Abstract:
This document specifies SOAP protocol bindings for oBIX. oBIX provides the core information model and interaction pattern for communication with building control systems. Specific implementations of oBIX must choose how to bind oBIX interactions. This document describes the SOAP Binding and includes a WSDL artifact.
Status:
This document was last revised or approved by the OASIS Open Building Information Exchange (oBIX) TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee’s email list. Others should send comments to the Technical Committee by using the “Send A Comment” button on the Technical Committee’s web page at http://www.oasis-open.org/committees/obix/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/obix/ipr.php).

Citation format:
When referencing this specification the following citation format should be used:

[oBIX-SOAP]

Notices

Copyright © OASIS Open 2013. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

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 section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, 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 OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

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' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on 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 implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see http://www.oasis-open.org/policies-guidelines/trademark for above guidance.
# Table of Contents

1 Introduction........................................................................................................................................5  
  1.1 Terminology....................................................................................................................................5  
  1.2 Normative References ...................................................................................................................5  
  1.3 Non-Normative References ...........................................................................................................5  
2 SOAP Binding ......................................................................................................................................6  
  2.1 SOAP Example...............................................................................................................................6  
  2.2 Error Handling...............................................................................................................................6  
  2.3 Security .........................................................................................................................................6  
  2.4 Localization ...................................................................................................................................6  
  2.5 WSDL ............................................................................................................................................7  
3 Conformance .......................................................................................................................................9  
Appendix A. Acknowledgments ..............................................................................................................10  
Appendix B. Revision History ...............................................................................................................11
1 Introduction

This document specifies the SOAP protocol bindings for oBIX.

1.1 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 RFC2119.

1.2 Normative References


oBIX oBIX Version 1.1.

See link in "Related work" section on cover page.


See link in "Related work" section on cover page.


1.3 Non-Normative References

2 SOAP Binding

The SOAP binding maps a SOAP operation to each of the three oBIX request types: read, write and invoke. Like the HTTP binding, read is supported by every object, write is supported by objects whose writable attribute is true, and invoke is only supported by operations. Inputs and outputs of each request are specific to the target object.

Unlike the HTTP binding, requests are not accessed via the URI of the target object, but instead via the URI of the SOAP server with the object’s URI encoded into the body of the SOAP envelope.

2.1 SOAP Example

The following is a SOAP request to an oBIX server’s About object:

```xml
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <obixWS:read xmlns:obixWS="http://obix.org/ns/wdsl/1.1"
                 xmlns="http://obix.org/ns/schema/1.1"
                 href="http://localhost/obix/about" />
  </env:Body>
</env:Envelope>
```

An example response to the above request:

```xml
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <obixWS:response xmlns:obixWS="http://obix.org/ns/wdsl1.1"
                     xmlns="http://obix.org/ns/schema/1.1">
      <obj name="about" href="http://localhost/obix/about/">
        <str name="obixVersion" val="1.1"/>
        <str name="serverName" val="obix"/>
        <abstime name="serverTime" val="2006-02-08T09:40:55.000+05:00:00Z"/>
        <abstime name="serverBootTime" val="2006-02-08T09:33:31.980+05:00:00Z"/>
        <str name="vendorName" val="Acmce, Inc."/>
        <uri name="vendorUrl" val="http://www.acme.com"/>
        <str name="productName" val="Acme oBIX Server"/>
        <str name="productVersion" val="1.0.3"/>
        <uri name="productUrl" val="http://www.acme.com/obix"/>
      </obj>
    </obixWS:response>
  </env:Body>
</env:Envelope>
```

2.2 Error Handling

The oBIX specification defines no SOAP faults. If a request is processed by an oBIX server, then a valid oBIX document SHOULD be returned with a failure indicated via the err object.

2.3 Security

Refer to the recommendations in WS-I Basic Profile 2.0 for security [WS-I].

2.4 Localization

SOAP bindings SHOULD follow localization patterns defined for the HTTP binding when applicable (see Section [oBIX REST]).
2.5 WSDL

In the types section of the WSDL document, the oBIX schema is imported. Server implementations might consider providing the schemaLocation attribute which is absent in the standard document.

Missing from the standard oBIX WSDL is the service element. This element binds a SOAP server instance with a network address. Each instance will have to provide its own services section of the WSDL document. The following is an example of the WSDL service element:

```xml
<wsdl:service name="obix">
  <wsdl:port name="obixPort" binding="tns:obixSoapBinding">
    <soap:address location="http://localhost/obix/soap"/>
  </wsdl:port>
</wsdl:service>
```

Standard oBIX WSDL is:

```xml
<wSDL:definitions targetNamespace="http://obix.org/ns/wsdl/1.1"
  xmlns="http://obix.org/ns/wsdl/1.1"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <xsd:schema xmlns:obix="http://obix.org/ns/schema/1.1"
    targetNamespace="http://obix.org/ns/wsdl/1.1"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:import namespace="http://obix.org/ns/schema/1.1"
      schemaLocation="obix.xsd" />

    <xsd:complexType name="ReadReq">
      <xsd:attribute name="href" type="xsd:anyURI"/>
    </xsd:complexType>

    <xsd:complexType name="WriteReq">
      <xsd:complexContent>
        <xsd:extension base="ReadReq">
          <xsd:group ref="obix:objGroup"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>

    <xsd:complexType name="InvokeReq">
      <xsd:complexContent>
        <xsd:extension base="ReadReq">
          <xsd:group ref="obix:objGroup"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>

    <xsd:complexType name="ResponseType">
      <xsd:group ref="obix:objGroup" minOccurs="1"
        maxOccurs="1"/>
    </xsd:complexType>

    <xsd:element name="read" type="obixWS:ReadReq"/>
    <xsd:element name="response" type="obixWS:ResponseType"/>
    <xsd:element name="write" type="obixWS:WriteReq"/>
    <xsd:element name="invoke" type="obixWS:InvokeReq"/>
  </xsd:schema>
</wsdl:types>
```
<wsdl:message name="writeSoapReq">
  <wsdl:part name="body" element="write" />
</wsdl:message>

<wsdl:message name="writeSoapRes">
  <wsdl:part name="body" element="response" />
</wsdl:message>

<wsdl:message name="invokeSoapReq">
  <wsdl:part name="body" element="invoke" />
</wsdl:message>

<wsdl:message name="invokeSoapRes">
  <wsdl:part name="body" element="response" />
</wsdl:message>

<wsdl:portType name="oBIXSoapPort">
  <wsdl:operation name="read">
    <wsdl:input message="readSoapReq" />
    <wsdl:output message="readSoapRes" />
  </wsdl:operation>

  <wsdl:operation name="write">
    <wsdl:input message="writeSoapReq" />
    <wsdl:output message="writeSoapRes" />
  </wsdl:operation>

  <wsdl:operation name="invoke">
    <wsdl:input message="invokeSoapReq" />
    <wsdl:output message="invokeSoapRes" />
  </wsdl:operation>
</wsdl:portType>

<wsdl:binding name="oBIXSoapBinding" type="oBIXSoapPort">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="read">
    <soap:operation soapAction="http://obix.org/ns/wsdl/1.1/read"
      style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>

  <wsdl:operation name="write">
    <soap:operation soapAction="http://obix.org/ns/wsdl/1.1/write"
      style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>

  <wsdl:operation name="invoke">
    <soap:operation soapAction="http://obix.org/ns/wsdl/1.1/invoke"
      style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
</wsdl:definitions>
3 Conformance

An implementation is compliant with this specification if it implements all MUST or REQUIRED level requirements.
Appendix A. Acknowledgments

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:
- Ron Ambrosio, IBM
- Brad Benson, Trane
- Ron Bernstein, LonMark International*
- Ludo Bertsch, Continental Automated Buildings Association (CABA)
- Chris Bogen, US Department of Defense
- Rich Blomseth, Echelon Corporation
- Anto Budiardjo, Clasma Events, Inc.
- Jochen Burkhardt, IBM
- JungIn Choi, Kyungwon University
- David Clute, Cisco Systems, Inc.*
- Toby Considine, University of North Carolina at Chapel Hill
- William Cox, Individual
- Robert Dolin, Echelon Corporation
- Marek Dziedzic, Treasury Board of Canada, Secretariat
- Brian Frank, SkyFoundry
- Craig Gemmill, Tridium, Inc.
- Matthew Giannini, Tridium, Inc.
- Harald Hofstätter, Institute of Computer Aided Automation
- Markus Jung, Institute of Computer Aided Automation
- Christopher Kelly, Cisco Systems
- Wonsuk Ko, Kyungwon University
- Perry Krol, TIBCO Software Inc.
- Corey Leong, Individual
- Ulf Magnusson, Schneider Electric
- Brian Meyers, Trane
- Jeremy Roberts, LonMark International
- Thorsten Roggendorf, Echelon Corporation
- Anno Scholten, Individual
- John Sublett, Tridium, Inc.
- Dave Uden, Trane
- Ron Zimmer, Continental Automated Buildings Association (CABA)*
- Robert Zach, Institute of Computer Aided Automation
- Rob Zivney, Hirsch Electronics Corporation
- Markus Jung, Vienna University of Technology
## Appendix B. Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>wd01</td>
<td>18 Mar 2013</td>
<td>Markus Jung</td>
<td>Initial creation, WSDL modifications, SOAP</td>
</tr>
<tr>
<td>wd02</td>
<td>26 Mar 2013</td>
<td>Markus Jung</td>
<td>Reverted changes to the state of oBIX 1.1 WD07.</td>
</tr>
<tr>
<td>wd03</td>
<td>13 Jun 2013</td>
<td>Markus Jung</td>
<td>Formatting changes</td>
</tr>
<tr>
<td>wd04</td>
<td>28 Jun 2013</td>
<td>Markus Jung</td>
<td>Modified WSDL to work with the oBIX 1.1 XML schema. Introduced wrapping response type and element.</td>
</tr>
<tr>
<td>WD05</td>
<td>8 Jul 2013</td>
<td>Toby Considine</td>
<td>Minor formatting and before PR</td>
</tr>
</tbody>
</table>