



Web Services Base Faults 1.2 (WS-BaseFaults)

Working Draft 02, June 24, 2004

Document identifier:

wsrf-WS-BaseFaults-1.2-draft-02

Location:

<http://docs.oasis-open.org/wsrf/2004/06/wsrf-BaseFaults-1.2-draft-02.pdf>

Editors:

Steve Tuecke, Argonne National Laboratory < tuecke@mcs.anl.gov >

Lily Liu, webMethods, Inc < lily.liu@webmethods.com >

Sam Meder, Argonne National Laboratory < meder@mcs.anl.gov >

Abstract:

Problem determination in a Web services setting is simplified by standardizing a base set of information that may appear in fault messages. WS-BaseFaults defines an XML Schema type for base faults, along with rules for how this base fault type is used and extended by Web services.

Status:

This document and associated schema are published by this TC as "working drafts" and represent the starting point for our standardization process. It is possible that they may change significantly during this process, but should nonetheless provide a stable reference for discussion and early adopters' implementations.

Committee members should send comments on this specification to the wsrf@lists.oasis-open.org list. Others should subscribe to and send comments to the wsrf-comment@lists.oasis-open.org list. To subscribe, send an email message to wsrf-comment-request@lists.oasis-open.org with the word "subscribe" as the body of the message.

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 WSRF TC web page (<http://www.oasis-open.org/committees/wsrf/>).

Table of Contents

36 1 Introduction 2

37 1.1 Goals and Requirements 2

38 1.1.1 Requirements..... 2

39 1.1.2 Non-Goals 2

40 1.2 Notational Conventions 3

41 1.3 Namespaces 3

42 2 Base Fault Type..... 3

43 3 Use of Base Faults in WSDL 1.1 4

44 4 Security Considerations 6

45 5 References..... 6

46 Appendix A. Acknowledgments 7

47 Appendix B. Revision History 8

48 Appendix C. Notices 9

49 Appendix D. XML Schema 10

50 Appendix E. WSDL 1.1 12

1 Introduction

52 A designer of a Web services application often uses interfaces defined by others. Managing faults
53 in such an application is more difficult when each interface uses a different convention for
54 representing common information in fault messages

55 Support for problem determination and fault management can be enhanced by specifying Web
56 services fault messages in a common way. When the information available in faults from various
57 interfaces is consistent, it is easier for requestors to understand faults. It is also more likely that
58 common tooling can be created to assist in the handling of faults.

59 WS-BaseFaults defines an XML Schema type for a base fault, along with rules for how this fault
60 type is used by Web services.

61 WS-BaseFaults is inspired by a portion of the Global Grid Forum’s “Open Grid Services
62 Infrastructure (OGSI) Version 1.0” specification [OGSI].

1.1 Goals and Requirements

64 The goal of WS-BaseFaults is to standardize the terminology, concepts, XML types, and WSDL
65 usage of a base fault type for Web service interfaces.

1.1.1 Requirements

67 This specification intends to meet the following requirements:

- 68 Define a standard XML Schema type containing base fault information.
- 69 Define how this base fault type is used within WSDL defined interfaces.

1.1.2 Non-Goals

71 The following topics are outside the scope of this specification:

72 It is not an objective of this specification to define a common hierarchy of common faults upon the
73 base fault.

74 **1.2 Notational Conventions**

75 The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD",
 76 "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be
 77 interpreted as described in [RFC 2119](#).

78 When describing abstract data models, this specification uses the notational convention used by
 79 the [XML Infoset](#). Specifically, abstract property names always appear in square brackets (e.g.,
 80 [some property]).

81 When describing concrete XML schemas, this specification uses the notational convention of
 82 [Web Services Security](#). Specifically, each member of an element's [children] or [attributes]
 83 property is described using an XPath-like notation (e.g., /x:MyHeader/x:SomeProperty/@value1).
 84 The use of {any} indicates the presence of an element wildcard (<xsd:any/>). The use of @{any}
 85 indicates the presence of an attribute wildcard (<xsd:anyAttribute/>).

86 **1.3 Namespaces**

87 The following namespaces are used in this document:

Prefix	Namespace
s12	http://www.w3.org/2003/05/soap-envelope
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance
wsbf	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-draft-02.xsd
wsa	http://schemas.xmlsoap.org/ws/2003/02/addressing

88 **2 Base Fault Type**

89 The basic fault has the following syntax. The normative XML Schema definition is in Appendix I:

```

90 <BaseFault>
91   <Timestamp>xsd:dateTime</Timestamp>
92   <OriginatorReference>
93     wsa:EndpointReferenceType
94   </OriginatorReference> ?
95   <ErrorCode dialect="anyURI">xsd:string</ErrorCode> ?
96   <Description>xsd:string</Description> *
97   <FaultCause>wsbf:BaseFault</FaultCause> *
98 </BaseFault>
    
```

99 /wsbf:BaseFault/Timestamp

100 This REQUIRED element MUST be the time at which the fault occurred. There MUST be
 101 only one timestamp element in BaseFault. In the absence of the time zone designation,
 102 the xsd:dateTime value MUST be interpreted as universal time (UTC) time.

103 /wsbf:BaseFault/OriginatorReference

104 This OPTIONAL element is a WS-Addressing [\[WS-Addressing\]](#) EndpointReference of the
 105 Web service that generated the fault. This element MAY be omitted if the fault originator

106 is clearly implied by the context in which the fault appears (for example in a simple
107 request response message exchange). One use of this element is in a situation of nested
108 faults. The outer-most fault may use this component to reference the actual original
109 source of the fault condition.

110 /wsbf:BaseFault/ErrorCode

111 This OPTIONAL element provides convenient support for legacy fault reporting systems
112 (e.g., POSIX errno). The dialect attribute on ErrorCode MUST be a URI that defines the
113 context in which the ErrorCode MUST be interpreted. For example, a URI might be
114 defined that describes how a POSIX errno is mapped to a ErrorCode and that URI must
115 appear on any ErrorCode element carrying a POSIX errno.

116 /wsbf:BaseFault/Description

117 This OPTIONAL element contains a plain language description of the fault. This
118 description is expected to be helpful in explaining the fault to users. There MAY be any
119 number of description elements.

120 /wsbf:BaseFault/FaultCause

121 This OPTIONAL element is a BaseFault that describes an underlying cause of this fault.
122 There MAY be any number of FaultCause elements. This element SHOULD be used with
123 xsi:type to describe a more specialized fault that extends BaseFault. The ability to include
124 FaultCause elements in a fault allows for *chaining* of fault information so that a recipient
125 of a fault MAY examine details underlying the cause of the fault.

126 Note that there is no required child element within BaseFault that identifies the particular type (or
127 class) of fault. Rather, an application-specific extension of BaseFault MUST be defined for each
128 distinct type of fault

129 BaseFault does NOT include open element extensibility. To define an extended fault, you MUST
130 use XML Schema extension to extend the BaseFault type to include additional attributes and/or
131 elements.

132 3 Use of Base Faults in WSDL 1.1

133 Each distinct type of fault associated with a WSDL operation MUST be listed as a separate fault
134 response in the WSDL operation definition, as follows. For each distinct fault associated with a
135 Web service operation:

- 136 1. As described above, there MUST be a distinct XML Schema complexType that extends
137 wsbf:BasicFaultType, which represents this fault's distinct type. This extended fault
138 complexType MAY contain additional attributes and/or elements.
- 139 2. An element MUST be defined for this distinct fault, whose type is the complexType of the
140 distinct fault as defined in step 1.
- 141 3. A WSDL message MUST be defined for this distinct fault. This message MUST have one
142 part. The value of the WSDL part's *name* attribute MUST be *fault*, and the value of its
143 *element* attribute MUST refer by QName to the element of this distinct fault as defined in
144 step 2.
- 145 4. The WSDL operation MUST have a fault element for this distinct fault. The value of the
146 WSDL fault element's *name* attribute SHOULD be the same as the NCName of the fault
147 element defined in step 2, although it MAY choose to ignore this rule (for example to
148 avoid NCName collisions between fault elements defined in different namespaces). The
149 value of the WSDL fault element's *message* attribute MUST refer by QName to the
150 WSDL message element of this distinct fault as defined in step 3.

151 In addition to any operation-specific faults, all WSDL operations MAY also have a WSDL fault
 152 element whose name attribute has the value "BaseFault" and whose message element has the
 153 value "wsbf:BaseFaultMessage".

154 The following non-normative example defines a portType named "pt" with a single operation
 155 named "op" that has two distinct faults, "hisFault" and "herFault", in addition to a basic
 156 "baseFault". The "hisFault" element does not extend "BaseFault" with any additional information
 157 (i.e. it just defines a distinct fault type with the base information), while the "herFault" element
 158 extends "BaseFault" with an additional details element.

```

159 ...
160 <wsdl:definitions ...>
161   <wsdl:types>
162     <xsd:schema ...>
163       <!-- Type and element declarations for each distinct fault -->
164       <xsd:complexType name="HisFaultType">
165         <xsd:complexContent>
166           <xsd:extension base="wsbf:BaseFaultType"/>
167         </xsd:complexContent>
168       </xsd:complexType>
169       <xsd:element name="hisFault" type="tns:HisFaultType"/>
170
171       <xsd:complexType name="HerFaultType">
172         <xsd:complexContent>
173           <xsd:extension base="wsbf:BaseFaultType">
174             <xsd:sequence>
175               <xsd:element name="details" type="xsd:string"/>
176             </xsd:sequence>
177           </xsd:extension>
178         </xsd:complexContent>
179       </xsd:complexType>
180       <xsd:element name="herFault" type="tns:HerFaultType"/>
181
182     </xsd:schema>
183   </wsdl:types>
184
185   <!-- WSDL messages for each distinct fault -->
186   <wsdl:message name="hisFaultMessage">
187     <wsdl:part name="fault" element="tns:hisFault"/>
188   </wsdl:message>
189   <wsdl:message name="herFaultMessage">
190     <wsdl:part name="fault" element="tns:herFault"/>
191   </wsdl:message>
192
193   <wsdl:portType name="pt">
194     <wsdl:operation name="op">
195       <!-- WSDL operation fault elements for each distinct fault -->
196       <wsdl:input ... />
197       <wsdl:output ... />
198       <wsdl:fault name="hisFault" message="tns:hisFaultMessage"/>
199       <wsdl:fault name="herFault" message="tns:herFaultMessage"/>
200       <wsdl:fault name="BaseFault" message="wsbf:BaseFaultMessage"/>
201     </wsdl:operation>
202   </wsdl:portType>
203 </wsdl:definitions>

```

204

205 A Web service MAY return a more refined fault in place of a particular fault that is defined by a
 206 WSDL operation. To do so, a complexType MUST be defined that extends one of the faults found
 207 in the WSDL operation. The fault message that is returned by the service MUST then use the
 208 element of the fault from which the more refined fault is derived with an xsi:type attribute whose
 209 value is the QName of the complexType for the more refined fault.

210 For example, if an implementation of the "pt" example above wants to return a more refined
 211 version hisFault for the "op" operation, it must define a complexType of hisFault such as:

```

212 ... targetNamespace="http://example.com/ExtendedFaults" ...
213
214 <xsd:complexType name="ExtendedHisFaultType">

```

```
215     <xsd:complexContent>
216         <xsd:extension base="tns:HisFaultType">
217             <xsd:sequence>
218                 <xsd:element name="otherDetails" type="xsd:string"/>
219             </xsd:sequence>
220         </xsd:extension>
221     </xsd:complexContent>
222 </xsd:complexType>
```

223 This example service can then return a fault message for the “op” operation such as:

```
224 <hisFault>
225     xmlns:ef="http://example.com/ExtendedFaults"
226     xsi:type="ef:ExtendedHisFaultType">
227     <timeStamp>...</timeStamp>
228     ...
229     <otherDetails>...</otherDetails>
230 </hisFault>
```

231 4 Security Considerations

232 Fault messages may contain sensitive information. Policies should be defined such that such
233 sensitive content of fault messages are appropriately protected. For example, the security policy
234 can be specified to require that the sensitive content be encrypted based on WS-Security.
235 Depending on the context in which the fault occurred, it may also be desired that the integrity of
236 the message be ensured. In such cases, the security policy can reflect this by specifying the need
237 to digitally sign the resulting fault messages based on WS-Security specification.

238 5 References

239 [SOAP 1.2]

240 <http://www.w3.org/TR/soap12-part1/>

241

242 [OGSI]

243 http://www.ggf.org/ogsi-wg/drafts/draft-ggf-ogsi-gridservice-29_2003-04-05.pdf

244

245 [WS-Addressing]

246 <http://www-106.ibm.com/developerworks/library/specification/ws-add/>

247

248 [Web Services Security]

249 [http://www.oasis-open.org/committees/download.php/5531/oasis-200401-wss-soap-](http://www.oasis-open.org/committees/download.php/5531/oasis-200401-wss-soap-message-security-1.0.pdf)
250 [message-security-1.0.pdf](http://www.oasis-open.org/committees/download.php/5531/oasis-200401-wss-soap-message-security-1.0.pdf)

251

252 [XML-Infoset]

253 <http://www.w3.org/TR/xml-infoset/>

254

255 [XML]

256 <http://www.w3.org/TR/REC-xml>

257

258

259 **Appendix A. Acknowledgments**

260 Special thanks to the Global Grid Forum's Open Grid Services Infrastructure working group,
261 which defined the OGSF v1.0 [OGSI] specification which was a large inspiration for the ideas
262 expressed in this specification.

263 The following individuals were members of the committee during the development of this
264 specification:

265 Akhil Arora (Sun Microsystems), Tim Banks (IBM), Jeff Bohren (OpenNetwork), Conor Cahill
266 (AOL), Fred Carter (AmberPoint), Martin Chapman (Oracle), Glen Daniels (Sonic Software),
267 Thomas Freund (IBM), Stephen Graham (IBM), Anish Karmarkar (Oracle), Hideharu Kato
268 (Hitachi), David Levine (IBM), Paul Lipton (Computer Associates), Mark Little (Arjuna
269 Technologies Limited), Lily Liu (WebMethods, Inc.), Tom Maguire (IBM), Susan Malaika (IBM),
270 David Martin (IBM), Samuel Meder (ArgonneNational Laboratory), Jeff Mischinsky (Oracle),
271 Bryan Murray (Hewlett-Packard), Dave Orchard (BEA Systems, Inc.), Savas Parastatidis
272 (Individual), Greg Pavlik (Oracle), Mark Peel (Novell), Alain Regnier (Ricoh Company, Ltd.), Ian
273 Robinson (IBM), Junaid Saiyed (Sun Microsystems), Igor Sedukhin (Computer Associates),
274 Hitoshi Sekine (Ricoh Company, Ltd.), Frank Siebenlist (ArgonneNational Laboratory), David
275 Snelling (Fujitsu), Latha Srinivasan (Hewlett-Packard), John Tollefsrud (Sun Microsystems), Jem
276 Treadwell (Hewlett-Packard), Steve Tuecke (ArgonneNational Laboratory), William Vambenepe
277 (Hewlett-Packard), Katy Warr (IBM), Alan Weissberger (NEC Corporation), and Pete Wenzel
278 (SeeBeyond Technology Corporation)

279 Appendix B. Revision History

Rev	Date	By Whom	What
wd-01	2004-06-02	Lily Liu, Sam Meder	Initial version created from submission by contributing companies. Minor modifications made to reflect OASIS formatting.
wd-02	2004-06-10	Sam Meder	Consistency fixes from Ian Robinson Updated namespaces Cleaned up the references

280 Appendix C. Notices

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

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

293 Copyright © OASIS Open 2004. *All Rights Reserved.*

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

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

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

310

311 Appendix D. XML Schema

312 The XML types and elements used in this specification are defined in the following
313 XML Schema:

```

314 <?xml version="1.0" encoding="UTF-8"?>
315 <!--
316 OASIS takes no position regarding the validity or scope of any intellectual
317 property or other rights that might be claimed to pertain to the implementation
318 or use of the technology described in this document or the extent to which any
319 license under such rights might or might not be available; neither does it
320 represent that it has made any effort to identify any such rights. Information
321 on OASIS's procedures with respect to rights in OASIS specifications can be
322 found at the OASIS website. Copies of claims of rights made available for
323 publication and any assurances of licenses to be made available, or the result
324 of an attempt made to obtain a general license or permission for the use of such
325 proprietary rights by implementors or users of this specification, can be
326 obtained from the OASIS Executive Director.
327
328 OASIS invites any interested party to bring to its attention any copyrights,
329 patents or patent applications, or other proprietary rights which may cover
330 technology that may be required to implement this specification. Please address
331 the information to the OASIS Executive Director.
332
333 Copyright (C) OASIS Open (2004). All Rights Reserved.
334
335 This document and translations of it may be copied and furnished to others, and
336 derivative works that comment on or otherwise explain it or assist in its
337 implementation may be prepared, copied, published and distributed, in whole or
338 in part, without restriction of any kind, provided that the above copyright
339 notice and this paragraph are included on all such copies and derivative works.
340 However, this document itself may not be modified in any way, such as by
341 removing the copyright notice or references to OASIS, except as needed for the
342 purpose of developing OASIS specifications, in which case the procedures for
343 copyrights defined in the OASIS Intellectual Property Rights document must be
344 followed, or as required to translate it into languages other than English.
345
346 The limited permissions granted above are perpetual and will not be revoked by
347 OASIS or its successors or assigns.
348
349 This document and the information contained herein is provided on an "AS IS"
350 basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT
351 LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE
352 ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
353 PARTICULAR PURPOSE.
354 -->
355
356 <xsd:schema
357   xmlns="http://www.w3.org/2001/XMLSchema"
358   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
359   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
360   xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
361   xmlns:wsbf=
362     "http://docs.oasis-open.org/wsr/2004/06/wsr/WS-BaseFaults-1.2-draft-
363 02.xsd"
364   targetNamespace=
365     "http://docs.oasis-open.org/wsr/2004/06/wsr/WS-BaseFaults-1.2-draft-
366 02.xsd">
367   <xsd:import
368     namespace="http://schemas.xmlsoap.org/ws/2003/03/addressing"
369     schemaLocation=
370       "http://schemas.xmlsoap.org/ws/2003/03/addressing" />
371
372   <xsd:import namespace="http://www.w3.org/XML/1998/namespace"
373     schemaLocation="http://www.w3.org/2001/xml.xsd">
374   <xsd:annotation>
375   <xsd:documentation>

```

```
376         Get access to the xml: attribute groups for xml:lang as declared on
377 'schema'
378         and 'documentation' below
379     </xsd:documentation>
380 </xsd:annotation>
381 </xsd:import>
382 <!-- ===== BaseFault Types ===== -->
383
384 <xsd:element name="BaseFault" type="wsbf:BaseFaultType"/>
385
386 <xsd:complexType name="BaseFaultType">
387     <xsd:sequence>
388         <xsd:element name="Timestamp" type="xsd:dateTime"
389             minOccurs="1" maxOccurs="1"/>
390         <xsd:element name="Originator" type="wsa:EndpointReferenceType"
391             minOccurs="0" maxOccurs="1"/>
392         <xsd:element name="ErrorCode"
393             minOccurs="0" maxOccurs="1">
394             <xsd:complexType>
395                 <xsd:complexContent mixed="true">
396                     <xsd:extension base="xsd:anyType">
397                         <xsd:attribute name="dialect" type="xsd:anyURI"
398                             use="required"/>
399                     </xsd:extension>
400                 </xsd:complexContent>
401             </xsd:complexType>
402         </xsd:element>
403
404         <xsd:element name="Description"
405             minOccurs="0" maxOccurs="unbounded">
406             <xsd:complexType>
407                 <xsd:simpleContent>
408                     <xsd:extension base="xsd:string">
409                         <xsd:attribute ref="xml:lang" use="optional"/>
410                     </xsd:extension>
411                 </xsd:simpleContent>
412             </xsd:complexType>
413         </xsd:element>
414
415         <xsd:element name="FaultCause" type="wsbf:BaseFaultType"
416             minOccurs="0" maxOccurs="unbounded"/>
417     </xsd:sequence>
418 </xsd:complexType>
419 </xsd:schema>
```

420

Appendix E. WSDL 1.1

421

The following illustrates the WSDL 1.1 for the Web service methods described in this

422

specification:

423

```
<?xml version="1.0" encoding="UTF-8"?>
```

424

```
<!--
```

425

```
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.
```

426

427

428

429

430

431

432

433

434

435

436

437

```
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.
```

438

439

440

441

442

```
Copyright (C) OASIS Open (2004). All Rights Reserved.
```

443

444

445

446

447

448

449

450

451

452

453

454

```
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 may 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.
```

455

456

457

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

458

459

460

461

462

463

```
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.
```

464

```
-->
<wSDL:definitions name="BaseFaults"
```

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

```
xmlns="http://schemas.xmlsoap.org/wSDL/"
xmlns:wSDL="http://schemas.xmlsoap.org/wSDL/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wsbf=
"http://docs.oasis-open.org/wsr/2004/06/wsr-WS-BaseFaults-1.2-draft-
02.xsd"
targetNamespace=
"http://docs.oasis-open.org/wsr/2004/06/wsr-WS-BaseFaults-1.2-draft-
02.wSDL">
<!-- ===== Types Definitions ===== -->
<wSDL:types>
  <xsd:schema >
    <xsd:import
      namespace=
        "http://docs.oasis-open.org/wsr/2004/06/wsr-WS-BaseFaults-1.2-
draft-02.xsd"
      schemaLocation=
```

```
484     "http://docs.oasis-open.org/wsr/2004/06/wsr/WS-BaseFaults-1.2-draft-  
485 02.xsd" />  
486     </xsd:schema>  
487 </wsdl:types>  
488  
489     <wsdl:message name="BaseFaultMessage" >  
490       <wsdl:part name="Fault" element="wsbf:BaseFault" />  
491     </wsdl:message>  
492 </wsdl:definitions>  
493
```