



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Εισαγωγή στα Δίκτυα Υπηρεσιών

Διάλεξη 17η: WSRF

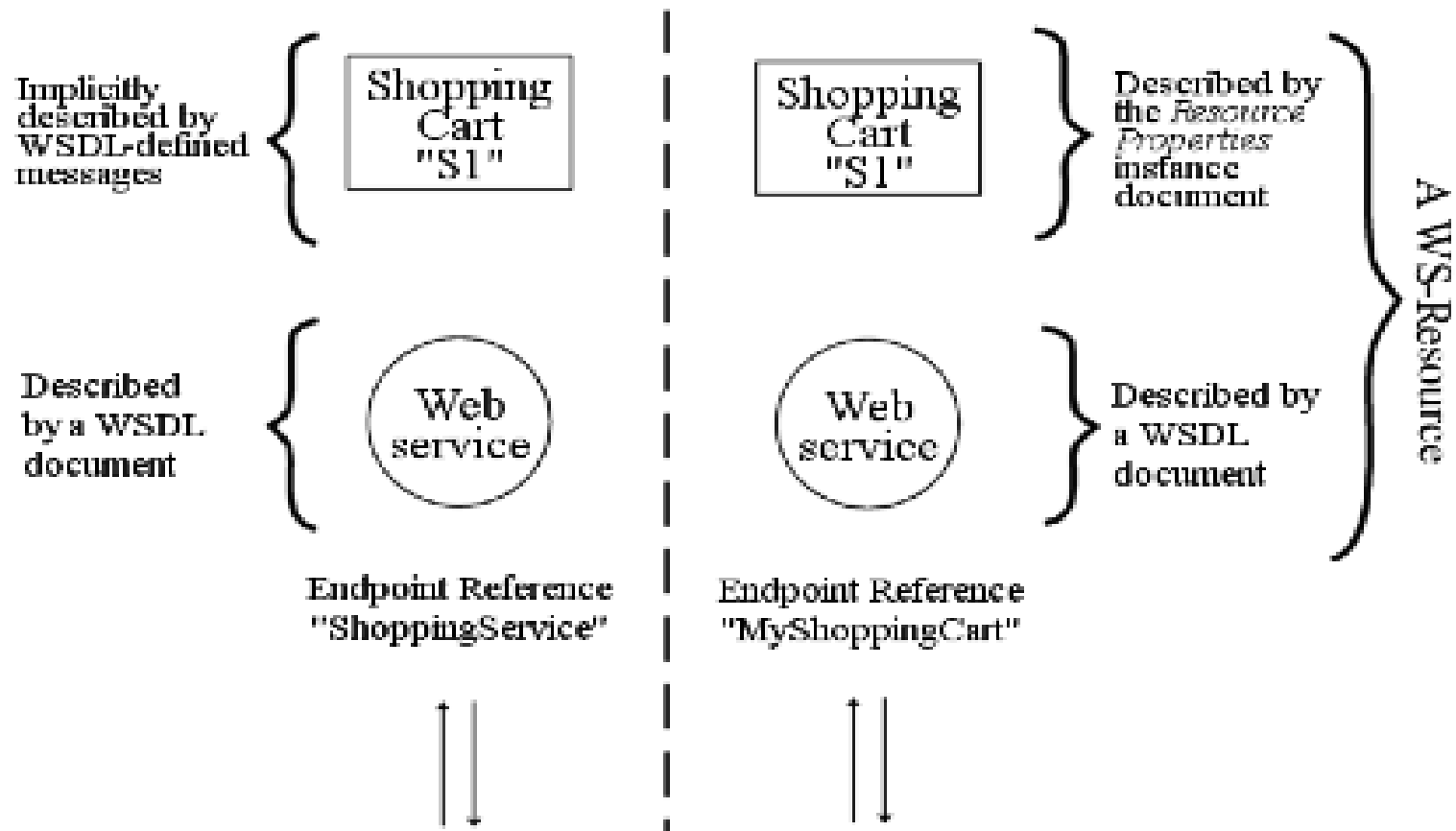
Χρήστος Νικολάου
Τμήμα Επιστήμης Υπολογιστών

Introduction to Service Networks

The Shopping Cart and Printer WSRF Examples Source: WSRF Primer

Christos Nikolaou
Mariana Karmazi
Transformation Services Laboratory
CSD/UoC

Non-WSRF Web Services and WSRF



Source: WSRF Primer

Non WSRF vs WSRF cont...

Previous Figure shows a comparison of a simple shopping service built with non-WSRF Web services (on the left) and WSRF (on the right). The left-hand side of the figure shows a ShoppingCart resource called 'S1' accessed through a Web service at a particular location. The location is described by a WS-Addressing [[WS-Addressing](#)] EndpointReference (EPR), which contains the URI of the Web service endpoint as follows:

```
<wsa:EndpointReference>  
  <wsa:Address>http://www.example.com/WSSimpleShoppingService</wsa:Address>  
</wsa:EndpointReference>
```

This EPR is known as “ShoppingService” by the requester.

The End Point Reference (EPR)

EPRs and ports

The use of an EndpointReference (EPR) to describe the location of a Web service is a recent innovation in Web services architecture. Just like the port in the service element of WSDL, the EPR contains a URI which indicates the location of the Web service.

We describe Web services locations using EPRs because this technique provides the clearest comparison with the mechanisms introduced by WSRF.

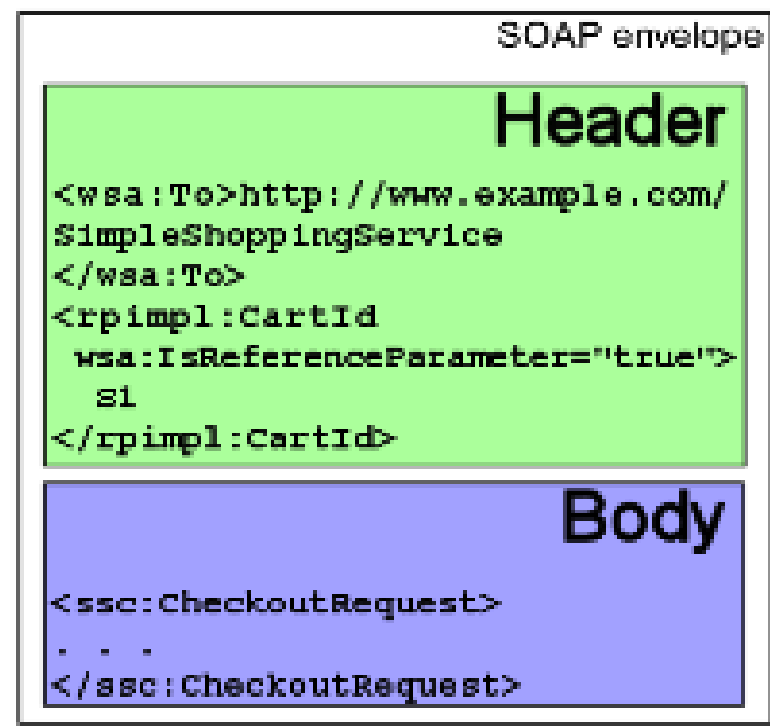
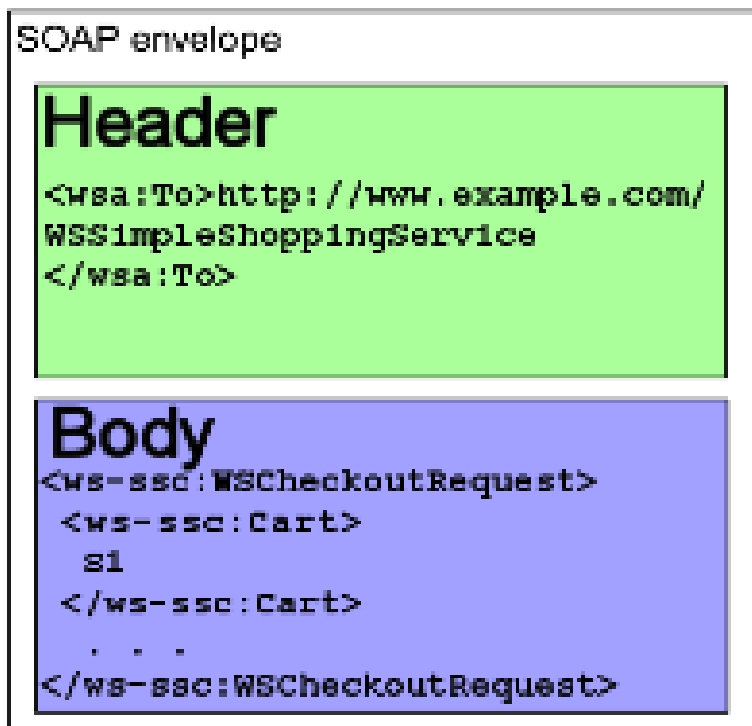
Addressing The Resource

- Arrows in the previous figure represent messages sent between the requester and the Web service.
- The mechanism by which the specific ShoppingCart 'S1' is identified amongst the many carts that may be accessed through the service at the same location is **not shown**:
 - the ShoppingCart identifier must be present somewhere in the messages and is typically passed as a parameter to the Web service and described in the WSDL.
- WSRF avoids the need to describe the identifier explicitly in the WSDL description.
- The identifier can be packaged as part of the EPR and implicitly included in all messages addressed through the EPR according to the rules of WS-Addressing.
- The resource identifier is not of interest to the requesting application: it uses the EPR as a reference to a specific WS-Resource. The right-hand side of Figure 1 shows the same service in the WSRF view:
 - a combined Web service and ShoppingCart described by a Resource Properties document are together known as a WS-Resource and referenced by the requester as "MyShoppingCart". The EPR might look like this:

The EPR of the combined Shopping Cart Resource

```
<wsa:EndpointReference>  
  <wsa:Address>  
    http://www.example.com/SimpleShoppingService  
  </wsa:Address>  
  <wsa:ReferenceParameters>  
    <rpimpl:CartId>S1</rpimpl:CartId>  
  </wsa:ReferenceParameters>  
</wsa:EndpointReference>
```

The Shopping Cart Example – SOAP Messages



Source: WSRF Primer

The Advantages of Defining a WS-Resource

By removing the resource identifier from the body of the messages, WSRF simplifies the form of the WSDL that describes the service.

The WSDL for a message that, for example, adds an item to a cart need only be concerned about the item to be added; the requesting application is not responsible for maintaining the identity of the cart to which the item must be added.

WSRF provides standardized forms for messages which interact with a WS-Resource, namely:

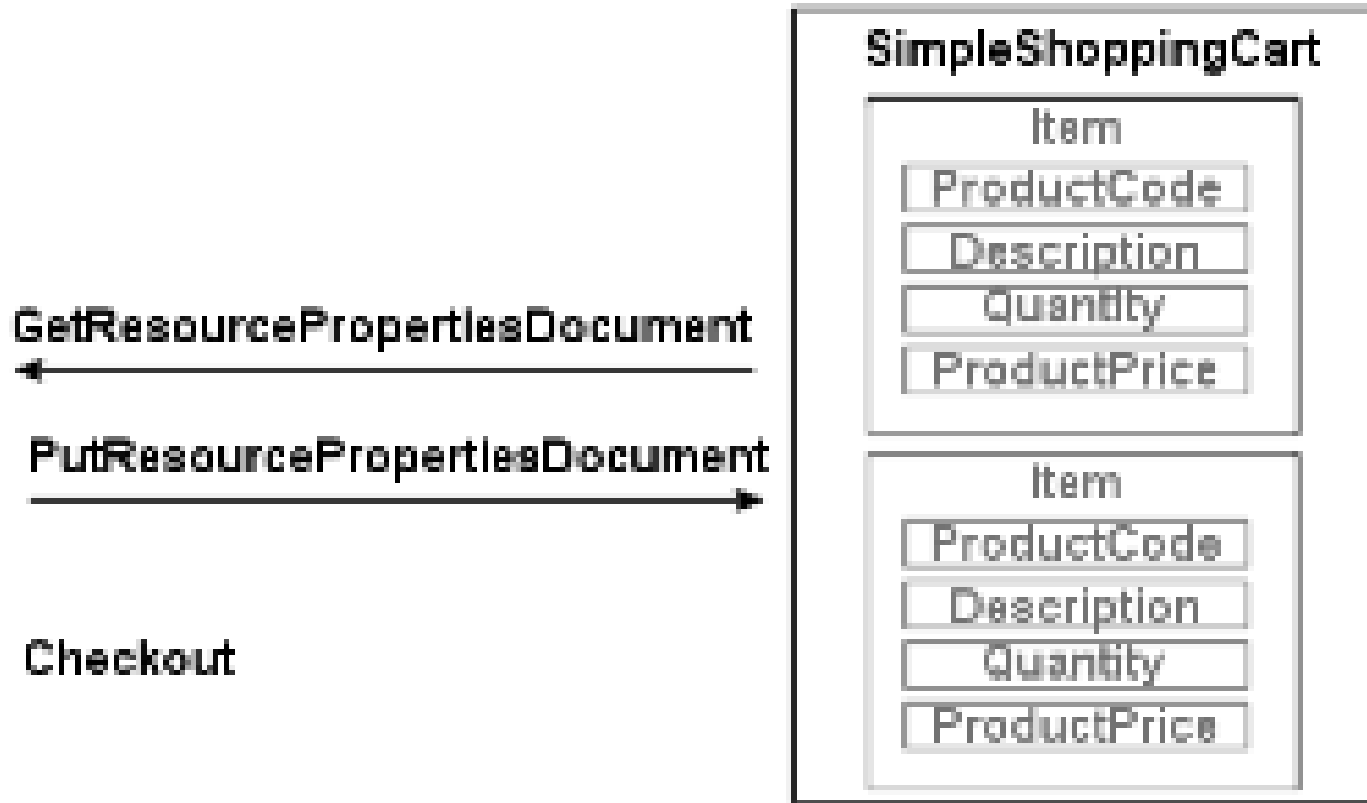
- A way to retrieve and replace the Resource Properties document describing the resource.
- Several ways to retrieve and replace named parts of the document.

In addition, client applications which must deal with multiple resources are simplified if the resource definitions and operations follow the same pattern. In particular, this is advantageous to resource management systems.

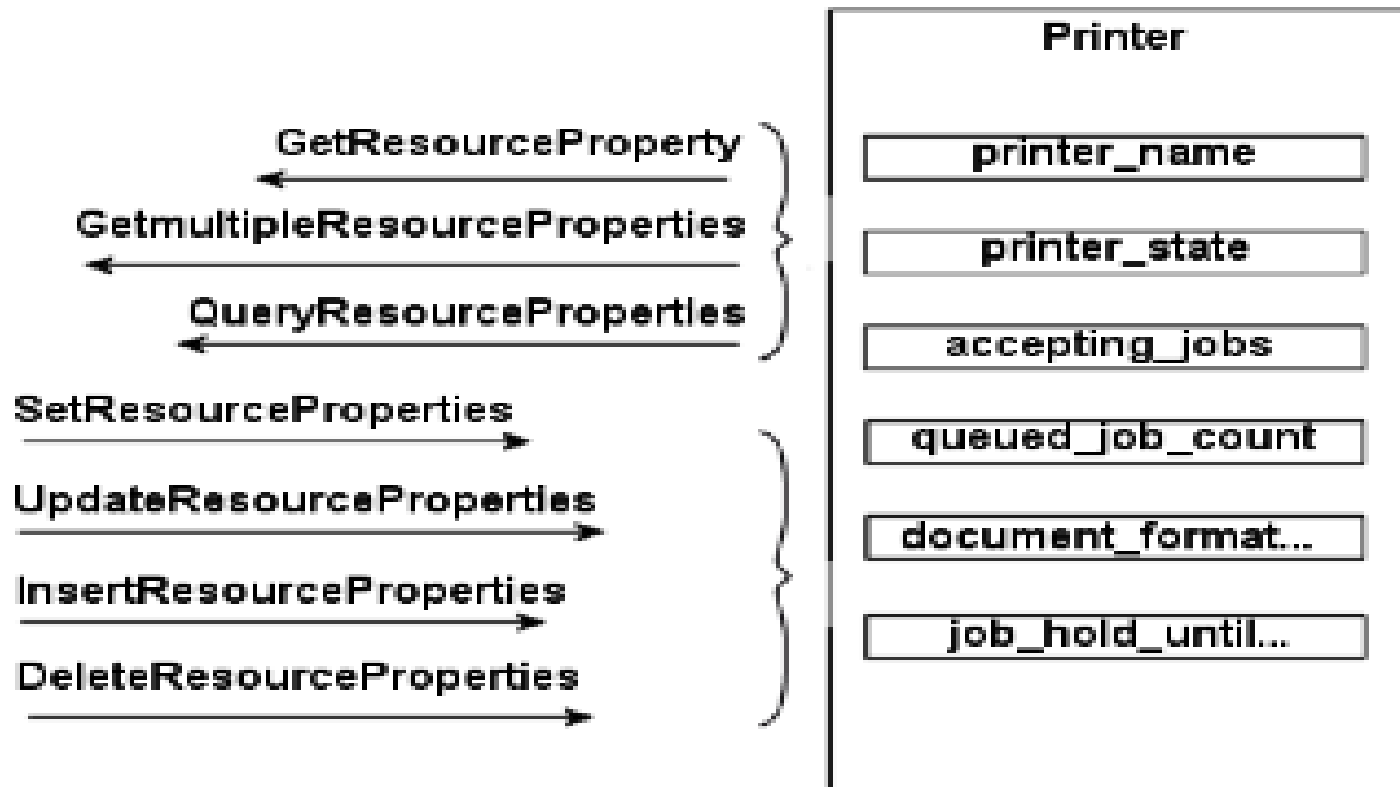
WSRF also provides standard solutions to the following issues which commonly arise in distributed systems :

- A way to describe and control the lifetime of a WS-Resource.
- The systematic description and handling of fault messages.
- The aggregation of information about resources and services to form directories.
- The description of messages which may publish, to interested parties, changes in state of WS-Resources, including changes their property values.

The Shopping Cart Example – The Resource



The Printer as a Resource: A Collection of Properties



The Simple Shopping Service



The WSDL Description without WSRF

PortType: WSSimpleShoppingCartCreation

Operation: WSCreateCart

input

output

PortType: WSSimpleShoppingCart

Operation: WSGetCart

input

output

WSCartUnknownFault

Operation: WSPutCart

input

output

WSCartUnknownFault

Operation: CartCheckout

input

output

WSCartUnknownFault

WSBillingFault

SOAP messages for creating a Shopping Cart without WSRF

<SOAP-ENV:Header>

...

<wsa:To SOAP-ENV:mustUnderstand="1">

http://www.example.com/WSSimpleShoppingService

</wsa:To>

</SOAP-ENV:Header>

<SOAP-ENV:Body>

<ws-ssc:WSCartCreateRequest>

<ws-ssc:ProductCode>Cat-A2004-87968556</ws-ssc:ProductCode>

<ws-ssc:Quantity>1</ws-ssc:Quantity>

</ws-ssc:WSCartCreateRequest>

</SOAP-ENV:Body>

And the Response SOAP message without WSRF

```
<SOAP-ENV:Body>  
  <ws-ssc:WSCartCreateResponse  
    <ws-ssc:Cart>S1</ws-ssc:Cart>  
    <ws-ssc:ServiceAddress>  
      <wsa:Address>http://example.com/ShoppingService</wsa:Address>  
    </ws-ssc:ServiceAddress>  
  </ws-ssc:WSCartCreateResponse>  
</SOAP-ENV:Body>
```

Subsequent Requests/Responses without WSRF

```
<ws-ssc:WSGetCart>
  <ws-ssc:Cart>S1</ws-ssc:Cart>
</ws-ssc:WSGetCart>
<SOAP-ENV:Body>
  <ws-ssc:WSGetCartResponse>
    <ws-ssc:WSSimpleShoppingCart>
      <ssc:Item>
        <ssc:ProductCode>Cat-A2004-87968556</ssc:ProductCode>
        <ssc:Description>Garden String - 150m</ssc:Description>
        <ssc:Quantity>1</ssc:Quantity>
        <ssc:ProductPrice>1.59</ssc:ProductPrice>
      </ssc:Item>
    </ws-ssc:WSSimpleShoppingCart>
  </ws-ssc:WSGetCartResponse>
</SOAP-ENV:Body>
```


The Shopping Cart Example with WSRF

WSDL definitions

PortType: SimpleShoppingCartCreation

Operation: CreateCart
input
output

PortType: SimpleShoppingCart

Operation: GetResourcePropertyDocument
input
output
ResourceUnknownFault
ResourceUnavailableFault

Operation: PutResourcePropertyDocument
input
output
ResourceUnknownFault
ResourceUnavailableFault

Operation: GetResourceProperty
input
output
InvalidResourcePropertyQNameFault
ResourceUnknownFault
ResourceUnavailableFault

Operation: Checkout
input
output
CheckoutFault
ResourceUnknownFault
ResourceUnavailableFault

ResourceProperties schema, root element declaration

```
ssc:SimpleShoppingCart
├── ssc:Item
│   ├── ssc:ProductCode
│   ├── ssc:Description
│   ├── ssc:Quantity
│   └── ssc:ProductPrice
```

The WSDL definitions with WSRF

```
<wsdl:definitions>
  <wsdl:import
    namespace="http://docs.oasis-open.org/wsrf/rpw-2"
    location=" http://docs.oasis-open.org/wsrf/rpw-2.wsdl"/>
  ...
  <wsdl:types>
    <xsd:schema . . .>
      <xsd:import . . .
        schemaLocation="./SimpleShoppingCart.xsd" />
    </xsd:schema>
  </wsdl:types>
  ...
```

Cont.

```
<wsdl:portType name="SimpleShoppingCart"
  wsrf-rp:ResourceProperties="ssc:SimpleShoppingCart">
  <wsdl:operation name="GetResourcePropertyDocument">
    <wsdl:input message="wsrf-rpw:GetResourcePropertyDocumentRequest" . . />
    <wsdl:output message="wsrf-rpw:GetResourcePropertyDocumentResponse" .
  ./>
  ...
  </wsdl:operation>
  ...
  </wsdl:portType>
</wsdl:definitions>
```

Operations on the Shopping Cart with WSRF

Same as before CreateCartRequest. However:

```
<ssc:CartCreateResponse>  
  <wsa:Address>http://www.example.com/SimpleShoppingService</wsa:Address>  
  <wsa:ReferenceParameters>  
    <rpimpl:CartId>S1</rpimpl:CartId>  
  </wsa:ReferenceParameters>  
</ssc:CartCreateResponse>
```

To see what is stored in the Shopping Cart (Request Message Header):

```
<SOAP-ENV:Header>  
...  
  <wsa:To SOAP-ENV:mustUnderstand="1">  
    http://www.example.com/SimpleShoppingService  
  </wsa:To>  
  <rpimpl:CartId wsa:IsReferenceParameter="true">S1</rpimpl:CartId>  
</SOAP-ENV:Header>
```

Cont.

<SOAP-ENV:Body>

<wsrf-rp:GetResourcePropertyDocument/>

</SOAP-ENV:Body> *and here comes the response*

<SOAP-ENV:Body>

<wsrf-rp:GetResourcePropertyDocumentResponse>

<ssc:SimpleShoppingCart>

<ssc:Item>

<ssc:ProductCode>Cat-A2004-87968556</ssc:ProductCode>

<ssc:Description>Garden String - 150m</ssc:Description>

<ssc:Quantity>1</ssc:Quantity>

<ssc:ProductPrice>1.59</ssc:ProductPrice>

</ssc:Item>

</ssc:SimpleShoppingCart>

</wsrf-rp:GetResourcePropertyDocumentResponse>

</SOAP-ENV:Body>

Updating a shopping Cart Item

```
<SOAP-ENV:Body>
  <wsrf-rp:PutResourcePropertyDocumentRequest>
    <ssc:SimpleShoppingCart>
      <ssc:Item>
        <ssc:ProductCode>Cat-A2004-87968556</ssc:ProductCode>
        <ssc:Description>Garden String - 150m</ssc:Description>
        <ssc:Quantity>2</ssc:Quantity>
        <ssc:ProductPrice>1.59</ssc:ProductPrice>
      </ssc:Item>
    </ssc:SimpleShoppingCart>
  </wsrf-rp:PutResourcePropertyDocumentRequest>
</SOAP-ENV:Body>
<SOAP-ENV:Body>
  <wsrf-rp:PutResourcePropertyDocumentResponse/>
</SOAP-ENV:Body>
```

The BaseFaultType

<wsrf-bf:BaseFault>

<wsrf-bf:Timestamp>2005-05-31T12:00:00.000Z</wsrf-bf:Timestamp>

<wsrf-bf:Originator>

<wsa:Address>http://www.example.com/SimpleShoppingService</wsa:Address>

<wsa:ReferenceParameters>

<rpimpl:CartId>S1</rpimpl:CartId>

</wsa:ReferenceParameters>

</wsrf-bf:Originator>

<wsrf-bf:Description>Descriptive Text</wsrf-bf:Description>

</wsrf-bf:BaseFault>

Customized Fault Types

```
<xsd:complexType name="CheckoutFaultType">
  <xsd:complexContent>
    <xsd:extension base="wsrf-bf:BaseFaultType">
      <xsd:sequence>
        <xsd:element name="CheckoutFaultDetails" type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```


Cont.

```
<ssc:CheckoutFault>
  <wsrf-bf:Timestamp>2005-05-31T12:00:00.000Z</wsrf-bf:Timestamp>
  <wsrf-bf:Originator>
    <wsa:Address>http://www.example.com/SimpleShoppingService</wsa:Address>
    <wsa:ReferenceParameters>
      <rpimpl:CartId>S1</rpimpl:CartId>
    </wsa:ReferenceParameters>
  </wsrf-bf:Originator>
  <wsrf-bf:Description>Faulty Checkout Information</wsrf-bf:Description>
  <ssc:CheckoutFaultDetails>Credit Limit Exceeded!</ssc:CheckoutFaultDetails>
</ssc:CheckoutFault>
```

The Universal Faults

Two standard faults are defined for use with every WSRF operation.
These are:

ResourceUnknownFault

- Used to indicate that the WS-Resource is not known by the service which receives the message.

ResourceUnavailableFault

- Used to indicate that the Web service is active, but unable to provide access to the resource.

Example of Resource Unknown Fault

```
<SOAP-ENV:Body>
```

```
<SOAP-ENV:Fault>
```

```
  <faultcode>SOAP-ENV:Client</faultcode>
```

```
  <faultstring>No such resource exists</faultstring>
```

```
  <faultactor>http://www.example.com/SimpleShoppingService</faultactor>
```

```
  <detail>
```

```
    <wsrf-r:ResourceUnknownFault>
```

```
      <wsrf-bf:Timestamp>2005-05-31T12:00:00.000Z</wsrf-bf:Timestamp>
```

```
      <wsrf-bf:Originator>
```

```
        <wsa:Address>http://www.example.com/SimpleShoppingService</wsa:Address>
```

```
        <wsa:ReferenceParameters>
```

```
          <rpimpl:CartId>S0</rpimpl:CartId>
```

```
        </wsa:ReferenceParameters>
```

```
      </wsrf-bf:Originator>
```

```
      <wsrf-bf:Description>Resource unknown</wsrf-bf:Description>
```

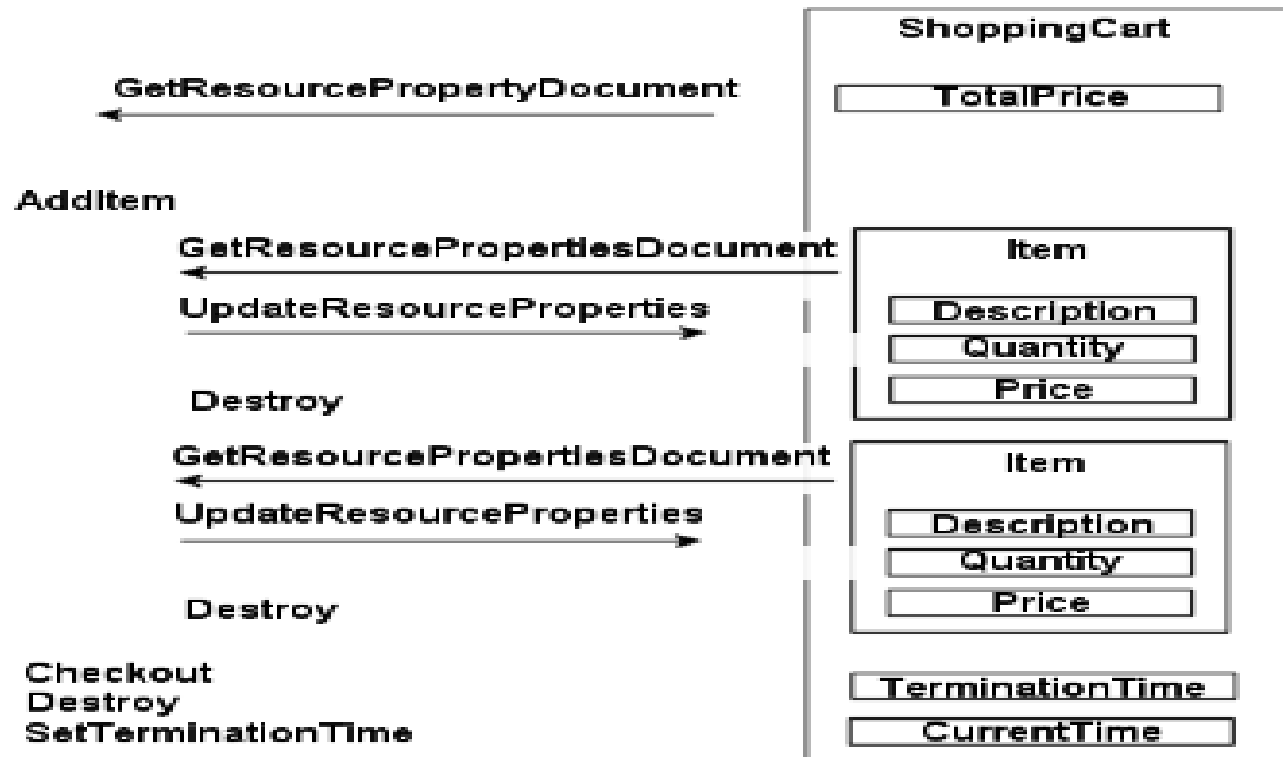
```
    </wsrf-r:ResourceUnknownFault>
```

```
  </detail>
```

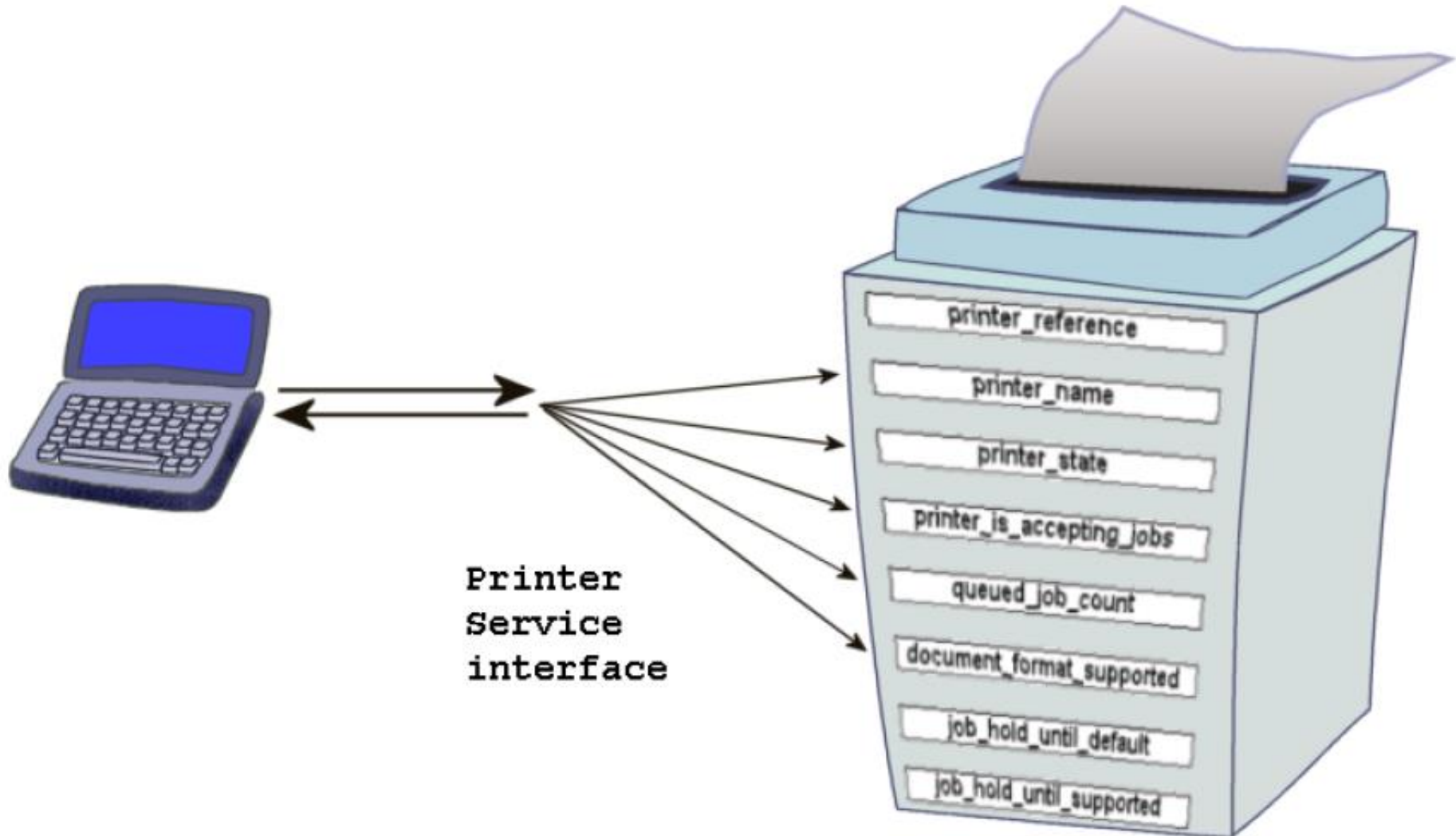
```
</SOAP-ENV:Fault>
```

```
</SOAP-ENV:Body>
```

The Shopping Cart as multiple WS Resources



The Printer Resource



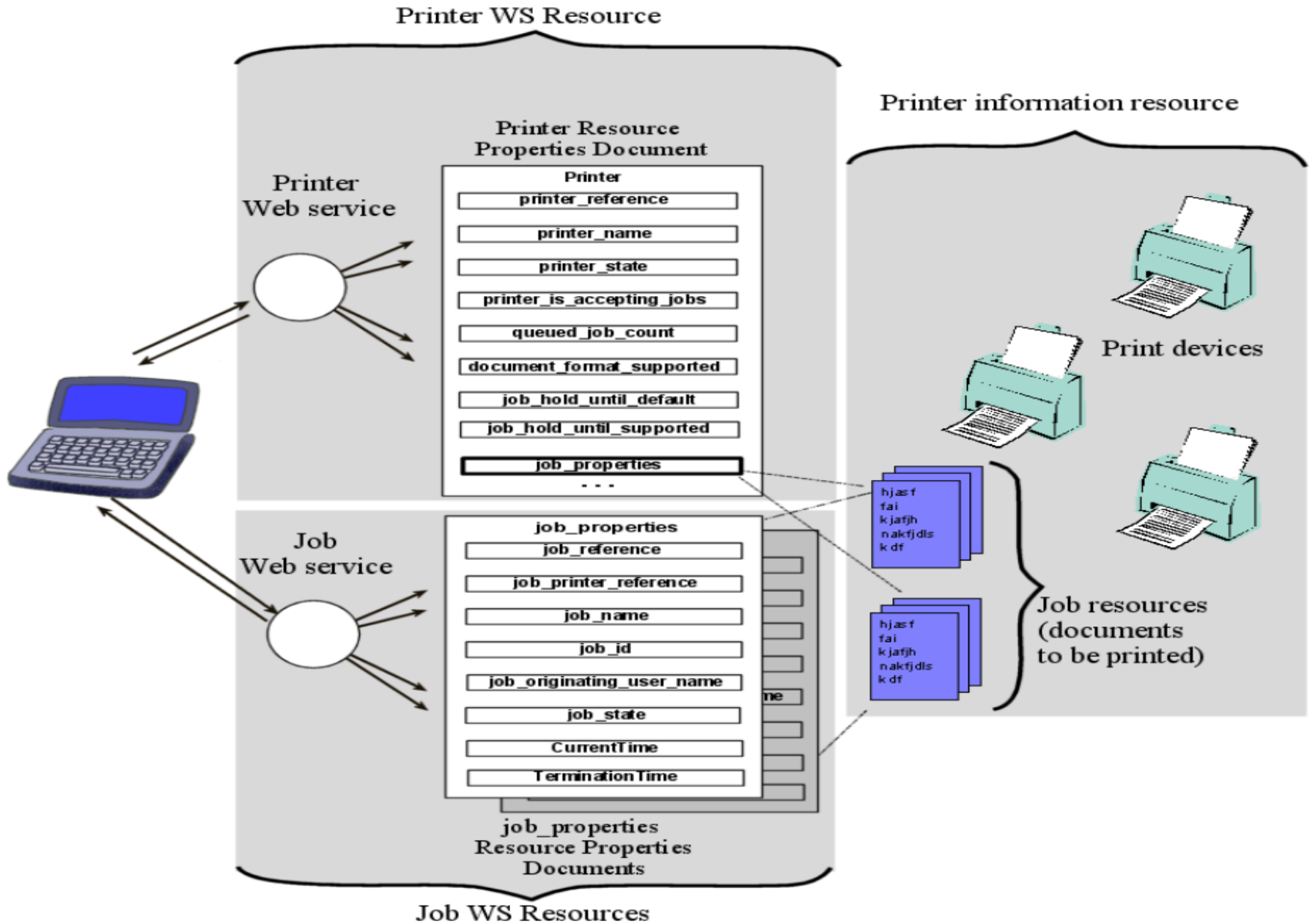
The Printer Resource Properties

Element name	Element value
<code>pr:Printer</code>	
<code>pr:printer_reference</code>	
<code>wsa:Address</code>	<code>http://www.example.com/Printer</code>
<code>wsa:ReferenceParameters</code>	<code><rpimpl:PrinterId>Printer01 </rpimpl:PrinterId></code>
<code>pr:printer_name</code>	<code>Printer01</code>
<code>pr:printer_state</code>	<code>Idle</code>
<code>pr:printer_is_accepting_jobs</code>	<code>true</code>
<code>pr:queued_job_count</code>	<code>0</code>
<code>pr:document_format_supported</code>	
<code>pr:mimeType</code>	<code>text/plain</code>
<code>pr:job_hold_until_default</code>	<code>No-Hold</code>
<code>pr:job_hold_until_supported</code>	<code>No-Hold</code>
<code>pr:job_hold_until_supported</code>	<code>Overnight</code>

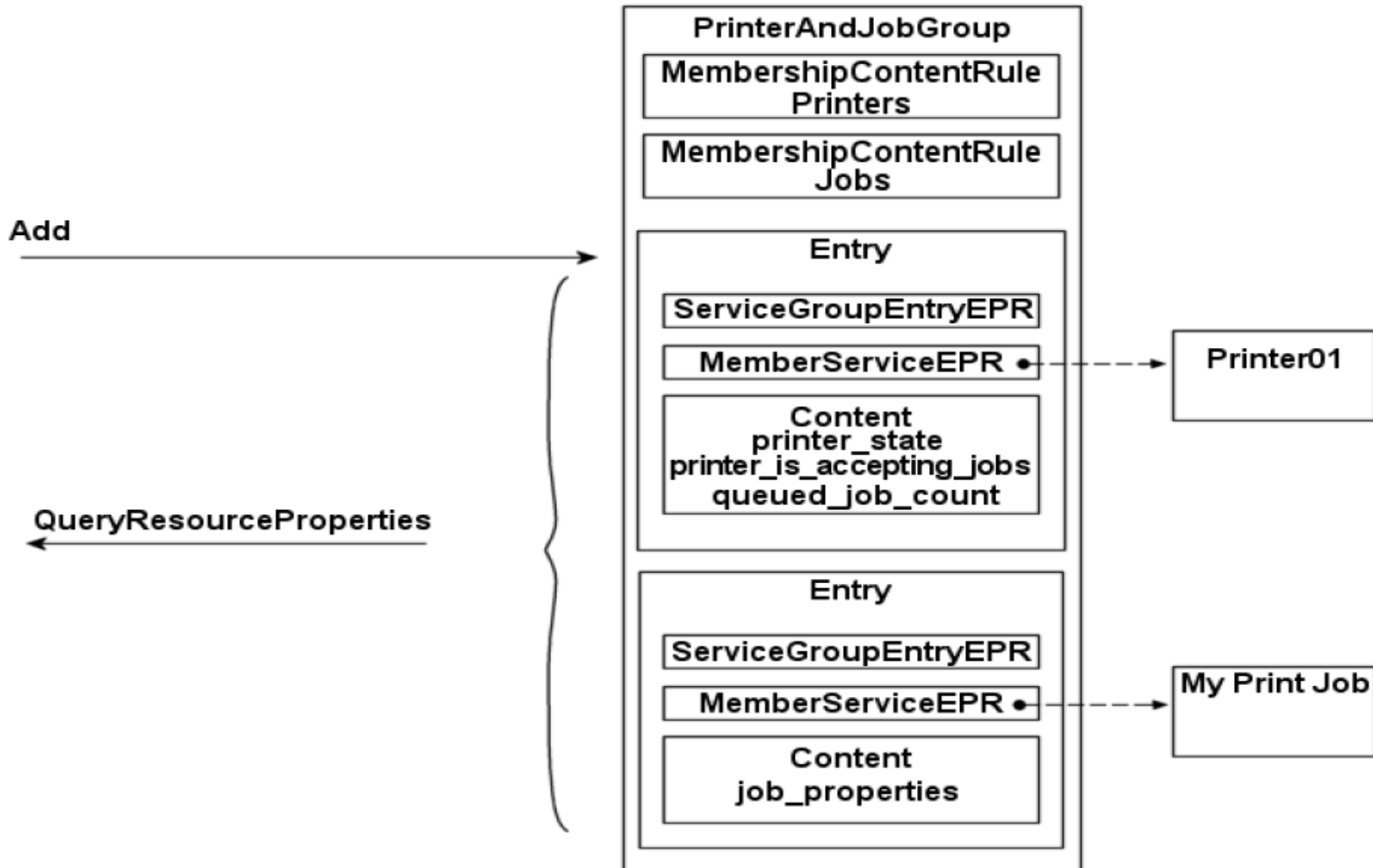
Printer Properties Multiplicities

PropertyName	Multiplicity
pr:printer_reference	1
pr:printer_name	1
pr:printer_state	1
pr:printer_is_accepting_jobs	1
pr:queued_job_count	1
pr:document_format_supported	1
pr:job_hold_until_default	0..1
pr:job_hold_until_supported	0..*
wsrf-rp:QueryExpressionDialect	0..*
pr:job_properties	0..*

WS-Resources as Views of Underlying Resources



Directories and Groups of Service



Service Groups cont.

The directory is described by a WSDL portType called PrinterAndJobGroup which contains the operations to add new entries and query existing entries using QueryResourceProperties.

The Resource Properties document is shown in the figure; it contains the entries, along with rules which describe them.

MembershipContentRules

```
<wsrf-sg:MembershipContentRule
  MemberInterfaces="preprinted"
  ContentElements="pr:printer_state
                  pr:printer_is_accepting_jobs
                  pr:queued_job_count"
/>
```

Member services which support the pr:Printer portType must be described by an entry which contain three XML elements defined by the printer

Updating the Entries: Notification Messages

To be useful, the entries in the PrinterAndJobGroup must contain up-to-date information about the state of the Printers and Jobs as they change. Interfaces which define the publication of such information to interested subscribers are defined by Web service specifications such as WS-BaseNotification [[WS-BaseNotification](#)].

In our example, the implementation of the Job WS-Resource must act as a publisher and allow the JobEntry WS-Resource to subscribe to changes in the values of the resource properties describing the job resource.

WSRF describes the format of several kinds of events which can be incorporated in notification messages defined by WS-BaseNotification. These are:

- Changes to resource property values defined in [[WS-Resource Properties](#)]
- WS-Resource termination, defined in [[WS-Resource Lifetime](#)]
- Addition and removal of ServiceGroupEntries, defined in [[WS-ServiceGroup](#)]

Example Notification Message

```
<wsnt:Notify>
```

```
<wsnt:NotificationMessage>
```

```
<wsnt:SubscriptionReference>(details omitted)</wsnt:SubscriptionReference>
```

```
<wsnt:Topic
```

```
  Dialect="http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
```

```
    pr:job_state
```

```
</wsnt:Topic>
```

```
<wsnt:ProducerReference>
```

```
<wsa:Address>http://www.example.com/Printer/Job</wsa:Address>
```

```
<wsa:ReferenceParameters>
```

```
<rpimpl:PrinterId>Printer01</rpimpl:PrinterId>
```

```
<rpimpl:JobId>Job523</rpimpl:JobId>
```

```
</wsa:ReferenceParameters>
```

```
</wsnt:ProducerReference>
```

Cont.

```
<wsnt:Message>
  <pr:Job_stateChangeNotification>
    <pr:printer_name>Printer01</pr:printer_name>
    <pr:job_name>My Print Job</pr:job_name>
    <wsrf-rp:ResourcePropertyValueChangeNotification>
      <wsrf-rp:OldValues>
        <pr:jobprinter_state>processing</pr:job_state>
      </wsrf-rp:OldValues>
      <wsrf-rp:NewValues>
        <pr:job_state>completed</pr:job_state>
      </wsrf-rp:NewValues>
    </wsrf-rp:ResourcePropertyValueChangeNotification>
  </pr:Job_stateChangeNotification>
</wsnt:Message>
<wsnt:NotificationMessage>
</wsnt:Notify>
```

Τέλος Ενότητας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «**Εκπαίδευση και Δια Βίου Μάθηση**» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Σημειώματα

Σημείωμα αδειοδότησης

- Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/>

•Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

•Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Κρήτης, Χρήστος Νικολαου. «**Εισαγωγή στα Δίκτυα Υπηρεσιών. Διάλεξη 17η: WSRF**». Έκδοση: 1.0. Ηράκλειο/Ρέθυμνο 2015.
Διαθέσιμο από τη δικτυακή διεύθυνση: <https://elearn.uoc.gr/course/view.php?id=416/>