

---

# WS-Context Demo Application

## Working Draft 03, 3 August 2005

### Document identifier:

ws\_caf\_implement-ws\_context-demo-draft-03

### Location:

[http://www.oasis-open.org/committees/documents.php?wg\\_abbrev=ws-caf-implement](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 creation, manipulation and propagation of a WS-CAF Context [WSCTX], inspired by the SCM Application implemented by the WS-I Sample Application working group. It provides the basis upon which the entire suite of WS-CAF specifications can be demonstrated.

### 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](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 Retailer Service.....	7
10	7 Web Store.....	8
11	8 References.....	10
12	8.1 Normative References.....	10
13	8.2 Non Normative References.....	10
14	9 Acknowledgements.....	11

---

## 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/2005/06/wsctxdemo>

#### 2.1.1 Prefix Namespace

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

Prefix	Namespace
wsctxdemo	<a href="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo">http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo</a>
wsctx	<a href="http://docs.oasis-open.org/wscaf/2005/06/wsctx">http://docs.oasis-open.org/wscaf/2005/06/wsctx</a>
xsd	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>
wsdl	<a href="http://schemas.xmlsoap.org/wsdl">http://schemas.xmlsoap.org/wsdl</a>
soap	<a href="http://schemas.xmlsoap.org/wsdl/soap">http://schemas.xmlsoap.org/wsdl/soap</a>
wsa	<a href="http://schemas.xmlsoap.org/ws/2004/08/addressing">http://schemas.xmlsoap.org/ws/2004/08/addressing</a>
tns	<a href="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo">http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo</a>

---

## 3 Overview

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

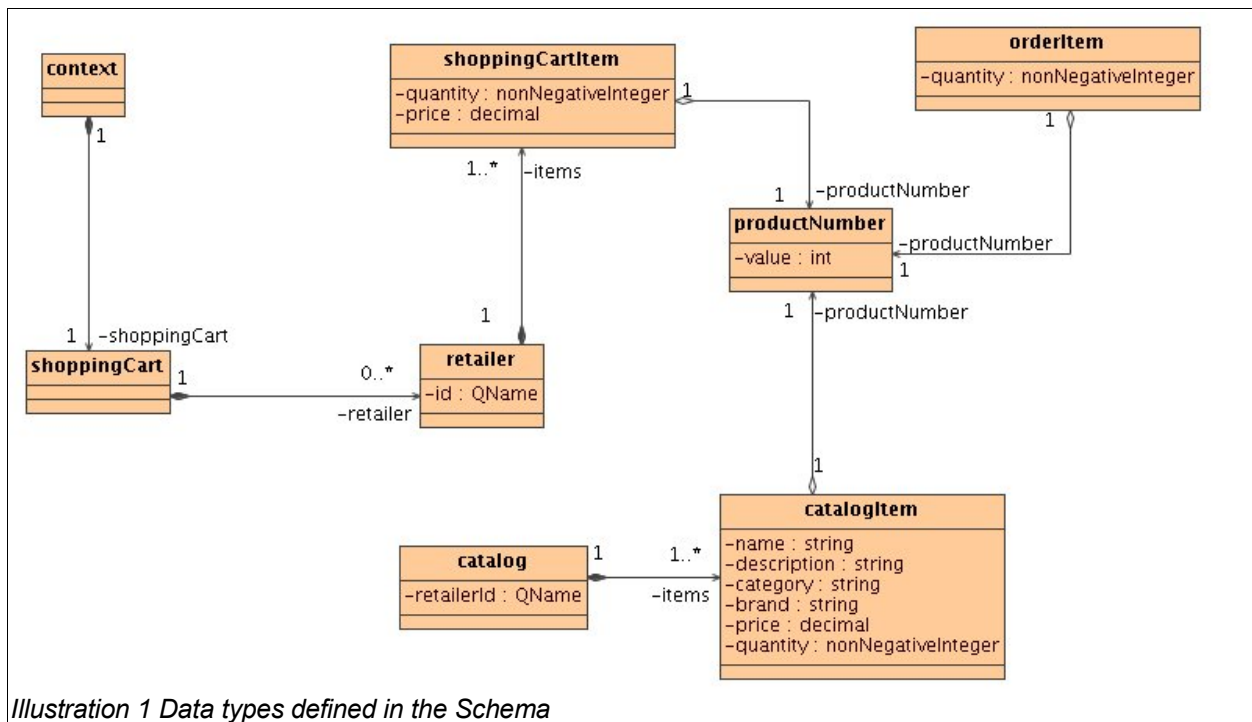
The specification of the application is divided into four sections

- The XML Schema document describing the elements used within the application context and the application WSDL messages.
- The Context Service for creating and destroying the context.
- 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

The schema document defines the XML structure of the application and context messages.



The schema document is divided into six sections. The first four sections define the types shown in Illustration 1, with the last two sections defining faults used in application messaging and the qualified name of the application context. The contents of the 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 `wsctxdemo: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 should contain an empty shopping cart.

```
<wsctxdemo: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>
  <wsctxdemo:shoppingCart/>
</wsctxdemo:context>
```

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

```
<wsctxdemo: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>
  <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>
      <wsctxdemo:item>
        <wsctxdemo:productNumber>12345<wsctxdemo:productNumber>
        <wsctxdemo:quantity>3<wsctxdemo:quantity>
        <wsctxdemo:price>9.55<wsctxdemo:price>
      </wsctxdemo:item>
    </wsctxdemo:retailer>
  </wsctxdemo:shoppingCart>
</wsctxdemo:context>
```

## 6 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.

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.

- **removeItemFromCart**

This operation *must* remove the specified quantity of the product to the shopping cart context associated with the current invocation.

- **submitOrder**

This operation *must* remove all products from the shopping cart context associated with the current invocation and return the details to the caller.

## 7 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.

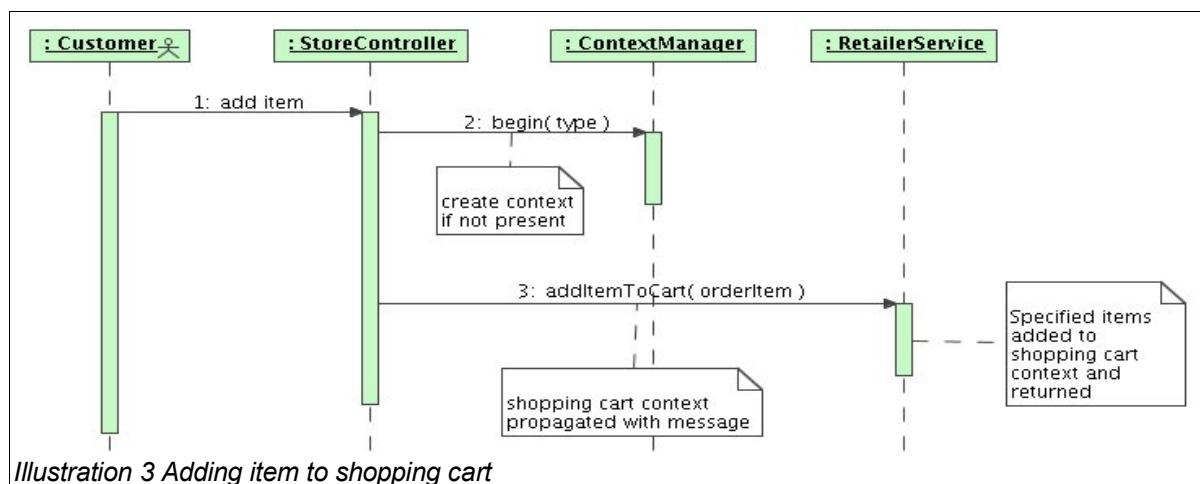
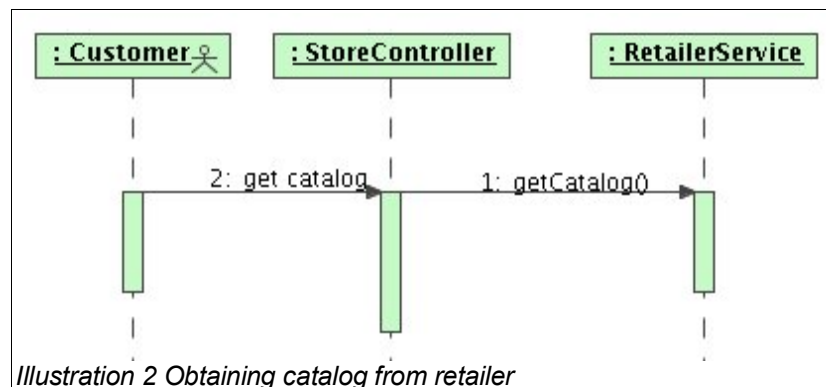
- Display and modification of the shopping cart contents.

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

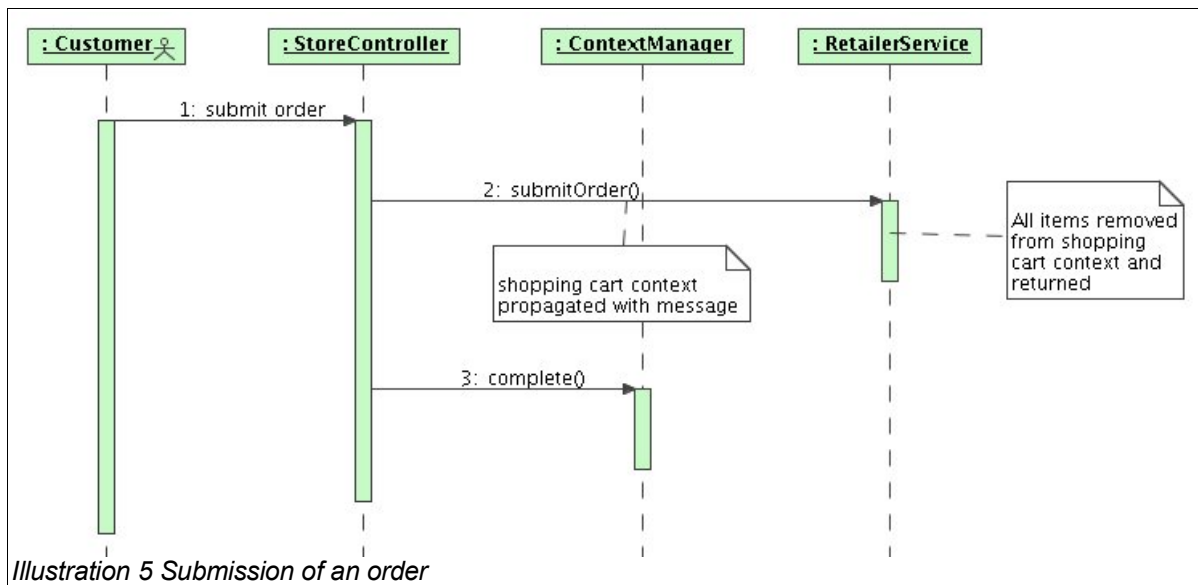
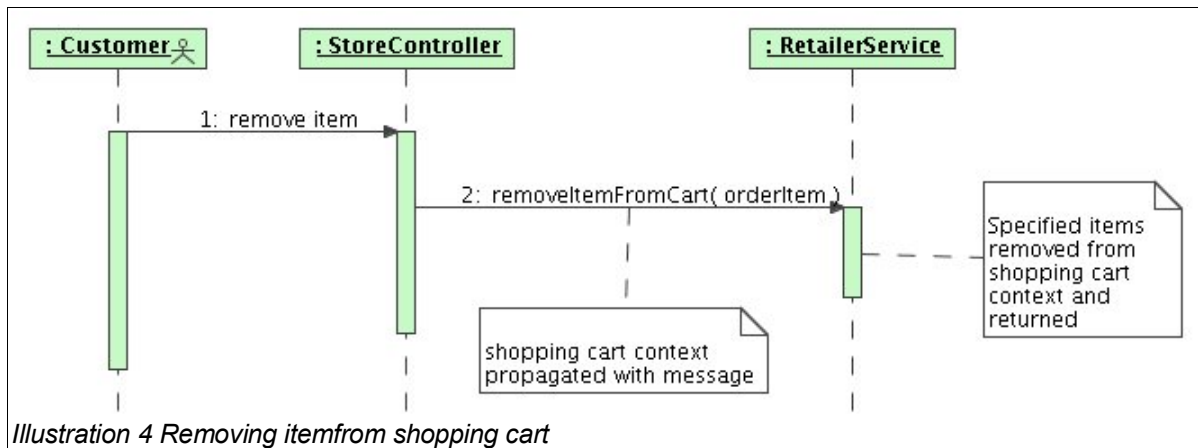
- Checkout of shopping cart

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.







---

## 8 References

### 8.1 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>
- [WSCTX] Web Services Context Specification, Oasis Committee Draft 0.9.2, July 2005, <http://www.oasis-open.org/apps/org/workgroup/ws-caf/documents.php>
- [WSADDR] Web Services Addressing Specification, 10<sup>th</sup> August 2004, <http://www.w3.org/Submission/ws-addressing/>

### 8.2 Non Normative References

- [WSCTXDEMO] S.M. Greene, WS-Context Demo Application, Oasis WS-CAF Implementation TC, February 2004, <http://www.oasis-open.org/apps/org/workgroup/ws-caf-implement/documents.php>

---

## 9 Acknowledgements

The editors would like to acknowledge the contributions of the OASIS Web Services Composite Application Framework Technical Committee, especially the 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/2005/06/wsctxdemo"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"
  xmlns:wsctx="http://docs.oasis-open.org/wscaf/2005/06/wsctx">
  <xsd:import namespace="http://docs.oasis-open.org/wscaf/2005/06/wsctx"/>

  <!-- Catalog types and elements -->
  <xsd:simpleType name="productNumber">
    <xsd:restriction base="xsd:int">
      <xsd:minInclusive value="1"/>
      <xsd:maxInclusive value="999999"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="catalogItem">
    <xsd:sequence>
      <xsd:element name="name" type="xsd:string"/>
      <xsd:element name="description" type="xsd:string"/>
      <xsd:element name="productNumber" type="tns:productNumber"/>
      <xsd:element name="category" type="xsd:string"/>
      <xsd:element name="brand" type="xsd:string"/>
      <xsd:element name="price" type="xsd:decimal"/>
      <xsd:element name="quantity" type="xsd:nonNegativeInteger"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="catalog">
    <xsd:sequence>
      <xsd:element name="retailerId" type="xsd:QName"/>
      <xsd:element name="item" type="tns:catalogItem" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

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

  <!-- Order types and elements -->
  <xsd:complexType name="orderItem">
    <xsd:sequence>
      <xsd:element name="productNumber" type="tns:productNumber"/>
      <xsd:element name="quantity" type="xsd:nonNegativeInteger"/>
    </xsd:sequence>
  </xsd:complexType>

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

  <!-- shopping cart type and elements -->
  <xsd:complexType name="shoppingCartItem">
    <xsd:sequence>
      <xsd:element name="productNumber" type="tns:productNumber"/>
      <xsd:element name="quantity" type="xsd:nonNegativeInteger"/>
      <xsd:element name="price" type="xsd:decimal"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="retailer">
    <xsd:sequence>
      <xsd:element name="id" type="xsd:QName"/>
      <xsd:element name="item" type="tns:shoppingCartItem"
        maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="shoppingCart">
    <xsd:sequence minOccurs="0">
      <xsd:element name="retailer" type="tns:retailer" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:element name="shoppingCart" type="tns:shoppingCart"/>
```

```

74 <!-- context type and element -->
75 <xsd:complexType name="context">
76   <xsd:complexContent>
77     <xsd:extension base="wsctx:ContextType">
78       <xsd:sequence>
79         <xsd:element ref="tns:shoppingCart"/>
80       </xsd:sequence>
81     </xsd:extension>
82   </xsd:complexContent>
83 </xsd:complexType>
84
85 <xsd:element name="context" type="tns:context"/>
86
87 <!-- faults -->
88 <xsd:complexType name="badOrderItem">
89   <xsd:sequence>
90     <xsd:element name="productNumber" type="tns:productNumber"/>
91     <xsd:element name="reason" type="xsd:string"/>
92   </xsd:sequence>
93 </xsd:complexType>
94
95 <xsd:complexType name="badOrder">
96   <xsd:sequence>
97     <xsd:element name="item" type="tns:badOrderItem" maxOccurs="unbounded"/>
98   </xsd:sequence>
99 </xsd:complexType>
100
101 <xsd:element name="badOrder" type="tns:badOrder"/>
102
103 <!-- The QName used for /ctx:begin/ctx:type -->
104 <xsd:simpleType name="BeginContextQName">
105   <xsd:restriction base="xsd:QName">
106     <xsd:enumeration value="tns:context"/>
107   </xsd:restriction>
108 </xsd:simpleType>
109 </xsd:schema>

```

## B. Application WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
  targetNamespace="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"
  xmlns:tns="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"
  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>
      <xs:import namespace="http://docs.oasis-open.org/wscaf/2005/06/wsctxdemo"/>
    </xs:schema>
  </wsdl:types>

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

  <!-- 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"
        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>

```

```

74 </wsdl:portType>
75
76 <!-- retailer soap binding -->
77 <wsdl:binding name="RetailerSoapBinding" type="tns:RetailerPortType">
78   <soap:binding style="document"
79     transport="http://schemas.xmlsoap.org/soap/http"/>
80   <wsdl:operation name="getCatalog">
81     <wsdl:input>
82       <soap:body use="literal"/>
83     </wsdl:input>
84     <wsdl:output>
85       <soap:body use="literal"/>
86     </wsdl:output>
87   </wsdl:operation>
88   <wsdl:operation name="addItemToCart">
89     <wsdl:input>
90       <soap:body use="literal"/>
91       <soap:header use="literal" message="tns:context" part="context"/>
92     </wsdl:input>
93     <wsdl:output>
94       <soap:body use="literal"/>
95       <soap:header use="literal" message="tns:context" part="context"/>
96     </wsdl:output>
97     <wsdl:fault name="invalidProductCode">
98       <soap:fault name="invalidProductCode" use="literal"/>
99     </wsdl:fault>
100   </wsdl:operation>
101   <wsdl:operation name="removeItemFromCart">
102     <wsdl:input>
103       <soap:body use="literal"/>
104       <soap:header use="literal" message="tns:context" part="context"/>
105     </wsdl:input>
106     <wsdl:output>
107       <soap:body use="literal"/>
108       <soap:header use="literal" message="tns:context" part="context"/>
109     </wsdl:output>
110     <wsdl:fault name="invalidProductCode">
111       <soap:fault name="invalidProductCode" use="literal"/>
112     </wsdl:fault>
113   </wsdl:operation>
114   <wsdl:operation name="submitOrder">
115     <wsdl:input>
116       <soap:body use="literal"/>
117       <soap:header use="literal" message="tns:context" part="context"/>
118     </wsdl:input>
119     <wsdl:output>
120       <soap:body use="literal"/>
121       <soap:header use="literal" message="tns:context" part="context"/>
122     </wsdl:output>
123     <wsdl:fault name="badOrder">
124       <soap:fault name="badOrder" use="literal"/>
125     </wsdl:fault>
126   </wsdl:operation>
127 </wsdl:binding>
128 </wsdl:definitions>

```

---

## C. Revision History

Rev	Date	By Whom	What
01	9 Jun 2005	Kevin Conner	The initial draft of the demo application specification.
02	16 Jun 2005	Kevin Conner	Unified namespaces and updated wsctx files.
03	3 Aug 2005	Kevin Conner	Updated context reference and WSDL



---

## 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.