

WS-Coordination Framework Demo Application

Working Draft 01, 3 August 2005

Document identifier:

ws_caf_implement-ws_coordination_framework-demo-draft-01

Location:

http://www.oasis-open.org/committees/documents.php?wg_abbrev=ws-caf-implement

Editors:

Kevin Conner, Arjuna Technologies

Gary Tully, Iona

John Fuller, Wernervas

Abstract:

This document describes an application that demonstrates the coordination of an activity group through the creation, manipulation and propagation of a WS-CAF Coordination Framework [WSCF] context. The application extends the WS-CAF Context Demo Application [WSCTXDEMO] to include a simple notification mechanism for participants registered with the activity group.

Status:

Committee members should send comments on this document to the ws-caf-implement@lists.oasis-open.org list. Others should use the comment form at http://www.oasis-open.org/committees/comments/form.php?wg_abbrev=ws-caf-implement.

1 Table of Contents

2	Terminology.....	3
3	2.1 Namespace.....	3
4	2.1.1 Prefix Namespace.....	3
5	3 Overview.....	4
6	4 Schema Document.....	5
7	5 Context Service.....	6
8	5.1 Context examples.....	6
9	6 Registration Service.....	8
10	7 Notification Service.....	9
11	8 Participant Service.....	10
12	9 Retailer Service.....	11
13	10 Web Store.....	12
14	11 References.....	14
15	11.1 Normative References.....	14
16	12 Acknowledgements.....	15

2 Terminology

The key words *must*, *must not*, *required*, *shall*, *shall not*, *should*, *should not*, *recommended*, *may*, and *optional* in this document are to be interpreted as described in IETF RFC 2119 [RFC2119].

2.1 Namespace

The XML namespace URI that *must* be used by implementations of this specification is:

<http://docs.oasis-open.org/wscaf/2005/08/wscfdemo>

2.1.1 Prefix Namespace

The following table contains the namespace prefixes used within this document.

Prefix	Namespace
wscfdemo	http://docs.oasis-open.org/wscaf/2005/08/wscfdemo
wsctxdemo	http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo
wsctx	http://docs.oasis-open.org/wscaf/2005/06/wsctx
wscf	http://docs.oasis-open.org/wscaf/2005/07/wscf
xsd	http://www.w3.org/2001/XMLSchema
wsdl	http://schemas.xmlsoap.org/wsdl
soap	http://schemas.xmlsoap.org/wsdl/soap
wsa	http://schemas.xmlsoap.org/ws/2004/08/addressing
tns	http://docs.oasis-open.org/wscaf/2005/08/wscfdemo

3 Overview

This document specifies an application used to demonstrate the creation, manipulation and propagation of a WS-CAF Coordination Framework context. It demonstrates a customer making purchases from a number of retailers via an online store using a notification mechanism to inform registered participants of significant events. The retailer services are provided by different implementations of this application.

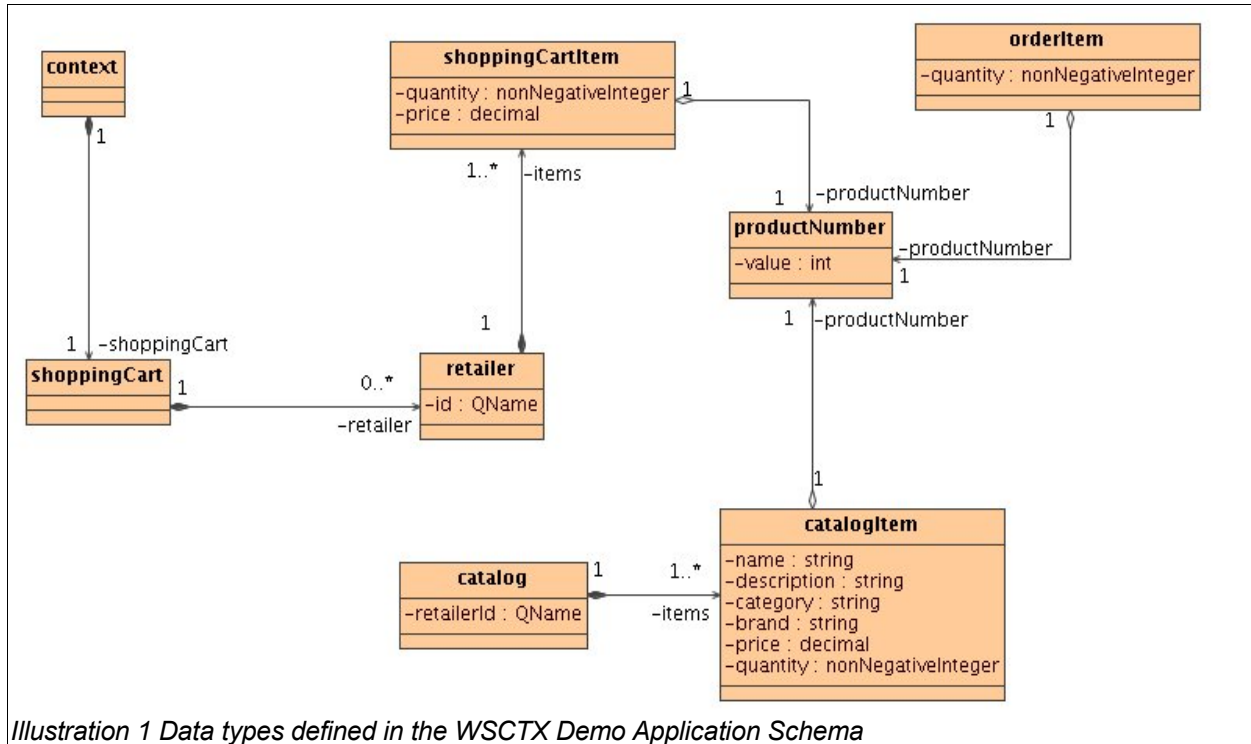
The specification of the application is divided into seven sections

- The XML Schema documents describing the elements used within the application context and the application WSDL messages.
- The Context Service for creating and destroying the context.
- The Registration Service for managing activity group participants.
- The Notification Service for initiating activity group protocol notifications.
- The Participant service for receiving activity group protocol notifications.
- The Retailer Service supplying items for purchase by a Store.
- The Web based Store allowing a user to purchase items from the Retailer services.

This version of the demo application does not support context propagation by reference. It is intended to add this functionality at a later date.

4 Schema Document

This application is an extension of the WS-CAF Context Demo Application [WSCTXDEMO] and, as such, makes use of the application types and elements defined within that document. The schema is extended to define a WS-CAF Coordination Framework [WSCF] context and a qualified name for the application context.



The contents of the WSCF Demo Application Schema are included in the appendix.

5 Context Service

The Context Service is an implementation of the Context Service as specified in the Web Services Context Specification [WSCTX]. The Context Service is responsible for the lifecycle management of a context, specifically the context identified by the qualified name `wscfdemo:context`, as specified in the appendix.

All Context Service communication adheres to the normative one way binding as specified in [WSCTX]. Service endpoints used as part of the Context Service communication are identified using the WS-Addressing specification [WSADDR].

5.1 Context examples

The initial context created by the Context Service *must* contain an empty shopping cart.

```
<wscfdemo:context>
  <wsctx:context-identifier>
    http://www.arjuna.com/wscafdemo/context/12345
  </wsctx:context-identifier>
  <wsctx:context-service>
    <wsa:EndpointReference>
      <wsa:Address>http://www.arjuna.com/demoContextService</wsa:Address>
    </wsa:EndpointReference>
    </wsctx:context-service>
    <wscf:registration-service>
      <wsa:EndpointReference>
        <wsa:Address>http://www.arjuna.com/demoRegistrationService</wsa:Address>
      </wsa:EndpointReference>
    </wscf:registration-service>
    <wscfdemo:notification-service>
      <wsa:EndpointReference>
        <wsa:Address>http://www.arjuna.com/demoNotificationService</wsa:Address>
      </wsa:EndpointReference>
    </wscfdemo:notification-service>
    <wsctxdemo:shoppingCart/>
  </wscfdemo:context>
```

The context will be extended as the customer chooses products from each store.

```
<wscfdemo:context>
  <wsctx:context-identifier>
    http://www.arjuna.com/wscafdemo/context/12345
  </wsctx:context-identifier>
  <wsctx:context-service>
    <wsa:EndpointReference>
      <wsa:Address>http://www.arjuna.com/demoContextService</wsa:Address>
    </wsa:EndpointReference>
    </wsctx:context-service>
    <wscf:registration-service>
      <wsa:EndpointReference>
        <wsa:Address>http://www.arjuna.com/demoRegistrationService</wsa:Address>
      </wsa:EndpointReference>
    </wscf:registration-service>
    <wscfdemo:notification-service>
      <wsa:EndpointReference>
        <wsa:Address>http://www.arjuna.com/demoNotificationService</wsa:Address>
      </wsa:EndpointReference>
    </wscfdemo:notification-service>
    <wsctxdemo:shoppingCart>
      <wsctxdemo:retailer>
        <wsctxdemo:id>arjuna:retail1</wsctxdemo:id>
        <wsctxdemo:item>
          <wsctxdemo:productNumber>12345<wsctxdemo:productNumber>
          <wsctxdemo:quantity>2<wsctxdemo:quantity>
          <wsctxdemo:price>10.55<wsctxdemo:price>
        </wsctxdemo:item>
        <wsctxdemo:item>
          <wsctxdemo:productNumber>314<wsctxdemo:productNumber>
          <wsctxdemo:quantity>1<wsctxdemo:quantity>
          <wsctxdemo:price>4.99<wsctxdemo:price>
        </wsctxdemo:item>
      </wsctxdemo:retailer>
      <wsctxdemo:retailer>
        <wsctxdemo:id>arjuna:retail2</wsctxdemo:id>
```

```
60      <wsctxdemo:item>
61          <wsctxdemo:productNumber>12345<wsctxdemo:productNumber>
62          <wsctxdemo:quantity>3<wsctxdemo:quantity>
63          <wsctxdemo:price>9.55<wsctxdemo:price>
64      </wsctxdemo:item>
65  </wsctxdemo:retailer>
66  </wsctxdemo:shoppingCart>
67  <wscfdemo:context>
```

6 Registration Service

The Registration Service is an implementation of the Registration Service as specified in the Web Services Coordination Framework Specification [WSCF]. The Service is responsible for managing the registration of participants that wish to enlist in the notification protocol., identified by the URI defined in Chapter . Participants registered for this protocol *must* implement the service endpoint as identified in Chapter .

All Registration Service communication adheres to the normative one way binding as specified in [WSCF]. Service endpoints used as part of the Registration Service communication are identified using the WS-Addressing specification [WSADDR].

7 Notification Service

The Notification Service is responsible for coordinating delivery of notifications to registered participants. Services wishing to send a notification to registered participants *must* send a one way message to the notification service endpoint specified in the wscfdemo:notification-service element of the wscfdemo:context.

The notifications service exposes its functionality through the NotificationPortType Web Service.

```
<wsdl:portType name="NotificationPortType">
  <wsdl:operation name="itemAddedToCart">
    <wsdl:input message="tns:itemAddedToCartRequest" name="itemToCartRequest"/>
  </wsdl:operation>
  <wsdl:operation name="itemRemovedFromCart">
    <wsdl:input message="tns:itemRemovedFromCartRequest"
      name="itemRemovedFromCartRequest"/>
  </wsdl:operation>
  <wsdl:operation name="orderSubmitted">
    <wsdl:input message="tns:orderSubmittedRequest"
      name="orderSubmittedRequest"/>
  </wsdl:operation>
</wsdl:portType>
```

There are three operations defined

- itemAddedToCart

This operation receives a notification that an item has been added to the shopping cart.

- itemRemovedFromCart

This operation receives a notification that an item has been removed the shopping cart.

- orderSubmitted

This operation receives notification that the contents of the shopping cart have been submitted as an order on behalf of the customer.

The notification service *must* attempt to deliver all notifications to the participants registered with the registration service. The ordering of those notifications and the handling of failed deliveries are implementation dependent. As such no assumptions should be made by the participants with regards to the sequencing, timing or successful delivery of notifications.

8 Participant Service

The Participant Service receives notifications from the Notification Service.

The participant service exposes its functionality through the ParticipantPortType Web Service.

```
<wsdl:portType name="ParticipantPortType">
  <wsdl:operation name="itemAddedToCart">
    <wsdl:input message="tns:itemAddedToCartRequest" name="itemToCartRequest"/>
  </wsdl:operation>
  <wsdl:operation name="itemRemovedFromCart">
    <wsdl:input message="tns:itemRemovedFromCartRequest"
      name="itemRemovedFromCartRequest"/>
  </wsdl:operation>
  <wsdl:operation name="orderSubmitted">
    <wsdl:input message="tns:orderSubmittedRequest"
      name="orderSubmittedRequest"/>
  </wsdl:operation>
</wsdl:portType>
```

There are three operations defined

- itemAddedToCart

This operation receives a notification that an item has been added to the shopping cart.

- itemRemovedFromCart

This operation receives a notification that an item has been removed the shopping cart.

- orderSubmitted

This operation receives notification that the contents of the shopping cart have been submitted as an order on behalf of the customer.

The functionality of the registered participant is implementation dependent.

9 Retailer Service

The Retailer Service provides is responsible for providing the Web Store with information about the products to be purchased, modifying the contents of the context shopping cart and fulfilling any submitted orders. Any modification to the shopping cart or fulfilment of a submitted order *must* result in a notification being raised via the Notification Service.

The retailer service exposes its functionality through the RetailerPortType Web Service.

```
<wsdl:portType name="RetailerPortType">
  <wsdl:operation name="getCatalog">
    <wsdl:input message="tns:getCatalogRequest" name="getCatalogRequest"/>
    <wsdl:output message="tns:getCatalogResponse" name="getCatalogResponse"/>
  </wsdl:operation>
  <wsdl:operation name="addItemToCart">
    <wsdl:input message="tns:addItemToCartRequest" name="addItemToCartRequest"/>
    <wsdl:output message="tns:addItemToCartResponse"
      name="addItemToCartResponse"/>
    <wsdl:fault message="tns:invalidProductCodeFault" name="invalidProductCode"/>
  </wsdl:operation>
  <wsdl:operation name="removeItemFromCart">
    <wsdl:input message="tns:removeItemFromCartRequest"
      name="removeItemFromCartRequest"/>
    <wsdl:output message="tns:removeItemFromCartResponse"
      name="removeItemFromCartResponse"/>
    <wsdl:fault message="tns:invalidProductCodeFault" name="invalidProductCode"/>
  </wsdl:operation>
  <wsdl:operation name="submitOrder">
    <wsdl:input message="tns:submitOrderRequest" name="submitOrderRequest"/>
    <wsdl:output message="tns:submitOrderResponse" name="submitOrderResponse"/>
    <wsdl:fault message="tns:badOrderFault" name="badOrder"/>
  </wsdl:operation>
</wsdl:portType>
```

There are four operations defined

- getCatalog

This operation *must* provide the Web Store with information regarding all products that can be purchased from the retailer.

- addItemToCart

This operation *must* add the specified quantity of the product to the shopping cart context associated with the current invocation and initiate an itemAddedToCart notification.

- removeItemFromCart

This operation *must* remove the specified quantity of the product to the shopping cart context associated with the current invocation and initiate an itemRemovedFromCart notification.

- submitOrder

This operation *must* remove all products from the shopping cart context associated with the current invocation, initiate an orderSubmitted notification and return the details of the order to the caller.

10 Web Store

The Web Store is responsible for all interactions with the customer and has to provide the following functionality

- Configuration of Context Service

The endpoint for the Context Service *must* be configurable through the GUI.

- Identification and Display of Retailer Services

The customer *must* be able to choose a Retailer Service from which to purchase goods. The details of how and when these services are chosen are implementation dependent.

- Display of Retailer Service catalog

The catalog of products offered for purchase by the Retailer Service *must* be displayed to the Customer.

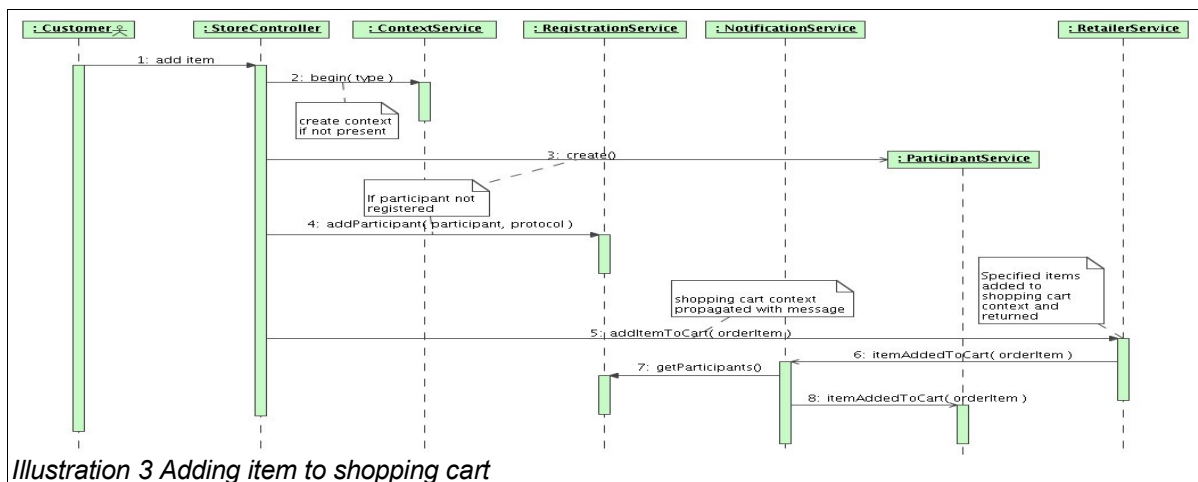
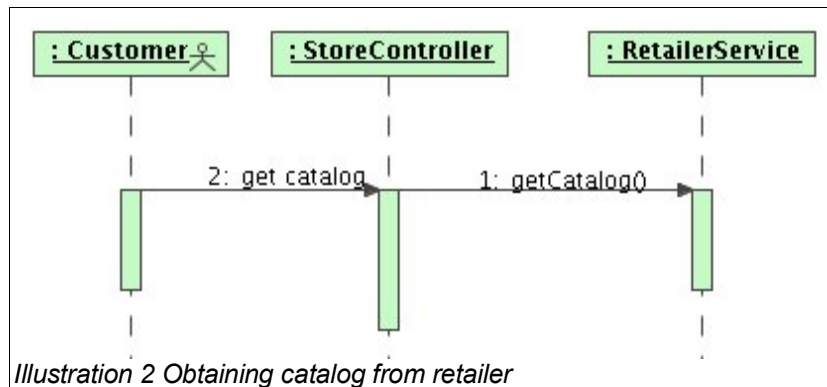
- Display and modification of the shopping cart contents.

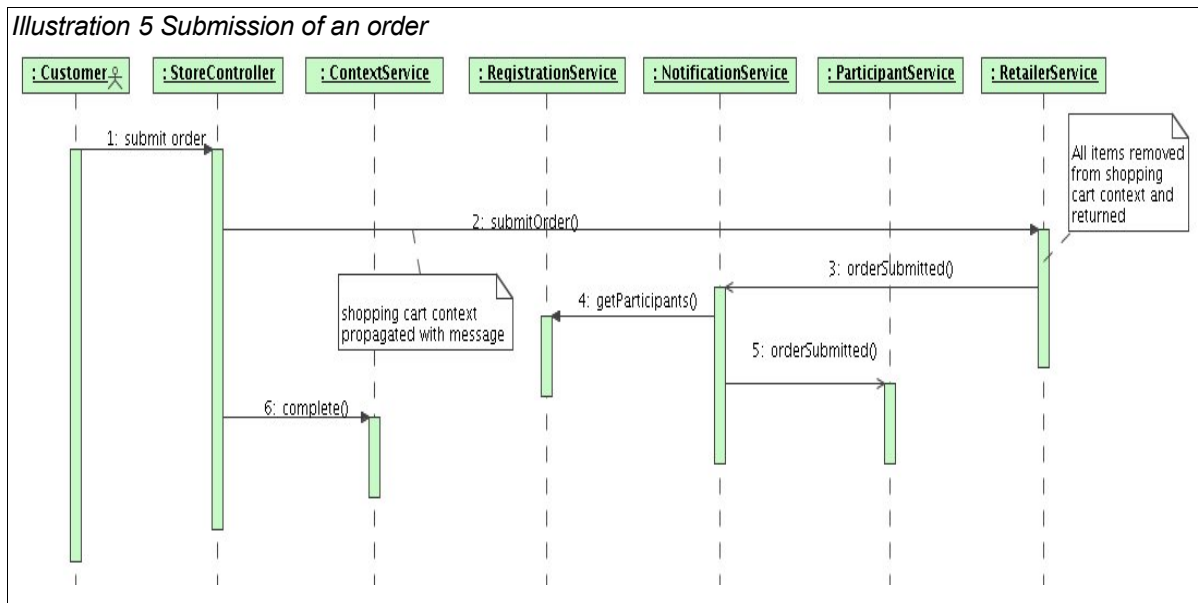
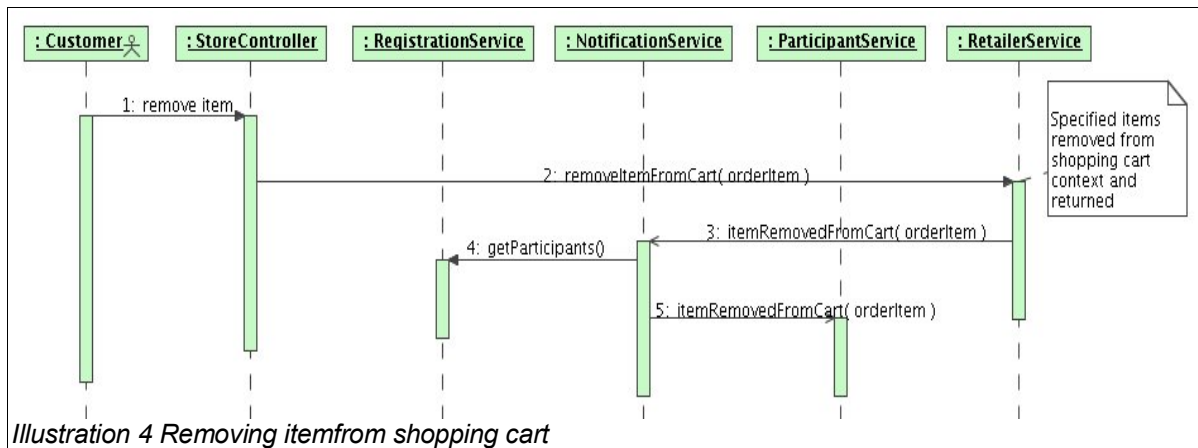
The customer *must* be able to view and modify the shopping cart.

- Checkout of shopping cart

The customer *must* be able to fulfil the purchase of the items in the shopping cart from any Retailer Service configured. The GUI *should* display a receipt of all products purchased after the order has been submitted.

The following sequence diagrams are not prescriptive, their purpose is to provide an illustration of the interactions expected during the purchase of product from a Retailer Service.





11 References

11.1 Normative References

- [RFC2119]** S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, IETF RFC 2119, March 1997, <http://www.ietf.org/rfc/rfc2119.txt>
- [WSCF]** Web Services Coordination Framework Specification, Oasis Committee Draft 0.2, August 2005, <http://www.oasis-open.org/apps/org/workgroup/ws-caf/documents.php>
- [WSCTX]** Web Services Context Specification, Oasis Committee Draft 0.9, May 2005, <http://www.oasis-open.org/apps/org/workgroup/ws-caf/documents.php>
- [WSADDR]** Web Services Addressing Specification, 10th August 2004, <http://www.w3.org/Submission/ws-addressing/>
- [WSCTXDEMO]** K. Conner, G. Tully, J. Fuller, WS-Context Demo Application, Oasis WS-CAF Implementation TC, August 2004

12 Acknowledgements

The editors would like to acknowledge the contributions of the OASIS Web Services Composite Application Framework Technical Committee, especially the original editors of the WS-Context Demo Application:

- Malik Saheb, Arjuna Technologies
- Martin Chapman, Oracle
- Simeon M. Greene, Oracle
- Jeff Mischkinsky, Oracle
- Greg Pavlik, Oracle

A. Application Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema elementFormDefault="qualified"
  targetNamespace="http://docs.oasis-open.org/wscaf/2005/08/wscfdemo"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://docs.oasis-open.org/wscaf/2005/08/wscfdemo"
  xmlns:wscf="http://docs.oasis-open.org/wscaf/2005/07/wscf"
  xmlns:wsctxdemo="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"
  xmlns:ref="http://docs.oasis-open.org/wsrn/2004/06/reference-1.1.xsd">
  <xsd:import namespace="http://docs.oasis-open.org/wscaf/2005/07/wscf"/>
  <xsd:import namespace="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"/>
  <xsd:import
    namespace="http://docs.oasis-open.org/wsrn/2004/06/reference-1.1.xsd"/>

  <!-- context type and element -->
  <xsd:complexType name="context">
    <xsd:complexContent>
      <xsd:extension base="wscf:RegistrationContextType">
        <xsd:sequence>
          <xsd:element name="notification-service" type="ref:ServiceRefType"/>
          <xsd:element ref="wsctxdemo:shoppingCart"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:element name="context" type="tns:context"/>

  <!-- The QName used for /ctx:begin/ctx:type -->
  <xsd:simpleType name="BeginContextQName">
    <xsd:restriction base="xsd:QName">
      <xsd:enumeration value="tns:context"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```


B. Application WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
  targetNamespace="http://docs.oasis-open.org/wscaf/2005/08/wscfdemo"
  xmlns:tns="http://docs.oasis-open.org/wscaf/2005/08/wscfdemo"
  xmlns:wscfdemo="http://docs.oasis-open.org/wscaf/2005/08/wscfdemo"
  xmlns:wsctxdemo="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <xs:schema targetNamespace="xx">
      <xs:import namespace="http://docs.oasis-open.org/wscaf/2005/08/wscfdemo"/>
    </xs:schema>
  </wsdl:types>

  <!-- context -->
  <wsdl:message name="context">
    <wsdl:part name="context" element="wscfdemo:context"/>
  </wsdl:message>

  <!-- get catalog -->
  <wsdl:message name="getCatalogRequest"/>
  <wsdl:message name="getCatalogResponse">
    <wsdl:part name="response" element="wsctxdemo:catalog"/>
  </wsdl:message>
  <!-- add item -->
  <wsdl:message name="addItemToCartRequest">
    <wsdl:part name="request" element="wsctxdemo:orderItem"/>
  </wsdl:message>
  <wsdl:message name="addItemToCartResponse"/>
  <!-- remove item -->
  <wsdl:message name="removeItemFromCartRequest">
    <wsdl:part name="request" element="wsctxdemo:orderItem"/>
  </wsdl:message>
  <wsdl:message name="removeItemFromCartResponse"/>
  <!-- submit order -->
  <wsdl:message name="submitOrderRequest"/>
  <wsdl:message name="submitOrderResponse">
    <wsdl:part name="response" element="wsctxdemo:shoppingCart"/>
  </wsdl:message>

  <!-- item added -->
  <wsdl:message name="itemAddedToCartRequest">
    <wsdl:part name="request" element="wsctxdemo:orderItem"/>
  </wsdl:message>
  <!-- item removed -->
  <wsdl:message name="itemRemovedFromCartRequest">
    <wsdl:part name="request" element="wsctxdemo:orderItem"/>
  </wsdl:message>
  <!-- order submitted -->
  <wsdl:message name="orderSubmittedRequest">
    <wsdl:part name="request" element="wsctxdemo:shoppingCart"/>
  </wsdl:message>

  <!-- faults -->
  <wsdl:message name="invalidProductCodeFault"/>
  <wsdl:message name="badOrderFault">
    <wsdl:part name="fault" element="wsctxdemo:badOrder"/>
  </wsdl:message>

  <!-- retailer port type -->
  <wsdl:portType name="RetailerPortType">
    <wsdl:operation name="getCatalog">
      <wsdl:input message="tns:getCatalogRequest" name="getCatalogRequest"/>
      <wsdl:output message="tns:getCatalogResponse" name="getCatalogResponse"/>
    </wsdl:operation>
    <wsdl:operation name="addItemToCart">
      <wsdl:input message="tns:addItemToCartRequest"
        name="addItemToCartRequest"/>
      <wsdl:output message="tns:addItemToCartResponse"
        name="addItemToCartResponse"/>
      <wsdl:fault message="tns:invalidProductCodeFault"

```

```

74         name="invalidProductCode"/>
75     </wsdl:operation>
76     <wsdl:operation name="removeItemFromCart">
77         <wsdl:input message="tns:removeItemFromCartRequest"
78             name="removeItemFromCartRequest"/>
79         <wsdl:output message="tns:removeItemFromCartResponse"
80             name="removeItemFromCartResponse"/>
81         <wsdl:fault message="tns:invalidProductCodeFault"
82             name="invalidProductCode"/>
83     </wsdl:operation>
84     <wsdl:operation name="submitOrder">
85         <wsdl:input message="tns:submitOrderRequest" name="submitOrderRequest"/>
86         <wsdl:output message="tns:submitOrderResponse" name="submitOrderResponse"/>
87         <wsdl:fault message="tns:badOrderFault" name="badOrder"/>
88     </wsdl:operation>
89 </wsdl:portType>
90
91 <!-- notification port type -->
92 <wsdl:portType name="NotificationPortType">
93     <wsdl:operation name="itemAddedToCart">
94         <wsdl:input message="tns:itemAddedToCartRequest" name="itemToCartRequest"/>
95     </wsdl:operation>
96     <wsdl:operation name="itemRemovedFromCart">
97         <wsdl:input message="tns:itemRemovedFromCartRequest"
98             name="itemRemovedFromCartRequest"/>
99     </wsdl:operation>
100    <wsdl:operation name="orderSubmitted">
101        <wsdl:input message="tns:orderSubmittedRequest"
102            name="orderSubmittedRequest"/>
103    </wsdl:operation>
104 </wsdl:portType>
105
106 <!-- participant port type -->
107 <wsdl:portType name="ParticipantPortType">
108     <wsdl:operation name="itemAddedToCart">
109         <wsdl:input message="tns:itemAddedToCartRequest" name="itemToCartRequest"/>
110     </wsdl:operation>
111     <wsdl:operation name="itemRemovedFromCart">
112         <wsdl:input message="tns:itemRemovedFromCartRequest"
113             name="itemRemovedFromCartRequest"/>
114     </wsdl:operation>
115     <wsdl:operation name="orderSubmitted">
116         <wsdl:input message="tns:orderSubmittedRequest"
117             name="orderSubmittedRequest"/>
118     </wsdl:operation>
119 </wsdl:portType>
120
121 <!-- retailer soap binding -->
122 <wsdl:binding name="RetailerSoapBinding" type="tns:RetailerPortType">
123     <soap:binding style="document"
124         transport="http://schemas.xmlsoap.org/soap/http"/>
125     <wsdl:operation name="getCatalog">
126         <wsdl:input>
127             <soap:body use="literal"/>
128         </wsdl:input>
129         <wsdl:output>
130             <soap:body use="literal"/>
131         </wsdl:output>
132     </wsdl:operation>
133     <wsdl:operation name="addItemToCart">
134         <wsdl:input>
135             <soap:body use="literal"/>
136             <soap:header use="literal" message="tns:context" part="context"/>
137         </wsdl:input>
138         <wsdl:output>
139             <soap:body use="literal"/>
140             <soap:header use="literal" message="tns:context" part="context"/>
141         </wsdl:output>
142         <wsdl:fault name="invalidProductCode">
143             <soap:fault name="invalidProductCode" use="literal"/>
144         </wsdl:fault>
145     </wsdl:operation>
146     <wsdl:operation name="removeItemFromCart">
147         <wsdl:input>
148             <soap:body use="literal"/>
149             <soap:header use="literal" message="tns:context" part="context"/>

```

```

150     </wsdl:input>
151     <wsdl:output>
152         <soap:body use="literal"/>
153         <soap:header use="literal" message="tns:context" part="context"/>
154     </wsdl:output>
155     <wsdl:fault name="invalidProductCode">
156         <soap:fault name="invalidProductCode" use="literal"/>
157     </wsdl:fault>
158 </wsdl:operation>
159 <wsdl:operation name="submitOrder">
160     <wsdl:input>
161         <soap:body use="literal"/>
162         <soap:header use="literal" message="tns:context" part="context"/>
163     </wsdl:input>
164     <wsdl:output>
165         <soap:body use="literal"/>
166         <soap:header use="literal" message="tns:context" part="context"/>
167     </wsdl:output>
168     <wsdl:fault name="badOrder">
169         <soap:fault name="badOrder" use="literal"/>
170     </wsdl:fault>
171 </wsdl:operation>
172 </wsdl:binding>
173
174 <!-- notification soap binding -->
175 <wsdl:binding name="NotificationSoapBinding" type="tns:NotificationPortType">
176     <soap:binding style="document"
177         transport="http://schemas.xmlsoap.org/soap/http"/>
178     <wsdl:operation name="itemAddedToCart">
179         <wsdl:input>
180             <soap:body use="literal"/>
181             <soap:header use="literal" message="tns:context" part="context"/>
182         </wsdl:input>
183     </wsdl:operation>
184     <wsdl:operation name="itemRemovedFromCart">
185         <wsdl:input>
186             <soap:body use="literal"/>
187             <soap:header use="literal" message="tns:context" part="context"/>
188         </wsdl:input>
189     </wsdl:operation>
190     <wsdl:operation name="orderSubmitted">
191         <wsdl:input>
192             <soap:body use="literal"/>
193             <soap:header use="literal" message="tns:context" part="context"/>
194         </wsdl:input>
195     </wsdl:operation>
196 </wsdl:binding>
197
198 <!-- participant soap binding -->
199 <wsdl:binding name="ParticipantSoapBinding" type="tns:ParticipantPortType">
200     <soap:binding style="document"
201         transport="http://schemas.xmlsoap.org/soap/http"/>
202     <wsdl:operation name="itemAddedToCart">
203         <wsdl:input>
204             <soap:body use="literal"/>
205             <soap:header use="literal" message="tns:context" part="context"/>
206         </wsdl:input>
207     </wsdl:operation>
208     <wsdl:operation name="itemRemovedFromCart">
209         <wsdl:input>
210             <soap:body use="literal"/>
211             <soap:header use="literal" message="tns:context" part="context"/>
212         </wsdl:input>
213     </wsdl:operation>
214     <wsdl:operation name="orderSubmitted">
215         <wsdl:input>
216             <soap:body use="literal"/>
217             <soap:header use="literal" message="tns:context" part="context"/>
218         </wsdl:input>
219     </wsdl:operation>
220 </wsdl:binding>
221 </wsdl:definitions>

```

C. Revision History

Rev	Date	By Whom	What
01	3 Aug 2005	Kevin Conner	The initial draft of the demo application specification.

D. Notices

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification, can be obtained from the OASIS Executive Director.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS Executive Director.

Copyright © OASIS Open 2004. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself does not be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS Intellectual Property Rights document must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.