

# Web Services Base Faults (WS-BaseFaults)

Version 1.0

3/31/2004

## Authors

*Steve Tuecke (Globus / Argonne) (Editor)*

*Karl Czajkowski (Globus / USC/ISI)*

*Jeffrey Frey (IBM)*

*Ian Foster (Globus / Argonne)*

*Steve Graham (IBM)*

*Tom Maguire (IBM)*

*Igor Sedukhin (Computer Associates International)*

*David Snelling (Fujitsu Laboratories of Europe)*

*William Vambenepe (Hewlett-Packard)*

## Copyright Notice

© Copyright Computer Associates International Inc., Fujitsu Limited, Hewlett-Packard Development Company, International Business Machines Corporation and The University of Chicago 2003, 2004. All Rights Reserved.

Permission to copy and display this "Web Services Base Faults" Specification ("this Specification"), in any medium without fee or royalty is hereby granted, provided that you include the following on ALL copies of this Specification, or portions thereof, that you make:

1. A link or URL to this Specification at this location.
2. This Copyright Notice as shown in this Specification.

THIS SPECIFICATION IS PROVIDED "AS IS," AND COMPUTER ASSOCIATES INTERNATIONAL, FUJITSU LIMITED, HEWLETT-PACKARD DEVELOPMENT COMPANY, IBM AND THE UNIVERSITY OF CHICAGO (COLLECTIVELY, THE "COMPANIES") MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR TITLE; THAT THE CONTENTS OF THIS SPECIFICATION ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

THE COMPANIES WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO ANY USE OR DISTRIBUTION OF THIS SPECIFICATION.

The companies each agree to grant you a royalty-free license, under commercially reasonable terms and conditions, to their respective patents that they deem necessary to implement this Specification.

The names and trademarks of the Companies may NOT be used in any manner, including advertising or publicity pertaining to this Specification or its contents, without specific, written prior permission. Title to copyright in this Specification will at all times remain with the Companies.

No other rights are granted by implication, estoppel or otherwise.

PORTIONS OF THIS MATERIAL WERE PREPARED AS AN ACCOUNT OF WORK SPONSORED BY IBM CORPORATION AT UNIVERSITY OF CHICAGO'S ARGONNE NATIONAL LABORATORY. NEITHER THE AUTHORS, NOR THE UNITED STATES GOVERNMENT OR ANY AGENCY THEREOF, NOR THE UNIVERSITY OF CHICAGO, NOR IBM, NOR ANY OF THEIR EMPLOYEES OR OFFICERS, NOR ANY OTHER COPYRIGHT HOLDERS OR CONTRIBUTORS, MAKES ANY WARRANTY, EXPRESS OR IMPLIED, OR ASSUMES ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF ANY INFORMATION, APPARATUS, PRODUCT, OR PROCESS DISCLOSED, OR REPRESENTS THAT ITS USE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS. REFERENCE HEREIN TO ANY SPECIFIC COMMERCIAL PRODUCT, PROCESS, OR SERVICE BY TRADE NAME, TRADEMARK, MANUFACTURER, OR OTHERWISE, DOES NOT NECESSARILY CONSTITUTE OR IMPLY ITS ENDORSEMENT, RECOMMENDATION, OR FAVORING BY IBM, THE UNITED STATES GOVERNMENT OR ANY AGENCY THEREOF OR ANY OTHER COPYRIGHT HOLDERS OR CONTRIBUTORS. THE VIEW AND OPINIONS OF AUTHORS EXPRESSED HEREIN DO NOT NECESSARILY STATE OR REFLECT THOSE OF IBM, THE UNITED STATES GOVERNMENT OR ANY AGENCY THEREOF, OR THE ENTITY BY WHICH AN AUTHOR MAY BE EMPLOYED.

This manuscript has been created in part by the University of Chicago as Operator of Argonne National Laboratory ("Argonne") under Contract No. W-31-109-ENG-38 with the U.S. Department of Energy. The U.S. Government retains for itself, and others acting on its behalf, a paid-up, nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

## **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 WS-BaseFaults specification is an initial draft release and is provided for review and evaluation only. The Companies hope to solicit your contributions and suggestions in the near future. The Companies make no warranties or representations regarding the specifications in any manner whatsoever.

## Table of Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>4</b>
1.1	GOALS AND REQUIREMENTS .....	4
1.1.1	<i>Requirements</i> .....	4
1.1.2	<i>Non-Goals</i> .....	4
1.2	NOTATIONAL CONVENTIONS .....	4
1.3	NAMESPACES .....	4
<b>2</b>	<b>BASE FAULT TYPE .....</b>	<b>5</b>
<b>3</b>	<b>USE OF BASE FAULTS IN WSDL 1.1.....</b>	<b>6</b>
<b>4</b>	<b>SECURITY CONSIDERATIONS .....</b>	<b>8</b>
<b>5</b>	<b>ACKNOWLEDGEMENTS .....</b>	<b>8</b>
<b>6</b>	<b>REFERENCES .....</b>	<b>8</b>
	<b>APPENDIX I – XML SCHEMA .....</b>	<b>9</b>
	<b>APPENDIX II – WSDL 1.1.....</b>	<b>10</b>

# 1 Introduction

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

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

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

WS-BaseFaults is inspired by a portion of the Global Grid Forum's "Open Grid Services Infrastructure (OGSI) Version 1.0" specification [OGSI].

## 1.1 Goals and Requirements

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

### 1.1.1 Requirements

This specification intends to meet the following requirements:

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

### 1.1.2 Non-Goals

The following topics are outside the scope of this specification:

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

## 1.2 Notational Conventions

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

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

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

## 1.3 Namespaces

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://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults
wsa	http://schemas.xmlsoap.org/ws/2003/02/addressing

## 2 Base Fault Type

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

```
<BaseFault>
  <Timestamp>xsd:dateTime</Timestamp>
  <OriginatorReference>
    wsa:EndpointReferenceType
  </OriginatorReference> ?
  <ErrorCode dialect="anyURI">xsd:string</ErrorCode> ?
  <Description>xsd:string</Description> *
  <FaultCause>wsbf:BaseFault</FaultCause> *
</BaseFault>
```

### /wsbf:BaseFault/Timestamp

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

### /wsbf:BaseFault/OriginatorReference

This OPTIONAL element is a WS-Addressing [WS-Addressing] EndpointReference of the Web service that generated the fault. This element MAY be omitted if the fault originator is clearly implied by the context in which the fault appears (for example in a simple request response message exchange). One use of this element is in a situation of nested faults. The outer-most fault may use this component to reference the actual original source of the fault condition.

### /wsbf:BaseFault/ErrorCode

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

### /wsbf:BaseFault/Description

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

/wsbf:BaseFault/FaultCause

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

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

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

### 3 Use of Base Faults in WSDL 1.1

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

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

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

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

```
...
<wsdl:definitions ...>
  <wsdl:types>
```

```

<xsd:schema ...>
  <!-- Type and element declarations for each distinct fault -->
  <xsd:complexType name="HisFaultType">
    <xsd:complexContent>
      <xsd:extension base="wsbf:BaseFaultType"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="hisFault" type="tns:HisFaultType"/>

  <xsd:complexType name="HerFaultType">
    <xsd:complexContent>
      <xsd:extension base="wsbf:BaseFaultType">
        <xsd:sequence>
          <xsd:element name="details" type="xsd:string"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="herFault" type="tns:HerFaultType"/>

</xsd:schema>
</wsdl:types>

<!-- WSDL messages for each distinct fault -->
<wsdl:message name="hisFaultMessage">
  <wsdl:part name="fault" element="tns:hisFault"/>
</wsdl:message>
<wsdl:message name="herFaultMessage">
  <wsdl:part name="fault" element="tns:herFault"/>
</wsdl:message>

<wsdl:portType name="pt">
  <wsdl:operation name="op">
    <!-- WSDL operation fault elements for each distinct fault -->
    <wsdl:input ... />
    <wsdl:output ... />
    <wsdl:fault name="hisFault" message="tns:hisFaultMessage"/>
    <wsdl:fault name="herFault" message="tns:herFaultMessage"/>
    <wsdl:fault name="BaseFault" message="wsbf:BaseFaultMessage"/>
  </wsdl:operation>
</wsdl:portType>
</wsdl:definitions>

```

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

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

```
... targetNamespace="http://example.com/ExtendedFaults" ...
```

```
<xsd:complexType name="ExtendedHisFaultType">
  <xsd:complexContent>
    <xsd:extension base="tns:HisFaultType">
      <xsd:sequence>
        <xsd:element name="otherDetails" type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

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

```
<hisFault>
  xmlns:ef="http://example.com/ExtendedFaults"
  xsi:type="ef:ExtendedHisFaultType">
  <timeStamp>...</timeStamp>
  ...
  <otherDetails>...</otherDetails>
</hisFault>
```

## 4 Security Considerations

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

## 5 Acknowledgements

Special thanks to the Global Grid Forum's Open Grid Services Infrastructure working group, which defined the OGSF v1.0 [OGSI] specification from which WS-BaseFaults was inspired.

This specification has been developed as a result of joint work with many individuals and teams. The authors wish to acknowledge the contributions from many people, including:

Bryan Murray, Nataraj Nagaratnam, Jay Unger.

## 6 References

### [SOAP 1.2]

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

### [OGSI]

*GGF GFD.15 "Open Grid Services Infrastructure (OGSI) Version 1.0". Available at <http://forge.gridforum.org/projects/ogsi-wg>*

### [WS-Addressing]

<http://www.ibm.com/developerworks/webservices/library/ws-add/>

### [WS-Security]

<http://www.oasis-open.org/committees/download.php/5531/oasis-200401-wss-soap-message-security-1.0.pdf>

### [XML-Infoset]

W3C Recommendation [“XML Information Set”](http://www.w3.org/TR/xml-infoset/). Available at <http://www.w3.org/TR/xml-infoset/>

## [XML]

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

## Appendix I – XML Schema

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

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

  Copyright Notice

  (c) Copyright Computer Associates International, Inc.,
  Fujitsu Limited, Hewlett-Packard Development Company,
  International Business Machines Corporation and
  The University of Chicago 2003, 2004. All Rights Reserved.
-->
<xsd:schema
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
  xmlns:wsbf=
    "http://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults"
  targetNamespace=
    "http://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults">

  <xsd:import
    namespace="http://schemas.xmlsoap.org/ws/2003/03/addressing"
    schemaLocation=
      "http://schemas.xmlsoap.org/ws/2003/03/addressing" />

  <xsd:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd">
    <xsd:annotation>
      <xsd:documentation>
        Get access to the xml: attribute groups for xml:lang as
        declared on 'schema'
        and 'documentation' below
      </xsd:documentation>
    </xsd:annotation>
  </xsd:import>
<!-- ===== BaseFault Types ===== -->

  <xsd:element name="BaseFault" type="wsbf:BaseFaultType"/>

  <xsd:complexType name="BaseFaultType">
    <xsd:sequence>
      <xsd:element name="Timestamp" type="xsd:dateTime"
        minOccurs="1" maxOccurs="1"/>
      <xsd:element name="Originator" type="wsa:EndpointReferenceType"
        minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```

    <xsd:element name="ErrorCode"
      minOccurs="0" maxOccurs="1">
      <xsd:complexType>
        <xsd:complexContent mixed="true">
          <xsd:extension base="xsd:anyType">
            <xsd:attribute name="dialect" type="xsd:anyURI"
              use="required"/>
          </xsd:extension>
        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>

    <xsd:element name="Description"
      minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
            <xsd:attribute ref="xml:lang" use="optional"/>
          </xsd:extension>
        </xsd:simpleContent>
      </xsd:complexType>
    </xsd:element>

    <xsd:element name="FaultCause" type="wsbf:BaseFaultType"
      minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:schema>

```

## Appendix II – WSDL 1.1

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

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  Legal Disclaimer

  Copyright Notice

  (c) Copyright Computer Associates International, Inc.,
  Fujitsu Limited, Hewlett-Packard Development Company,
  International Business Machines Corporation and
  The University of Chicago 2003, 2004. All Rights Reserved.
-->
<wsdl:definitions name="BaseFaults"
  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://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults"
  targetNamespace=
    "http://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults">
<!-- ===== Types Definitions ===== -->
<wsdl:types>

```

```
<xsd:schema >
  <xsd:import
    namespace=
      "http://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults"
    schemaLocation=
      "http://www.ibm.com/xmlns/stdwip/web-services/WS-BaseFaults.xsd"/>
</xsd:schema>
</wsdl:types>

<wsdl:message name="BaseFaultMessage" >
  <wsdl:part name="Fault" element="wsbf:BaseFault" />
</wsdl:message>
</wsdl:definitions>
```