D14v0.1. Choreography in WSMO

WSMO Working Draft 04 October 2004

This version:
http://www.wsmo.org/2004/d14/v0.1/20041004/
Latest version:
http://www.wsmo.org/2004/d14/v0.1/
Previous version:
http://www.wsmo.org/2004/d14/v0.1/20040717/
Editor:
Dumitru Roman
Co-Authors:
Uwe Keller
Michael Stolberg
Dieter Fensel

Table of contents

---

1. Introduction

Choreography in WSMO is part of a service interface description; it describes the behaviour of the service by providing the necessary information to communicate with the service. The aim of this document is to provide a core conceptual model for describing the different types of choreographies in WSMO. The state-based mechanism for describing WSMO choreographies is based on the Abstract State Machines [Gurevich, 1995] methodology. The reason for choosing the ASMs as a basis for WSMO choreography is that ASMs provide a high flexibility in modelling

systems, being at the same time scientifically well founded. For a detailed explanation on ASMs we refer the reader to [Börger, 1998].

Taking the ASMs methodology as starting point, a WSMO choreography is defined as follows:

Listing 1. WSMO choreography definition

```
class wsmoChoreography
    states type state
    guardedTransitions typeSet guardedTransition
```

States
: A full instantaneous description of the behaviour of the service in its communication.

Guarded transitions
: Transition rules that transform the states.

The rest of the document is organized as follows: Section 2 of the document describes what is the signature of a state, Section 3 describes the guarded transitions, and Section 4 presents an example of how a choreography is modeled and Section 5 presents the conclusions and further directions.

2. States

A `state` is a specification in WSMO Ontology. The signature of the state is given by the elements of the WSMO Ontology:

- Non functional properties
- Imported Ontologies
- Used mediators
- Concepts
- Relations
- Functions
- Instances
- Axioms

Besides these declarations, a signature of a state may consists also of an infinite number of WSMO identifier declarations which will serve as means of changing the values of instances of concepts and relations in the state.

Out of the function declarations, two are of special importance and have a special meaning:

- `in` - a boolean, unary function. This function is an external function, meaning that its value is set by the environment and not by the service. Its argument may consists of a *concept*, *relation* or an *instance* of a concept or relation.

  Note: Receiving a message is a special type of `in` function, which returns true in case a message (the function's argument) is received. The message may consist of a *concept*, *relation* or an *instance* of a concept or relation.

- `out` - a boolean, unary function. Its argument may consists of a *concept*,

relation or an *instance* of a concept or relation.

Note: Sending a message is a special type of `out` function, which returns true in case the message it has as its parameter is sent. The message may consist of a *concept*, *relation* or an *instance* of a concept or relation.

3. Guarded transitions

`Guarded Transitions` are used to express changes of states and are expressible in the following form:

if *Cond* **then** *Updates*.

Cond is an arbitrary axiom without free variables, formulated in the given signature of the state using the logical language for defining formal statements defined in WSMO.

The *Updates* consist of arbitrary WSMO Ontology instance (see Section 3.7 of WSMO 1.0) statements.

4. Choreography description example

A service sells train tickets for a route. It receives a request for a complete route between 2 location; if there is a route, the service informs the user about the complete route between the two locations. The user requests a ticket for the route. The service requests the credit card of the user. The user informs the service about its credit card. The service checks the validity of the credit card and if the credit card is valid and there is an available seat, it sends the ticket (a string in this case) to the user.

[next draft]

5. Conclusions and further work

This document presented a core conceptual model for modeling WSMO Choreographies based on the ASMs methodology. Future versions of this document will give a precise translation of the states and the guarded transitions to ASMs in order to benefit from the using of ASMs interpreters, thus having an environment for running choreographies.

References

[Börger, 1998] Egon Börger: "High Level System Design and Analysis Using Abstract State Machines", Proceedings of the International Workshop on Current Trends in Applied Formal Method: Applied Formal Methods, p.1-43, October 07-09, 1998

[Gurevich, 1995] Yuri Gurevich: "Evolving Algebras 1993: Lipari Guide", Specification and Validation Methods, ed. E. Börger, Oxford University Press, 1995, 9--36.

Acknowledgement

The work is funded by the European Commission under the projects DIP, Knowledge Web, SEKT, SWWS, and Esperonto; by Science Foundation Ireland under the DERI-Lion project; and by the Vienna city government under the CoOperate program.

The editors would like to thank to all the members of the WSMO working group for their advice and input into this document.

webmaster

$Date: Monday 04 October 2004 - 00:15:05$
