DIP
Data, Information, and Process Integration with Semantic Web Services
FP6 - 507483

Deliverable

WP 4: Service Usage
D4a.10
Monitoring module specification

Marc Pellmann

Dec 19th, 2005

11/01/2006
EXECUTIVE SUMMARY

This deliverable focuses on monitoring of Semantic Web Services (SWS) and is a specification for the monitoring module in DIP. A prototype will be implemented in the next step.

This specification is based on the OASIS [1] Standards for Web Service Distributed Management (WSDM) [2], in particular on Management Using Web Services (MUWS) and Management Of Web Services (MOWS). The concepts in these specifications were enhanced to fulfill the special requirements of Semantic Web Services.

In addition to the execution monitoring defined in WSDM, the monitoring component in DIP will include some logging functionality and the ability to query logs and log files. The events that will be logged are the same as defined for the execution monitoring.

The monitoring will be included in at least one of the DIP use-cases. The prototype implementation for this specification (D4a.16) is used there.

This specification should be read by Web Services component developers, the use-case partners and all others that are interested in monitoring their services.
## Document Information

<table>
<thead>
<tr>
<th>IST Project Number</th>
<th>FP6 – 507483</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acronym</td>
<td>DIP</td>
</tr>
<tr>
<td>Full title</td>
<td>Data, Information, and Process Integration with Semantic Web Services</td>
</tr>
<tr>
<td>Project URL</td>
<td><a href="http://dip.semanticweb.org">http://dip.semanticweb.org</a></td>
</tr>
<tr>
<td>Document URL</td>
<td></td>
</tr>
<tr>
<td>EU Project officer</td>
<td>Kai Tullius</td>
</tr>
<tr>
<td>Deliverable Number</td>
<td>4a.10</td>
</tr>
<tr>
<td>Title</td>
<td>Monitoring Module Specification</td>
</tr>
<tr>
<td>Work package Number</td>
<td>4a</td>
</tr>
<tr>
<td>Title</td>
<td>Service Usage</td>
</tr>
<tr>
<td>Date of delivery</td>
<td>Contractual M 24 Actual 19-Dec-05</td>
</tr>
<tr>
<td>Status</td>
<td>Version 1.00 Final ✓</td>
</tr>
<tr>
<td>Nature</td>
<td>Prototype Report Dissemination Ontology ✓</td>
</tr>
<tr>
<td>Dissemination Level</td>
<td>Public Consortium ✓</td>
</tr>
<tr>
<td>Authors (Partner)</td>
<td>Marc Pellmann (inubit)</td>
</tr>
<tr>
<td>Responsible Author</td>
<td>Marc Pellmann Email <a href="mailto:marc.pellmann@inubit.com">marc.pellmann@inubit.com</a></td>
</tr>
<tr>
<td>Partner</td>
<td>inubit Phone +49-30 72 61 12-132</td>
</tr>
<tr>
<td>Abstract (for dissemination)</td>
<td></td>
</tr>
<tr>
<td>Keywords</td>
<td></td>
</tr>
<tr>
<td>Version Log</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue Date</th>
<th>Rev No.</th>
<th>Author</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>12-Dec-05</td>
<td>Draft 001</td>
<td>Marc Pellmann</td>
<td>First Draft</td>
</tr>
<tr>
<td>19-Dec-05</td>
<td>Final</td>
<td>Marc Pellmann</td>
<td>Changes from the reviews integrated</td>
</tr>
</tbody>
</table>
# Project Consortium Information

<table>
<thead>
<tr>
<th>Partner</th>
<th>Acronym</th>
<th>Contact</th>
</tr>
</thead>
</table>
| National University of Ireland Galway | NUIG | Prof. Dr. Christoph Bussler  
Digital Enterprise Research Institute (DERI)  
National University of Ireland, Galway  
Galway  
Ireland  
Email: chris.bussler@deri.org  
Tel: +353 91 512460 |
| Fundacion De La Innovacion.Bankinter | Bankinter | Monica Martinez Montes  
Fundación de la Innovación. Bankinter  
Paseo Castellana, 29  
28046 Madrid,  
Spain  
Email: mmminez@bankinter.es  
Tel: 916234238 |
| Berlecon Research GmbH | Berlecon | Dr. Thorsten Wichmann  
Berlecon Research GmbH  
Oranienburger Str. 32  
10117 Berlin,  
Germany  
Email: tw@berlecon.de  
Tel: +49 30 2852960 |
| British Telecommunications Plc. | BT | Dr John Davies  
BT Exact (Orion Floor 5 pp12)  
Adastral Park Martlesham  
Ipswich IP5 3RE,  
United Kingdom  
Email: john.nj.davies@bt.com  
Tel: +44 1473 609583 |
| Swiss Federal Institute of Technology, Lausanne | EPFL | Prof. Karl Aberer  
Distributed Information Systems Laboratory  
École Polytechnique Fédérale de Lausanne  
Bât. PSE-A  
1015 Lausanne, Switzerland  
Email: Karl.Aberer@epfl.ch  
Tel: +41 21 693 4679 |
| Essex County Council | Essex | Mary Rowlatt,  
Essex County Council  
PO Box 11, County Hall, Duke Street  
Chelmsford, Essex, CM1 1LX  
United Kingdom.  
Email: maryr@essexcc.gov.uk  
Tel: +44 (0)1245 436524 |
| Forschungszentrum Informatik | FZI | Andreas Abecker  
Forschungszentrum Informatik  
Haid-und-Neu Strasse 10-14  
76131 Karlsruhe  
Germany  
Email: abecker@fzi.de  
Tel: +49 721 9654 0 |
<table>
<thead>
<tr>
<th>Partner</th>
<th>Acronym</th>
<th>Contact</th>
</tr>
</thead>
</table>
| Institut für Informatik, Leopold-Franzens Universität Innsbruck | UIBK | Prof. Dieter Fensel  
Institute of computer science  
University of Innsbruck  
Technikerstr. 25  
A-6020 Innsbruck, Austria  
Email: dieter.fensel@deri.org  
Tel: +43 512 5076485 |
| ILOG SA | ILOG | Christian de Sainte Marie  
9 Rue de Verdun, 94253  
Gentilly, France  
Email: csma@ilog.fr  
Tel: +33 1 49082981 |
| inubit AG | Inubit | Torsten Schmae  
inubit AG  
Lützowstraße 105-106  
D-10785 Berlin  
Germany  
Email: ts@inubit.com  
Tel: +49 30726112 0 |
| Intelligent Software Components, S.A. | iSOCO | Dr. V. Richard Benjamins, Director R&D  
Intelligent Software Components, S.A.  
Pedro de Valdivia 10  
28006 Madrid, Spain  
Email: rbenjamins@isoco.com  
Tel. +34 913 349 797 |
| NIWA WEB Solutions | NIWA | Alexander Wahler  
NIWA WEB Solutions  
Niederacher & Wahler OEG  
Kirchengasse 13/1a  
A-1070 Wien  
Email: wahler@niwa.at  
Tel:+43(0)1 3195843-11 |
| The Open University | OU | Dr. John Domingue  
Knowledge Media Institute  
The Open University, Walton Hall  
Milton Keynes, MK7 6AA  
United Kingdom  
Email: j.b.domingue@open.ac.uk  
Tel.: +44 1908 655014 |
| SAP AG | SAP | Dr. Elmar Dorner  
SAP Research, CEC Karlsruhe  
SAP AG  
Vincenz-Priessnitz-Str. 1  
76131 Karlsruhe, Germany  
Email: elmar.dorner@sap.com  
Tel: +49 721 6902 31 |
<table>
<thead>
<tr>
<th>Sirma Al Ltd.</th>
<th>Sirma</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Atanas Kiryakov, Ontotext Lab, - Sirma AI EAD</td>
</tr>
<tr>
<td></td>
<td>Office Express IT Centre, 3rd Floor</td>
</tr>
<tr>
<td></td>
<td>135 Tzarigradsko Chausse</td>
</tr>
<tr>
<td></td>
<td>Sofia 1784, Bulgaria</td>
</tr>
<tr>
<td></td>
<td>Email: <a href="mailto:atanas.kiryakov@sirma.bg">atanas.kiryakov@sirma.bg</a></td>
</tr>
<tr>
<td></td>
<td>Tel.: +359 2 9768 303</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unicorn Solution Ltd.</th>
<th>Unicorn</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jeff Eisenberg</td>
</tr>
<tr>
<td></td>
<td>Unicorn Solutions Ltd, Malcha Technology Park 1</td>
</tr>
<tr>
<td></td>
<td>Jerusalem 96951</td>
</tr>
<tr>
<td></td>
<td>Israel</td>
</tr>
<tr>
<td></td>
<td>Email: <a href="mailto:Jeff.Eisenberg@unicorn.com">Jeff.Eisenberg@unicorn.com</a></td>
</tr>
<tr>
<td></td>
<td>Tel.: +972 2 6491111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vrije Universiteit Brussel</th>
<th>VUB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pieter De Leenheer</td>
</tr>
<tr>
<td></td>
<td>Starlab- VUB</td>
</tr>
<tr>
<td></td>
<td>Vrije Universiteit Brussel</td>
</tr>
<tr>
<td></td>
<td>Pleinlaan 2, G-10</td>
</tr>
<tr>
<td></td>
<td>1050 Brussel ,Belgium</td>
</tr>
<tr>
<td></td>
<td>Email: <a href="mailto:Pieter.De.Leenheer@vub.ac.be">Pieter.De.Leenheer@vub.ac.be</a></td>
</tr>
<tr>
<td></td>
<td>Tel.: +32 (0) 2 629 3749</td>
</tr>
</tbody>
</table>
LIST OF KEY WORDS/ABBREVIATIONS

MOWS Management Of Web Services
MUWS Management Using Web Services
OASIS Organization for the Advancement of Structured Information Standards
QoS Quality of Service
SOA Service Oriented Architecture
SWS Semantic Web Services
URL Unified Resource Loader
WSDL Web Service Description Language
WSDM Web Service Distributed Management
XML Extensible Markup Language
TABLE OF CONTENTS

EXECUTIVE SUMMARY .................................................................................................................. 1
LIST OF KEY WORDS/ABBREVIATIONS ...................................................................................... VI
TABLE OF CONTENTS .................................................................................................................. VII
1 INTRODUCTION .......................................................................................................................... 1
2 PRINCIPLES AND APPROACH .................................................................................................. 1
3 SWS AND WSDM ...................................................................................................................... 2
   3.1 General .................................................................................................................................. 3
   3.2 Events .................................................................................................................................. 3
   3.3 Capabilities .......................................................................................................................... 4
   3.4 Mandatory Capabilities ....................................................................................................... 4
   3.5 Logging and Querying ............................................................................................................ 5
      3.5.1 Example - NumberOfRequests .................................................................................. 6
   3.6 Monitoring Calls To Existing Web Services ....................................................................... 6
4 MONITORING PROXY ................................................................................................................. 6
   4.1 In-band/Out-of-band/Proxy ................................................................................................... 7
   4.2 How to use monitoring proxy component .......................................................................... 7
5 CONCLUSION ............................................................................................................................. 8
REFERENCES .................................................................................................................................. 8
APPENDIX I - WSDL ................................................................................................................... 8

LIST OF FIGURES
Figure 1: WSDM Concepts ........................................................................................................... 2
Figure 2: SWS in WSDM architecture ....................................................................................... 2
Figure 3: Metrics UML ................................................................................................................ 4
Figure 4: In-band/Out-of-band/Proxy ....................................................................................... 7
1 INTRODUCTION

In a distributed environment of services as in the service oriented architecture (SOA) a distributed mechanism for execution monitoring is needed. Semantic Web Services use service-oriented architecture as well.

The overall goal is to ensure the QoS objectives. Monitoring checks the health of a service in real-time and tries to reduce application downtimes by detecting signs of failure. It ensures that the service is available, accessible, and capable of meeting the throughput and latency requirements. It tracks availability, accessibility, and performance of the Web Service.

The management capabilities should include

- Monitoring the quality of a service
- Enforcing a service level agreement
- Controlling a task
- Managing a resource lifecycle

OASIS [1], which is responsible for a lot of standards in the Web Services world, defines a family of standards to address these requirements using standard Web Services.

MUWS (Management Using Web Services) [2] defines a flexible, expandable approach to monitor manageable resources. Such manageable resource can be any type of distributed information technology resource. It is part of WSDM (Web Services Distributed Management) [2], which also defines MOWS (Management Of Web Services) [2]. This is an application of MUWS to manage Web Service end points. This specification of monitoring semantic Web Services is also an application of MUWS and is based on MOWS because we have (semantic) Web Services.

This specification only refers to the monitoring aspects of WSML, which describes the whole management aspect. Monitoring in case of this specification means information about the service-execution at execution time including events and furthermore about queries on logged execution data of previous executions.

2 PRINCIPLES AND APPROACH

MUWS defines how the ability to manage, or, how the manageability of, an arbitrary resource can be made accessible via Web services.
Manageable resources can be accessed with a Web Service endpoint. The manageability consumer exchanges messages with the endpoint in order to subscribe to events, get events, and request information (see Figure 1).

Manageability is one possible aspect of a resource. A manageable resource may support some number of capabilities. Each capability has distinct semantics. For example this can be the number of requests of a Web Service or whether the resource is on-line or off-line. An implementation of a manageable resource provides a set of manageability capabilities via Web Service endpoints.

More detailed information is available in MUWS [2].

3 SWS AND WSDM
3.1 General
Composition of aspects of a Web Services implementation – for example, messaging, description, discovery, security, asynchronous notifications, etc. are provided by the Web Services platform and the respective standards specifications. Semantics are another aspect or a formal description of the service. It is addressed with WSDM as well.

MUWS uses XML Schema to describe properties. It allows a description of the semantics of the property and relevant metadata "in some form".

In order to use properties described in WSML, the XML Syntax of WSML [4] is used. Since this is an XML Schema description of the XML Syntax used by WSML, it can be used to describe the properties. Each property in MUWS must be described with a global element declaration. All elements of WSML XML Syntax are defined as such global element declarations and can be directly used as MUWS properties.

Thus either a full WSML description can be used as a property or only the wsmIIRI as a basic sort of an identifier in WSML to link to the description of the service that is grounded to this Web Service.

The operations of a service must be described in WSDL [3]. Thus all operational parts of a semantic Web Service that are managed/monitored need to be described in a WSDL.

3.2 Events
Events are used to deliver management information to interested parties. There could be any type of such an event. Special events for the semantic Web Services are described, if needed. The following shows the XML format of such an event:

```
<muws-pl-xs:ManagementEvent ...>
    <muws-pl-xs:ReportTime="xs:dateTime"/>
    <muws-pl-xs:EventId>xs:anyURI</muws-pl-xs:EventId>

    <muws-pl-xs:SourceComponent ...>
        <muws-pl-xs:ResourceId>xs:anyURI</muws-pl-xs:ResourceId> ?
        <muws-pl-xs:ComponentAddress>{any}</muws-pl-xs:ComponentAddress> *
            {any}*
    </muws-pl-xs:SourceComponent>

    <muws-pl-xs:ReporterComponent ...>
        <muws-pl-xs:ResourceId>xs:anyURI</muws-pl-xs:ResourceId> ?
        <muws-pl-xs:ComponentAddress>{any}</muws-pl-xs:ComponentAddress> *
            {any}*
    </muws-pl-xs:ReporterComponent>

</muws-pl-xs:ManagementEvent>
```
3.3 Capabilities

MOWS [2] defines a lot of capabilities that should all be used for semantic Web Services too, e.g. `GetManageabilityReferences()` to get the Web Service endpoints that provide the manageability implementation for this service. Another example is the metrics:

![Figure 3: Metrics UML](image)

In addition to these capabilities, special capabilities for semantic Web Services can be defined using the MUWS [2] specification. Currently it is not obvious, which additional capabilities are needed or if there are additional capabilities at all.

3.4 Mandatory Capabilities

The mandatory capabilities for monitoring Web Services are the metrics defined in MOWS. These are described in detail later. Web Service endpoints for a semantic Web Service that will implement the monitoring capabilities by itself need to implement these monitoring capabilities to monitor events for this metrics.

How to monitor these metrics for existing Web Service is described at the implementation prototype (see section 4).

The metrics are:

- **NumberOfRequests**
  A counter of the number of request messages that the Web Service endpoint has received. This counter is incremented by 1 whenever a request reaches the received state.

- **NumberOfFailedRequests**
  A counter of the number of request messages that the Web Service endpoint has received, and a (SOAP) fault that was sent in reply. This counter is incremented by 1 whenever a request reaches the state `Failed`.

- **NumberOfSuccessfulRequests**
  A counter of the number of request messages that the Web Service endpoint has received, and anything but a (SOAP) fault that was sent in reply. This counter is incremented by 1 whenever a request reaches the state `Completed`. 
• **ServiceTime**
  A counter of the total elapsed time (in seconds) that the Web Service endpoint has taken to process all requests (successfully or not).

• **MaxResponseTime**
  A gauge indicating the maximum time duration (in seconds) between all requests received and their completion or failure.

• **LastResponseTime**
  A gauge indicating the last recorded time duration (in seconds) between the last request received and its completion or failure.

If the prototype implementation for the monitoring of SWS shows, that the given events and definitions in the WSDM specification are more complex than needed in DIP, they can and should be reduced to the needed level of complexity. (see section 4)

### 3.5 Logging and Querying

In addition to the runtime information of the eventing of WSDM, the prototype provides abilities to log some information and query them later. This will be all of the above metrics in a given time period.

The operations will be named like the metric – e.g. NumberOfRequest. The given attributes are two *datetime* values as defined in XML Schema and are included in an own type and element named *TimePeriod* (see below). If some result values are not applicable to this time period, they are ignored. The type definition and all calls are described in Appendix I - WSDL.

The following element and complex type shows the structure of the time period information that need to be given with the call (in XML Schema syntax):

```xml
<xsd:complexType name="TimePeriod">
  <xsd:sequence>
    <xsd:element type="xsd:dateTime" name="Start"/>
    <xsd:element type="xsd:dateTime" name="End"/>
  </xsd:sequence>
</xsd:complexType>
```

The result value of each call has the same type as defined for this call in the MOWS schema definition, e.g. *mows-xs:NumberOfRequests* in the example below (the same as of the MOWS event – but now for the time period).
3.5.1 Example - NumberOfRequests

The following displays the XML structure of this example result in detail. The call gives back an element called NumberOfRequests with an integer value. And it has three attributes which are named Duration, LastUpdated and ResetAt – all of type dateTime:

```
<numberOfRequests>
  <Duration>1.1</Duration>
  <LastUpdated>2001-01-21T14:30:00.000Z</LastUpdated>
  <ResetAt>2001-01-21T14:30:10.000Z</ResetAt>
</numberOfRequests>
```

This is defined in XML Schema and shown below. The attribute-group is defined in the original MUWS schema definition file and defines the attributes Duration, LastUpdated, and ResetAt.

```
<xs:schema xmlns:xs='http://www.w3.org/2001/XMLSchema'>
  <xs:element name='NumberOfRequests'>
    <xs:complexType>
      <xs:extension base='xs:IntegerCounter'>
        <xs:attributeGroup ref='mows:MetricAttributes'/>
        <xs:attribute name='Duration' type='xs:dateTime'/>
        <xs:attribute name='LastUpdated' type='xs:dateTime'/>
        <xs:attribute name='ResetAt' type='xs:dateTime'/>
      </xs:complexType>
    </xs:element>
  </xs:schema>
```

3.6 Monitoring Calls To Existing Web Services

For monitoring the metrics of existing Web Services a proxy approach implementation will be provided. Because the user cannot change the actual implementation of the callable service, the proxy will provide the needed information for the monitoring.

This proxy will also be the reference implementation of monitoring and is further explained below.

4 Monitoring Proxy

A monitoring proxy prototype will be developed. It will provide (at least) the metrics as defined in MOWS. It broadcast events as described in MOWS and in addition with its own events with WSML properties (described in the XML syntax – see section 3.1). All of these events will be logged and can be queried as described in 3.5.
4.1 In-band/Out-of-band/Proxy

Each service itself, which is a special developed service for DIP, can use an In-band approach to give the manageability Web Service end-point or an own Out-of-band manageability Web Service can be developed in addition to the Web Service itself.

If there is an external Web Service the proxy approach can be used.

![Diagram showing In-band/Out-of-band/Proxy](image)

Figure 4: In-band/Out-of-band/Proxy

Figure 4 shows three different approaches to offer manageability Web Service endpoints for a manageable Web Service endpoint. In an in-band approach it is implemented directly in the manageable Web Service endpoint. In the out-of-band approach it is implemented in a second Web Service endpoint, which gets information directly from the endpoint. It is implemented in a way to generate these. The third approach, which is used in the prototype-implementation, is explained in the next section.

4.2 How to use monitoring proxy component

The monitoring proxy component provides a Web Service end-point to register a service observation. You can call this Web Service and pass the Web Service URL of the service you wish to monitor. You will get back the Web Service endpoint that you need to call for using your given Web Service with monitoring.
This Web Service end-point will have the same WSDL as the given original Web Service, but in addition it includes the monitoring subscription operations and the query operations as described above.

Thus the WSDL of the original Web Service's end-point will be enriched with the needed information to get the manageability information.

If the Web Service is called it will forward the call to the original Web Service and broadcast the monitoring information to the subscribed end-points. In addition it logs all metrics and can answer queries about the logged metrics later.

You will find the WSDL for these queries in Appendix I - WSDL. The other monitoring services are described in WSDM [2]. The WSDL and XML Schema can be found there.

5 CONCLUSION
In this deliverable we have shown how the monitoring of semantic Web Services will work and which execution monitoring information and logging queries are provided by monitoring a component prototype.

REFERENCES
[3] Web Service Description Language; http://www.w3.org/TR/wsd1

APPENDIX I - WSDL
WSDL for the query calls to the logging data.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmns:wsc="http://wsmx.org/2005/12/monitoring"
xmlns:mows-xsd="http://docs.oasis-open.org/wsdm/2004/12/mows/wsdm-mows.xsd"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmns:ws:mmx.org/2005/12/monitoring"
xmlns:soap=">http://docs.oasis-open.org/wsdm/2004/12/mows/wsdm-mows.xsd"
xmlns:targetNamespace=">http://wsmx.org/2005/12/monitoring"
xmlns:types>
<wsdl:complexType name="TimePeriod">
<wsdl:sequence>
<wsdl:element name="Start" type="xsd:dateTime"/>
<wsdl:element name="End" type="xsd:dateTime"/>
</wsdl:sequence>
</wsdl:complexType>
</wsdl:types>
</wsdl:definitions>
```
9

</xsd:complexType>
10       <xsd:element name="TimePeriod" type="wsc:TimePeriod"/>
11     </xsd:schema>
12     <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
13         targetNamespace="http://docs.oasis-open.org/wsdm/2004/12/mows/wsdm-mows.wsdl">
14         <xs:import namespace="http://docs.oasis-open.org/wsdm/2004/12/mows/wsdm-mows.xsd"
15           schemaLocation="http://docs.oasis-open.org/wsdm/2004/12/mows/wsdm-mows.xsd"/>
16     </xs:schema>
17     <wsdl:types>
18       <wsdl:message name="LastResponseTime">
19         <wsdl:part name="Period" element="wsc:TimePeriod"/>
20       </wsdl:message>
21       <wsdl:message name="ServiceTime">
22         <wsdl:part name="TimePeriod" element="wsc:TimePeriod"/>
23       </wsdl:message>
24       <wsdl:message name="NumberOfRequestsResponse">
25         <wsdl:part name="return" element="mows-xs:NumberOfRequests"/>
26       </wsdl:message>
27       <wsdl:message name="NumberOfSuccessfulRequests">
28         <wsdl:part name="Period" element="wsc:TimePeriod"/>
29       </wsdl:message>
30       <wsdl:message name="NumberOfSuccessfulRequestsResponse">
31         <wsdl:part name="return" element="wsc:TimePeriod"/>
32       </wsdl:message>
33       <wsdl:message name="NumberOfRequests">
34         <wsdl:part name="Period" element="wsc:TimePeriod"/>
35       </wsdl:message>
36       <wsdl:message name="LastResponseTimeResponse">
37         <wsdl:part name="return" element="mows-xs:LastResponseTime"/>
38       </wsdl:message>
39       <wsdl:message name="MaxResponseTime">
40         <wsdl:part name="Period" element="wsc:TimePeriod"/>
41       </wsdl:message>
42       <wsdl:message name="NumberOfFailedRequestsResponse">
43         <wsdl:part name="return" element="mows-xs:NumberOfFailedRequests"/>
44       </wsdl:message>
45       <wsdl:message name="NumberOfFailedRequests">
46         <wsdl:part name="Period" element="wsc:TimePeriod"/>
47       </wsdl:message>
48       <wsdl:message name="ServiceTimeResponse">
49         <wsdl:part name="return" element="mows-xs:ServiceTime"/>
50       </wsdl:message>
51     </wsdl:types>
52     <wsdl:portType name="MonitoringPort">
53       <wsdl:operation name="NumberOfRequests">
54         <wsdl:input name="NumberOfRequests" message="wsc:NumberOfRequests"/>
55         <wsdl:output name="NumberOfRequestsResponse" message="wsc:NumberOfRequestsResponse"/>
56       </wsdl:operation>
57       <wsdl:operation name="NumberOfFailedRequests">
58         <wsdl:input name="NumberOfFailedRequests" message="wsc:NumberOfFailedRequests"/>
59         <wsdl:output name="NumberOfFailedRequestsResponse" message="wsc:NumberOfFailedRequestsResponse"/>
60       </wsdl:operation>
61       <wsdl:operation name="NumberOfSuccessfulRequests">
62         <wsdl:input name="NumberOfSuccessfulRequests" message="wsc:NumberOfSuccessfulRequests"/>
63         <wsdl:output name="NumberOfSuccessfulRequestsResponse" message="wsc:NumberOfSuccessfulRequestsResponse"/>
64       </wsdl:operation>
65     </wsdl:portType>
10

73     <wsdl:output name="NumberOfSuccessfulRequestsResponse"
74         message="wsc:NumberOfSuccessfulRequestsResponse"/>
75     </wsdl:operation>
76     <wsdl:operation name="ServiceTime">
77     <wsdl:input name="ServiceTime" message="wsc:ServiceTime"/>
78     <wsdl:output name="ServiceTimeResponse" message="wsc:ServiceTimeResponse"/>
79     </wsdl:operation>
80     <wsdl:operation name="MaxResponseTime">
81         <wsdl:input name="MaxResponseTime" message="wsc:MaxResponseTime"/>
82         <wsdl:output name="MaxResponseTimeResponse" message="wsc:MaxResponseTimeResponse"/>
83     </wsdl:operation>
84     <wsdl:operation name="LastResponseTime">
85         <wsdl:input name="LastResponseTime" message="wsc:LastResponseTime"/>
86         <wsdl:output name="LastResponseTimeResponse" message="wsc:LastResponseTimeResponse"/>
87     </wsdl:operation>
88     </wsdl:portType>
89     <wsdl:portType>
90         <wsdl:binding name="MonitoringBinding" type="wsc:MonitoringPort">
91             <wsdlsoap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
92             <wsdl:operation name="NumberOfRequests">
93                 <wsdlsoap:operation soapAction=""/>
94                 <wsdl:input name="NumberOfRequests">
95                     <wsdlsoap:body use="literal"/>
96                 </wsdl:input>
97                 <wsdl:output name="NumberOfRequestsResponse">
98                     <wsdlsoap:body use="literal"/>
99                 </wsdl:output>
100            </wsdl:operation>
101            <wsdl:operation name="NumberOfFailedRequests">
102                <wsdlsoap:operation soapAction=""/>
103                <wsdl:input name="NumberOfFailedRequests">
104                    <wsdlsoap:body use="literal"/>
105                </wsdl:input>
106                <wsdl:output name="NumberOfFailedRequestsResponse">
107                    <wsdlsoap:body use="literal"/>
108                </wsdl:output>
109            </wsdl:operation>
110            <wsdl:operation name="NumberOfSuccessfulRequests">
111                <wsdlsoap:operation soapAction=""/>
112                <wsdl:input name="NumberOfSuccessfulRequests">
113                    <wsdlsoap:body use="literal"/>
114                </wsdl:input>
115                <wsdl:output name="NumberOfSuccessfulRequestsResponse">
116                    <wsdlsoap:body use="literal"/>
117                </wsdl:output>
118            </wsdl:operation>
119            <wsdl:operation name="ServiceTime">
120                <wsdlsoap:operation soapAction=""/>
121                <wsdl:input name="ServiceTime">
122                    <wsdlsoap:body use="literal"/>
123                </wsdl:input>
124                <wsdl:output name="ServiceTimeResponse">
125                    <wsdlsoap:body use="literal"/>
126                </wsdl:output>
127            </wsdl:operation>
128            <wsdl:operation name="MaxResponseTime">
129                <wsdlsoap:operation soapAction=""/>
130                <wsdl:input name="MaxResponseTime">
131                    <wsdlsoap:body use="literal"/>
132                </wsdl:input>
133                <wsdl:output>
134            </wsdl:operation>
<wsdl:input/>
<wsdl:output name="MaxResponseTimeResponse">
<wsdlsoap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="LastResponseTime">
<wsdlsoap:operation soapAction=""/>
<wsdl:input name="LastResponseTime">
<wsdlsoap:body use="literal"/>
</wsdl:input>
<wsdl:output name="LastResponseTimeResponse">
<wsdlsoap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="MonitoringService">
<wsdl:port name="MonitoringPort" binding="wsc:MonitoringBinding">
<wsdlsoap:address location="..."/>
</wsdl:port>
</wsdl:service>
</wsdl:definitions>