style, uploaded through the workshop's website

Web Site

http://wcsi10.lcc.uma.es

DESCRIPTION

The development of software systems requires mechanisms to structure them in order to tackle their complexity. This has led to the appearance of different kinds of abstractions to encapsulate system functionality, e.g., modules, objects, components, and more recently, services. Systems are then built as assemblies of these smaller and reusable entities, which are commonly developed by third parties, and that often present interoperability issues when assembled. Hence, interoperability is one of the key aspects related to the construction of large software systems, and can be defined as the ability of two or more entities to communicate in a proper way.

Several levels of interoperability, and accordingly of interface description languages (IDL), have been described. The **signature** level deals with the static aspects of component interoperability. At this level, IDLs (e.g.,CORBA-IDL, or WSDL descriptions in the case of Web Services) provide operation names, type of arguments and return values, as well as exception types. The **behaviour or protocol** level specifies the order in which the operations described in the signature interface should be invoked. Indeed, non-trivial software elements are stateful, and operation availability depends on their internal state. Some relevant examples of Behavioural IDLs (BIDLs) are Abstract BPEL, automata-based languages such as UML state diagrams, Petri Nets, high-level MSCs, etc. The **service** level includes the description of non-functional properties like temporal requirements, security, computational cost, etc. Quality of Service (QoS) descriptions and their related notations, such as the QoS Modeling Language (QLM), are examples of interface descriptions at this level. Finally, the **functional or conceptual** level concerns the functional or semantic specification of the component or service (i.e., what it actually does). In the field of Web Services, this level of description provides semantic information about services using ontology-based notations such as OWL-S or WSMO, which are particularly interesting for service discovery.

Interoperability problems may arise at any of the interface levels above. Detecting mismatch and providing the means to solve it is crucial for building systems out of components or services, enabling composition of entities developed by third parties. Some important objectives are the following: (i) proposing or extending existing interface models and description languages, (ii) detecting interoperability issues, (iii) controlling and ensuring correct interactions among the entities that form the system, and (iv) analyzing and verifying global system properties based on the features of its constituent parts.

TOPICS OF INTEREST

Topics of interest include, but are not limited to:

- Extended Interface Description Languages.
- Component mining and service discovery.
- Component/service types and contracts.
- Interface-based compatibility and substitutability.
- Software composition and adaptation.
- Self-adaptive systems; context-awareness.
- Controller Synthesis.
- Model-based approaches to component and service interoperability.
- Synthesis, negotiation and refinement of composition contracts.
- Formal models and approaches to interface mismatch detection, including static analysis, run-time verification, testing and model checking techniques.
- Runtime support for dynamic composition, adaptation and reconfiguration.
- Surveys, case studies and industrial or experience reports.

The goal of the workshop is to bring together researchers and practitioners interested in the aforementioned fields, to share and identify common problems, and to devise general solutions in the context of component and service interoperability. It is expected that formal paper presentations will be followed by lively discussions.

SUBMISSION & PUBLICATION

WCSI-10 calls for technical papers. Papers should be up to 12 pages, use the EPTCS style, and include the authors' names, affiliation and contact information. Submissions should be uploaded through the workshop's Easychair submission site (http://www.easychair.org/conferences/?conf=wcsi2010) **by April 14, 2010**. All submissions will be reviewed by the Program Committee. Accepted papers will be published in the proceedings of the workshop in the EPTCS series.

PROGRAM COMMITTEE

Luciano Baresi	Politecnico di Milano (Italy)	Gordon Pace	University of Malta (Malta)
Nelly Bencomo	Lancaster University (UK)	Pascal Poizat	University of Evry and LRI (France)
Antoine Beugnard	ENST Bretagne (France)	Razvan Popescu	Trinity College Dublin (Ireland)
Javier Cámara	INRIA (France)	Iman Poernomo	King's College London (UK)
Carlos Canal	University of Málaga (Spain)	Gwen Salaün	Grenoble INP-INRIA-LIG (France)
Rogerio de Lemos	University of Kent (UK)	Gerardo Schneider	IT University Göteborg (Sweden)
Marlon Dumas	University of Tartu (Estonia)	Massimo Tivoli	University of L'Aquila (Italy)
Markus Lumpe	Swinburne University (Australia)	Karsten Wolf	University of Rostock (Germany)