1

2

# Web Services Resource Lifetime 1.2 (WS-ResourceLifetime)
## Working Draft 03, 10 June 2004

6

**Document identifier:**
    wsrf-WS-ResourceLifetime-1.2-draft-03

**Location:**
    http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-1.2-draft-03.pdf

**Editors:**
    Latha Srinivasan, Hewlett Packard Company <Latha.Srinivasan@hp.com>
    Tim Banks, IBM <Tim_Banks@uk.ibm.com>

**Abstract:**
    The definition of a WS-Resource, which is expressed in terms of a stateful resource and
    its relationship with a Web service, is defined in the document titled "Modeling Stateful
    Resources with Web services" [State Paper]. This specification defines message
    exchanges to standardize the means by which a WS-Resource may be destroyed, and
    resource properties [WS-ResourceProperties] that may be used to inspect and monitor
    the lifetime of a WS-Resource. This specification defines two means of destroying a WS-
    Resource: immediate destruction and time-based, scheduled destruction.

22

**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

# 1 Introduction

In this document, we consider a distributed computing environment consisting of WS-Resources. The definition of WS-Resource, in terms of a stateful resource and its relationship with a Web service, is detailed in "Modeling Stateful Resources with Web services" [State Paper].

The lifetime of a WS-Resource is defined as the period between its instantiation and its destruction. The WS-ResourceLifetime specification standardizes the means by which a WS-Resource can be destroyed. The specification also defines the means by which the lifetime of a WS-Resource can be monitored. However, this specification does not prescribe (nor proscribe) the means by which a WS-Resource is created.

Normally, a service requestor's interest in a WS-Resource is for some period of time - rarely is it indefinite. In many scenarios, it is appropriate for clients of a WS-Resource to cause its immediate destruction. The immediate destruction of a WS-Resource may be accomplished using the message exchanges defined in this specification.

In addition, this specification defines the means by which a resource may be destroyed after a period of time. In a distributed computing environment, a client may become disconnected from the service provider's endpoint and therefore may be unable to, or unwilling to, cause the immediate destruction of the WS-Resource. This specification defines the means by which any client of a WS-Resource may establish and extend the scheduled termination time of a WS-Resource. If that time expires, the WS-Resource may *self-destruct* without the need for an explicit destroy request message from a client. Periodically extending the termination time of a WS-Resource can serve to extend its lifetime. WS-ResourceLifetime defines a standard message exchange by which a service requestor can establish and renew a scheduled termination time for the WS-Resource, and defines the circumstances under which a service requestor can determine that this termination time has elapsed.

A service requestor may want to determine the current time and the termination time of a WS-Resource. WS-ResourceLifetime defines resource properties, as defined in [WS-ResourceProperties], for accessing this information.

WS-ResourceLifetime 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-ResourceLifetime is to standardize the terminology, concepts, message exchanges, WSDL and XML needed to monitor the lifetime of, and destroy, WS-Resources as defined in [State Paper].

### 1.1.1 Requirements

This specification intends to meet the following requirements:

- Define the standard message exchange by which a requestor can request the immediate destruction of a WS-Resource.
- Define the means by which a service requestor can set an initial termination time for the scheduled termination of a WS-Resource.
- Define the means by which a service requestor can update the termination time associated with a WS-Resource that is scheduled for termination.
- Define the means by which a service requestor can determine the current termination time as known by a WS-Resource.

110    This specification MUST NOT require entities in the system to share synchronized clocks.

## 1.1.2 Non-Goals

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

113    • It is not an objective of this specification to define the message exchanges representing
114      the function of a WS-Resource factory. Factory requirements are too varied to allow a
115      general-purpose factory message exchange to be usefully defined. However, the factory
116      pattern is described in more detail in [State Paper].

## 1.2 Terminology

118    The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD",
119    "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be
120    interpreted as described in RFC 2119.

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

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

## 1.3 Namespaces

130

131 The following namespaces are used in this document:

| Prefix | Namespace |
|--------|-----------|
| s12 | http://www.w3.org/2003/05/soap-envelope |
| wsp | http://schemas.xmlsoap.org/ws/2002/12/policy |
| wsa | http://schemas.xmlsoap.org/ws/2003/02/addressing |
| wsrp | http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-1.2-draft-01.xsd |
| wsrpw | http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-1.2-draft-01.wsdl |
| wsbf | http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-draft-01.wsdl |
| wsbfx | http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-draft-01.xsd |
| wsrl | http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-1.2-draft-01.xsd |
| wsrlw | http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-1.2-draft-01.wsdl |
| wstop | http://docs.oasis-open.org/wsn/2004/06/wsn-WS-Topics-1.2-draft-01.pdf |
| xsd | http://www.w3.org/2001/XMLSchema |
| xsi | http://www.w3.org/2001/XMLSchema-instance |

# 2 Terminology and Concepts

133 This section specifies the notations, namespaces, and terminology used in this specification.

134 **WS-Resource:**

135 • A Web service having an association with a stateful resource, where the stateful resource
136 is defined by a resource properties document and the association is expressed by means
137 of an attachment of the resource properties document to the WSDL port type of the
138 service.

139 **Implied Resource Pattern:**

140 The way WS-Addressing must be used to designate the stateful resource component of
141 the WS-Resource to be used in the execution of message exchanges.

142 An EndpointReference that follows the implied resource pattern may include a
143 ReferenceProperties child element that identifies the stateful resource component of the
144 WS-Resource to be used in the execution of all message exchanges performed using
145 this EndpointReference.

146 A message that follows the implied resource pattern MUST be sent to a Web service
147 referred to by an EndpointReference that follows the implied resource pattern, and MUST
148 include the ReferenceProperties information from that EndpointReference, if present, as
149 specified by WS-Addressing.

150 • A Web service that follows the implied resource pattern MAY use the
151 ReferenceProperties information from a message that follows the implied resource
152 pattern in order to identify the stateful resource to be used in the execution requested by
153 that message.

154 WS-Resource Qualified Endpoint Reference:

155 An Endpoint Reference used to refer to a WS-Resource composed of a Web service and
156 a stateful resource.

157 • A stateful resource identifier MAY be contained within the ReferenceProperties element
158 of the Endpoint Reference.

159 • The address of the Web service associated with the WS-Resource must be contained in
160 the Address element of the Endpoint Reference.

161 **Resource Property:**

162 • A resource property is a piece of information defined as part of the state model of a WS-
163 Resource.

164 • A resource property may reflect a part of the resource's state, meta-data, manageability
165 information, etc.

166 **Resource Properties Document:**

167 • The XML document representing a logical composition of resource property elements.
168 The resource properties document defines a particular view or projection of the state data
169 implemented by the WS-Resource.

170 • The *type* (e.g. the XML Schema definition of the root element) of a resource properties
171 document is associated with the WSDL portType defining the Web service interface. This
172 association is the basis of the WS-Resource definition. All instances of a particular WS-

173  Resource type MUST implement a logical resource properties document of the type
174  declared in the WSDL portType.

**Resource Property Element:**

176  • The XML representation of a resource property.

177  • A resource property element MUST appear as the immediate child of the root element of
178  a resource properties document.

179  • A resource property element MUST be an XML global element definition (GED), and is
180  uniquely identified by QName.

**Resource Property Value:**
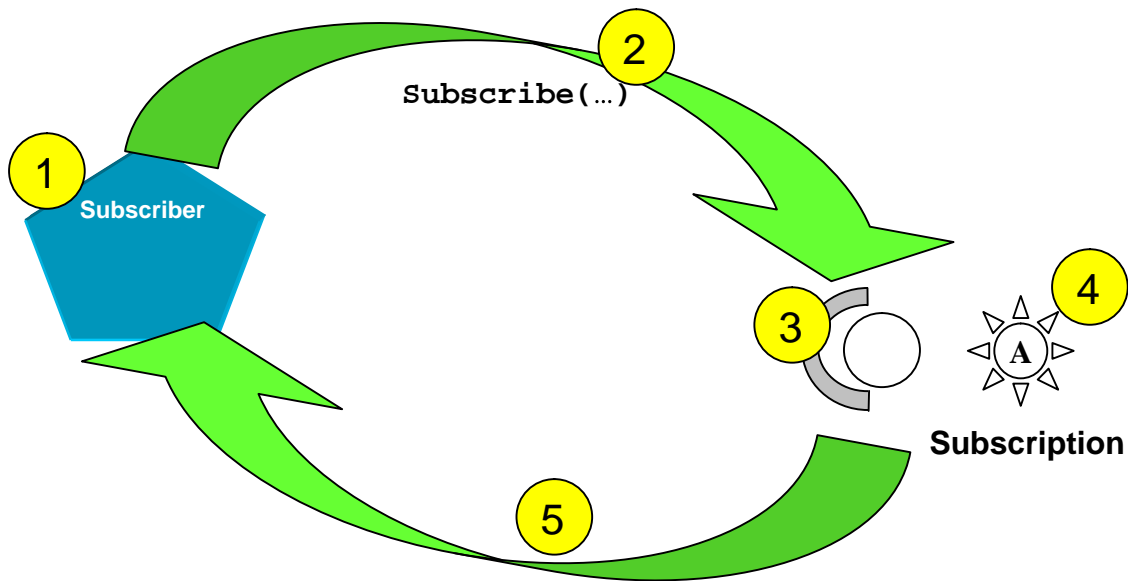
182  • The value(s) associated with a resource property.

## 183 **3 Example**

184 Consider the case of a subscription entity within a notification system such as WS-Notification
185 [WS-Notification]. This situation is depicted in the following figure:



186 *Figure 1 - Example WS-Resource Creation*

187 A service requestor (1), playing the role of a subscriber, sends a subscribe message (2) to a
188 NotificationProducer (3) because it wishes to receive notifications related to a particular situation
189 such as a failure of a component. A subscription WS-Resource (4) is created as a result of the
190 subscribe message, and a WS-Resource qualified EndpointReference (5) [State Paper] is
191 returned to the requestor. As part of the application-specific understanding of the subscribe
192 message exchange, both the requestor and provider understand that part of the semantics of
193 processing a subscribe message is the creation (usually for a limited period of time) of a
194 subscription WS-Resource. The subscribe request message contains the initial scheduled
195 termination time of the subscription WS-Resource.

196 The endpoint reference that is returned as a result of the subscribe message is a WS-Resource
197 qualified endpoint reference as described in [State Paper]. It contains a stateful resource identifier
198 that refers to the newly-created subscription state represented by the WS-Resource. The
199 endpoint reference also contains the address of the Web service component of the WS-Resource
200 that implements the message exchanges defined by WS-Notification's SubscriptionManager
201 interface.

202 Subsequent to the creation of the subscription WS-Resource, the application-specific behavior of
203 delivering notifications continues. Occasionally, the subscriber may examine the subscription WS-
204 Resource using standard WS-ResourceLifetime resource properties to inquire about the
205 remaining time before the subscription WS-Resource may be destroyed. If the subscriber wishes
206 to extend the lifetime of the subscription WS-Resource beyond its scheduled termination time, it
207 sends a specific WS-ResourceLifetime message to the subscription WS-Resource identified by
208 its EndpointReference, prior to the expiration of its current scheduled termination time. The

209 response to this message contains the (potentially unchanged) termination time associated with
210 the subscription WS-Resource.

211 When the subscriber no longer wishes to receive notifications, it may cause the immediate
212 destruction of the subscription WS-Resource by sending another WS-ResourceLifetime message
213 to the WS-Resource through use of its EndpointReference. As another option, it may simply allow
214 the termination time of the subscription WS-Resource to expire, at which time the subscription
215 WS-Resource may be destroyed.

 

## 216   **4   Immediate Destruction**

217 A WS-Resource MAY support a message exchange pattern that allows a service requestor to
218 request its immediate destruction.

219 The format of the destroy request message is:

```
220 …
221    <wsrl:Destroy/>
222 …
```

223 The Destroy request message MUST follow the implied resource pattern, as defined in Section 2.

224 If the WS-Resource accepts the Destroy request message, upon receipt of this message the WS-
225 Resource MUST either (1) destroy the implied stateful resource component of the WS-Resource
226 and return the following DestroyResponse message to acknowledge successful destruction, or
227 (2) return a fault message indicating failure. Note that the destruction of the stateful resource
228 component of the WS-Resource effectively destroys the WS-Resource.

```
229 …
230    <wsrl:DestroyResponse />
231 …
```

232 The receipt of the DestroyResponse message serves as a confirmation of the destruction of the
233 WS-Resource. Once it has sent a DestroyResponse message, any further message exchanges
234 directed at the subject WS-Resource MUST respond with a fault. In the absence of any other fault
235 conditions that may take precedence this MUST be the "ResourceUnknown" fault message.

236 If the WS-Resource does not respond to the Destroy request message with the DestroyResponse
237 message, then it MUST send one of the following fault messages:

238 •   ResourceUnknownFault

239      o   The stateful resource identified in the message (which follows the implied resource
240         pattern) is not known to the Web service.

241 •   ResourceNotDestroyedFault

242      o   The WS-Resource could not be destroyed for some reason.

243 •   Others tbd.

244

245 Note: All faults generated must be compliant with the WS-BaseFaults [WS-BaseFaults]
246 specification.

## 247   **4.1 Example SOAP Encoding of the Destroy Message Exchange**

248 The following is a non-normative example of a Destroy request message using SOAP 1.2 [SOAP
249 1.2]:

```
250 <s12:Envelope
251    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
252    xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
253    xmlns:wsrl=
254 "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-
255 1.2-draft-01.xsd"
256    xmlns:ex="http://example.com/exampleNS">
257  <s12:Header>
258    <wsa:Action>
259        http://docs.oasis-open.org/wsrf/2004/06/WS-
```

```
260   ResourceLifetime/Destroy
261       </wsa:Action>
262       <wsa:To s12:mustUnderstand="1">
263           http://www.provider.org/ProviderEndpoint
264       </wsa:To>
265       <ex:ResourceDisambiguator>
266           uuid:84decd55-7d3f-65ad-ac44-675d9fce5d22
267       </ex:ResourceDisambiguator>
268     </s12:Header>
269     <s12:Body>
270       <wsrl:Destroy/>
271     </s12:Body>
272   </s12:Envelope>
```

273   The following is an example DestroyResponse message using SOAP 1.2 [SOAP 1.2]:

```
274   <s12:Envelope
275       xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
276       xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
277       xmlns:wsrl=
278   " http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-
279   1.2-draft-01.xsd"
280       xmlns:resp="http://www.other.org/otherNS">
281     <s12:Header>
282       <wsa:Action>
283           http://docs.oasis-open.org/wsrf/2004/06/WS-
284   ResourceLifetime/DestroyResponse
285       </wsa:Action>
286       <wsa:To s12:mustUnderstand="1">
287           http://www.requestor.org/someEndpoint
288       </wsa:To>
289       <resp:SomeResourceReference>
290           uuid:9fef5fec-6dc3-44a2-ba32-8680cace43f9
291       </resp:SomeResourceReference>
292     </s12:Header>
293     <s12:Body>
294       <wsrl:DestroyResponse />
295     </s12:Body>
296   </s12:Envelope>
```

# 5 Scheduled Destruction

298 A time-based approach MAY be used for managing the destruction of a WS-Resource. In this
299 case, the WS-Resource has an associated termination time that defines the time after which the
300 WS-Resource is expected to be destroyed and thus before which the WS-Resource can
301 reasonably be expected to be available. As defined in the following subsections, a WS-
302 Resource's termination time may be inspected through the TerminationTime resource property,
303 and may be changed using the SetTerminationTime request message.

304 Typical use of scheduled destruction is to allow a service requestor to keep a WS-Resource
305 active by adjusting the WS-Resource's termination time to some appropriate point in time using
306 the SetTerminationTime request message.

307 Note that termination time is not required to monotonically increase, nor is a service required to
308 accept a requested termination time. An implementation MAY refuse a request to adjust
309 termination time for various reasons, including, for example, to enforce a policy that allows
310 termination time to only change monotonically.

311 If a WS-Resource wishes to provide support for scheduled WS-Resource destruction, it MUST
312 support all of the message exchanges and resource properties specified in this section.

## 5.1 Regarding Time

314 This specification assumes that services and clients use the UTC global time standard,
315 expressed as type dateTime from XML Schema. Note that xsd:dateTime includes an optional
316 designation of a time zone. The use of the time zone designation is RECOMMENDED. In the
317 absence of the time zone designation, the xsd:dateTime value MUST be interpreted as universal
318 time (UTC).

319 The approach allows operations and resource properties to refer unambiguously to absolute
320 times. However, assuming the GMT time standard to represent time does *not* imply any particular
321 level of clock synchronization between clients and services. No specific accuracy of
322 synchronization is specified or expected by this specification, as this is a service-quality issue.

323 The scheduled destruction operations and resource properties have been designed to allow for
324 tolerance of lack of clock synchronization between clients and services. The CurrentTime
325 resource property may be used by a client to determine the clock skew between the client and the
326 service, within a margin of error determined by the round-trip latency of the message exchange to
327 retrieve that value. This clock skew and margin of error can then be factored into subsequent
328 decisions of when to send subsequent requests to change the termination time, and what
329 termination times to request. The skew can also be monitored and adjusted with each
330 SetTerminationTime message exchange, based on the CurrentTime that is returned from this
331 request. This approach can also be used, to a limited extent, to accommodate clocks that "jump"
332 either forward or backward in time.

## 5.2 Querying Current Time

334 In order to assist the service requestor in inspecting and setting a WS-Resource's termination
335 time without requiring a specific accuracy of clock synchronization between the service requestor
336 and the service provider, the WS-Resource MUST provide a resource property element that
337 provides the current time as it is known by the WS-Resource. The form of this resource property
338 element is:

339 …
340 ```
    <xsd:element name="CurrentTime" type="xsd:dateTime"/>
```
341 …

342 The resource properties definition of the WS-Resource MUST contain exactly one element of
343 QName wsrl:CurrentTime. The constraints on this element are as follows:

344 /wsrl:CurrentTime

345 A WS-Resource MUST NOT allow the CurrentTime resource property to be modified by a
346 SetResourceProperties request message as defined in [WS-ResourceProperties].

347 If the element does not include the time zone designation, the value of the element MUST be
348 interpreted as universal time (UTC).

## 349 **5.3 Determining Current Termination Time**

350 In order to allow the service requestor to determine the current termination time of a WS-
351 Resource, the WS-Resource MUST provide a resource property element that indicates the
352 current termination time of the WS-Resource. The form of this resource property element is:

```
353  …
354    <xsd:element name="TerminationTime" nillable="true"
355                type="xsd:dateTime" />
356  …
```

357 The resource properties definition of the WS-Resource MUST contain exactly one element of
358 QName wsrl:TerminationTime. The constraints on this element are as follows:

359 /wsrl:TerminationTime

360 The time, relative to the time source used by the WS-Resource, after which the WS-
361 Resource MAY be destroyed.

362 If the value of this resource property element contains the xsi:nil attribute with value "true"
363 then the lifetime of the WS-Resource is considered to be *indefinite*; that is, there is no
364 scheduled destruction time.

365 A WS-Resource MUST NOT allow the TerminationTime resource property to be modified
366 by a SetResourceProperties request message as defined in [WS-ResourceProperties].

367 If the element does not include the time zone designation, the value of the element MUST
368 be interpreted as universal time (UTC).

## 369 **5.4 Requesting Change to Termination Time**

370 The SetTerminationTime request message MUST be implemented by a WS-Resource supporting
371 scheduled destruction in order to allow a requestor to change its scheduled termination time. The
372 form of the SetTerminationTime request message is:

```
373    <wsrl:SetTerminationTime>
374      <wsrl:RequestedTerminationTime>
375          xsd:dateTime
376      </wsrl:RequestedTerminationTime>
377    </wsrl:SetTerminationTime>
```

378 The SetTerminationTime request message MUST follow the implied resource pattern, as defined
379 in Section 2.

380 Further constraints on the processing of the SetTerminationTime request message are as follows:

381 /wsrl:SetTerminationTime/wsrl:RequestedTerminationTime

382 This is the new WS-Resource termination time that is being requested by the client. This
383 value is interpreted relative to the time source known to the WS-Resource. If the element
384 does not include the time zone designation, the value of the element MUST be interpreted
385 as universal time (UTC).

386         If the value is "in the past" relative to the current time as known by the WS-Resource, then
387         the WS-Resource MAY be destroyed immediately. This facility provides the ability to support
388         an asynchronous form of immediate destruction.

389         If the value is xsi:nil, then the intent of the service requestor is to specify there is no
390         scheduled termination time for the WS-Resource. In such situations it is RECOMMENDED
391         that the WS-Resource support the immediate WS-Resource destruction operations
392         described in Section 4.

393 A WS-Resource that receives this message MAY reject the request to change the WS-
394 Resource's termination time for any reason (e.g. policy). In this case, a fault message MUST be
395 returned to the service requestor.

396 If a WS-Resource accepts the request to set the WS-Resource's termination time, it MUST
397 update the TerminationTime resource property of the WS-Resource to the value specified in the
398 message or to a value "in the future" relative to the requested time. If the SetTerminationTime
399 request is accepted, the WS-Resource MUST respond with the following message:

```
400    <wsrl:SetTerminationTimeResponse>
401      <wsrl:NewTerminationTime>
402         xsd:dateTime
403      </wsrl:NewTerminationTime>
404      <wsrl:CurrentTime>
405         xsd:dateTime
406      </wsrl:CurrentTime>
407    <wsrl:SetTerminationTimeResponse>
```

408 Further constraints on the SetTerminationTimeResponse message are as follows:

409 /wsrl:SetTerminationTimeResponse/wsrl:NewTerminationTime

410         This value MAY be "in the future" relative to the xsd:dateTime requested by the service
411         requestor in the SetTerminationTime request message.

412         This value reflects the new date and time at which the WS-Resource is scheduled to be
413         destroyed.

414         This value MUST also be reflected through the value of the TerminationTime resource
415         property.

416 /wsrl:SetTerminationTimeResponse/wsrl:CurrentTime

417         This value MUST be the time, as it is known by the WS-Resource, at which the WS-
418         Resource processed this SetTerminationTime request.

419 If the WS-Resource does not respond to the SetTerminationTime request message with the
420 SetTerminationTimeResponse message, then it MUST send one of the following fault messages :

421 •   ResourceUnknownFault

422       o   The stateful resource identified in the message (which follows the implied resource
423           pattern) is not known to the Web service.

424 •   UnableToSetTerminationTimeFault

425       o   The request for termination time could not be changed for some reason.

426 •   TerminationTimeChangeRejectedFault

427       o   In the case where a WS-Resource is willing to update its termination time, but only
428           with a value "in the past" relative to the requested termination time, then the WS-
429           Resource MAY include a "hint" in the TerminationTimeUnchangedFault message
430           indicating the time to which it is willing to extend TermationTime.

431 Note: All faults generated must be compliant with the WS-BaseFaults [WS-BaseFaults]
432 specification.

433

## 5.5 Example SOAP Encoding of the SetTerminationTime Message Exchange

The following is a non-normative example of a SetTerminationTime request message using SOAP 1.2 [SOAP 1.2]:

```
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
    xmlns:wsrl=
  "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-
1.2-draft-01.xsd"
    xmlns:ex="http://example.com/exampleNS">
  <s12:Header>
    <wsa:Action>
        http://docs.oasis-open.org/wsrf/2004/06/WS-
ResourceLifetime/SetTerminationTime
    </wsa:Action>
    <wsa:To s12:mustUnderstand="1">
        http://www.provider.org/ProviderEndpoint
    </wsa:To>
    <ex:ResourceDisambiguator>
        uuid:9fef5fec-6dc3-44a2-ba32-8680cace43f9
    </ex:ResourceDisambiguator>
  </s12:Header>
  <s12:Body>
    <wsrl:SetTerminationTime>
      <wsrl:RequestedTerminationTime>
        2001-12-31T12:00:00
      </wsrl:RequestedTerminationTime>
    </wsrl:SetTerminationTime>
  </s12:Body>
</s12:Envelope>
```

The following is an example SetTerminationTimeResponse message using SOAP 1.2 [SOAP 1.2]:

```
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
    xmlns:wsrl=
  " http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-
1.2-draft-01.xsd"
    xmlns:resp="http://www.other.org/otherNS">
  <s12:Header>
    <wsa:Action>
        http://docs.oasis-open.org/wsrf/2004/06/WS-
ResourceLifetime/SetTerminationTimeResponse
    </wsa:Action>
    <wsa:To s12:mustUnderstand="1">
        http://www.requestor.org/someEndpoint
    </wsa:To>
    <resp:SomeResourceReference>
        Disk_3
    </resp:SomeResourceReference>
```

```
485     </s12:Header>
486     <s12:Body>
487       <wsrl:SetTerminationTimeResponse>
488         <wsrl:NewTerminationTime>
489            2001-12-31T12:00:00
490         </wsrl:NewTerminationTime>
491         <wsrl:CurrentTime>
492            2001-12-31T11:00:00
493         </wsrl:CurrentTime>
494       </wsrl:SetTerminationTimeResponse>
495     </s12:Body>
496   </s12:Envelope>
```

## 5.6 Termination Time Expiration

498   If the service requestor fails to successfully update the termination time of a WS-Resource before
499   the termination time expires, the WS-Resource MAY be destroyed and therefore no longer be
500   accessible. Termination time has expired when the termination time of the WS-Resource (as
501   reflected by the value of the WS-Resource's TerminationTime resource property element) is "in
502   the past" relative to the current time as expressed in the value of the WS-Resource's CurrentTime
503   resource property element.

504   The specific mechanisms employed to destroy the WS-Resource after termination time has
505   expired is implementation dependent. An implementation MAY delay destruction of the WS-
506   Resource at its own discretion. The requestor MUST NOT depend on the destruction of the WS-
507   Resource occurring at termination time expiration but SHOULD assume that the WS-Resource is
508   no longer accessible after termination time has expired.

509

# 6  Notification of Resource Destruction

A WS-Resource MAY choose to support the pattern of notifying interested parties when it is destroyed. If a WS-Resource chooses to support this pattern and if the WS-Resource uses WS-Notification [WS-Notification] to implement this pattern, then it MUST follow the approach described in this section. An implementation MAY choose to not support this pattern, or it MAY choose to do so using some means other than WS-Notification; in such circumstances, the implementation MAY ignore the approach described in this section.

If the WS-Resource is also a NotificationProducer, according to the WS-BaseNotification specification [WS-BaseNotification], then it SHOULD provide a topic [WS-Topics] to allow requestors to subscribe for notification of its destruction. The notification applies to both immediate and scheduled destruction. The form of the topic is:

```
<wstop:TopicSpace name="ResourceLifetime"
   targetNamespace=
     "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-
1.2-draft-01.xsd"
… >
   <wstop:Topic name="ResourceTermination" …>
      <wstop:MessagePattern>
         <wsrp:QueryExpression
           dialect="http://www.w3.org/TR/1999/REC-xpath-19991116" >
              boolean(/*/TerminationNotification)
         </wsrp:QueryExpression>
      </wstop:MessagePattern>
…
```

The value of /wstop:Topic/@MessageTypes is implementation-dependent; this specification does not define the exact content of the notification messages produced on this topic. However, the notification message associated with this topic MUST contain the following element:

```
<wsrl:TerminationNotification>
  <wsrl:TerminationTime>xsd:dateTime</wsrl:TerminationTime>
  <wsrl:TerminationReason>xsd:any</wsrl:TerminationReason>?
</wsrl:TerminationNotification>
```

This constraint is specified in the /wstop:Topic/wstop:MessagePattern element. The TerminationNotification element is further constrained as follows:

/wsrl:TerminationTime

> This element contains the date and time when the WS-Resource was destroyed.

/wsrl:TerminationReason

> This OPTIONAL element contains an explanation of the situation surrounding the destruction of the WS-Resource. This element is specific to the type of the WS-Resource that was destroyed.

A requestor would send a subscribe request message, following the WS-Notification specification, specifying the "ResourceTermination" topic and referencing a chosen WS-Resource using a WS-Resource qualified endpoint reference [State Paper].

# 7 Security Considerations

This specification defines the message exchanges used to request the destruction of a WS-Resource, or to obtain information about the termination time of the WS-Resource. In this context, there are two categories of security aspects that need to be considered: (a) securing the message exchanges and (b) securing the operations that perform the WS-Resource destruction.

## 7.1 Securing the Message Exchanges

When messages are exchanged between a requestor and a WS-Resource in order to access or act upon the resource properties, it is strongly RECOMMENDED that the communication between them be secured using the mechanisms described in WS-Security. In order to properly secure messages, the body and all relevant headers need to be included in the digital signature so as to prove the integrity of the message. In addition the reference properties within an Endpoint Reference may be encrypted to ensure their privacy. In the event that a requestor communicates with a WS-Resource to access its resource properties, either directly through a query or indirectly through a notification of resource property state change, it is RECOMMENDED that a security context be established using the mechanisms described in WS-Trust [WS-Trust] and WS-SecureConversation [WS-SecureConversation].

It is common for communication between requestors and WS-Resources to exchange multiple messages. As a result, the usage profile is such that it is susceptible to key attacks. For this reason it is strongly RECOMMENDED that the keys used to secure the channel be changed frequently. This "re-keying" can be effected a number of ways. The following list outlines four common techniques:

- Attaching a nonce to each message and using it in a derived key function with the shared secret
- Using a derived key sequence and switch "generations"
- Closing and re-establishing a security context
- Exchanging new secrets between the parties

It should be noted that the mechanisms listed above are independent of the security context token (SCT) and secret returned when subscribed the first time. That is, the keys used to secure the channel during notifications may be independent of the key used to prove the right to subscribe with a NotificationSource.

The security context MAY be re-established using the mechanisms described in WS-Trust and WS-SecureConversation. Similarly, secrets can be exchanged using the mechanisms described in WS-Trust. Note, however, that the current shared secret SHOULD NOT be used to encrypt the new shared secret. Derived keys, the preferred solution from this list, can be specified using the mechanisms described in WS-SecureConversation.

The following list summarizes common classes of attacks that apply to this protocol and identifies the mechanism to prevent/mitigate the attacks:

- **Message alteration** – Alteration is prevented by including signatures of the message information using WS-Security.
- **Message disclosure** – Confidentiality is preserved by encrypting sensitive data using WS-Security.
- **Key integrity** – Key integrity is maintained by using the strongest algorithms possible (by comparing secured policies – see WS-Policy and WS-SecurityPolicy).

596 • **Authentication** – Authentication is established using the mechanisms described in WS-
597 Security and WS-Trust.  Each message is authenticated using the mechanisms described in
598 WS-Security.

599 • **Accountability** – Accountability is a function of the type of and string of the key and
600 algorithms being used.  In many cases, a strong symmetric key provides sufficient
601 accountability.  However, in some environments, strong PKI signatures are required.

602 • **Availability** – Many services are subject to a variety of availability attacks.  Replay is a
603 common attack and it is RECOMMENDED that this be addressed as described in the
604 "Replay" item below.  Other attacks, such as network-level denial of service attacks are
605 harder to avoid and are outside the scope of this specification.  That said, care should be
606 taken to ensure that minimal processing be performed prior to any authenticating sequences.

607 • **Replay** – Messages may be replayed for a variety of reasons.  To detect and eliminate this
608 attack, mechanisms should be used to identify replayed messages such as the
609 timestamp/nonce outlined in WS-Security and the sequences outlined in WS-
610 ReliableMessaging.

## 7.2 Securing Resource Destruction

612 Given that WS-ResourceLifetime defines a mechanism to destroy WS-Resources, security
613 policies should be established to ensure that only authorized requestors can destroy a WS-
614 Resource. Authorization policies should be defined so that the implications of destroying a WS-
615 Resource either through immediate requests or by setting termination time, are considered. The
616 two approaches for destruction may be considered equivalent for authorization reasons. In other
617 words, an authorization policy that describes the ability to perform a Destroy operation on a WS-
618 Resource, conforming to the ImmediateResourceTermination portType, may also need to be
619 applied when the SetTerminationTime operation is performed on the same resource.

620 It should be noted that this specification does not allow modifications to the CurrentTime and
621 TerminationTime resource properties through the SetResourceProperty request message of WS-
622 ResourceProperties.  Therefore, there should be no authorization enforcement performed when
623 these resource properties are accessed using the Set request message;  however, it should be
624 left to the runtime to enforce the requirement as specified. Given a requestor can subscribe for
625 notification of the destruction of the resource using "ResourceLifetime" topic, the security
626 considerations specified in WS-Notification specification are applicable to this topic.

627

# 8 References

**[OGSI]**

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

**[State Paper]**

http://www.oasis-open.org/committees/download.php/6795/ws-modelingresources.pdf

**[WS-Addressing]**

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

**[WS-BaseNotification]**

http://docs.oasis-open.org/wsn/2004/06/wsn-WS-BaseNotification-1.2-draft-03.pdf

**[WS-Notification]**

http://www.oasis-open.org/committees/download.php/6661/WSNpubsub-1-0.pdf

**[WS-BaseFaults]**

http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-draft-01.pdf

**[WS-ResourceProperties]**

http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-1.2-draft-04.pdf

**[WS-SecureConversation]**

http://www-106.ibm.com/developerworks/library/ws-secon/

**[WS-Security]**

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

**[WS-Topics]**

http://docs.oasis-open.org/wsn/2004/06/wsn-WS-Topics-1.2-draft-01.pdf

**[WS-Trust]**

http://www-106.ibm.com/developerworks/library/ws-trust/

**[XML-Infoset]**

http://www.w3.org/TR/xml-infoset/

**[XML]**

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

# Appendix A. Acknowledgments

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

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

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

In addition, the following people made contributions to this specification:

Karl Czajkowski (Globus / USC/ISI), Donald F Ferguson (IBM), Ian Foster (Globus / Argonne), Jeffrey Frey (IBM), Frank Leymann (IBM), Nataraj Nagaratnam (IBM), Martin Nally (IBM), Tony Storey (IBM), Sanjiva Weerawarana (IBM)

# 687 Appendix B. XML Schema

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

```
689  <?xml version="1.0" encoding="UTF-8"?>
690  <!--
691
692  OASIS takes no position regarding the validity or scope of any
693  intellectual property or other rights that might be claimed to pertain
694  to the implementation or use of the technology described in this
695  document or the extent to which any license under such rights might or
696  might not be available; neither does it represent that it has made any
697  effort to identify any such rights. Information on OASIS's procedures
698  with respect to rights in OASIS specifications can be found at the
699  OASIS website. Copies of claims of rights made available for
700  publication and any assurances of licenses to be made available, or the
701  result of an attempt made to obtain a general license or permission for
702  the use of such proprietary rights by implementors or users of this
703  specification, can be obtained from the OASIS Executive Director.
704
705  OASIS invites any interested party to bring to its attention any
706  copyrights, patents or patent applications, or other proprietary rights
707  which may cover technology that may be required to implement this
708  specification. Please address the information to the OASIS Executive
709  Director.
710
711  Copyright (C) OASIS Open (2004). All Rights Reserved.
712
713  This document and translations of it may be copied and furnished to
714  others, and derivative works that comment on or otherwise explain it or
715  assist in its implementation may be prepared, copied, published and
716  distributed, in whole or in part, without restriction of any kind,
717  provided that the above copyright notice and this paragraph are
718  included on all such copies and derivative works. However, this
719  document itself may not be modified in any way, such as by removing the
720  copyright notice or references to OASIS, except as needed for the
721  purpose of developing OASIS specifications, in which case the
722  procedures for copyrights defined in the OASIS Intellectual Property
723  Rights document must be followed, or as required to translate it into
724  languages other than English.
725
726  The limited permissions granted above are perpetual and will not be
727  revoked by OASIS or its successors or assigns.
728
729  This document and the information contained herein is provided on an
730  "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED,
731  INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE
732  INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
733  WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
734
735  -->
736  <xsd:schema xmlns="http://www.w3.org/2001/XMLSchema"
737  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
738  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
739  xmlns:wsrl="http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-
740  ResourceLifetime-1.2-draft-01.xsd"
741  elementFormDefault="qualified" attributeFormDefault="unqualified"
742  targetNamespace=" http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-
743  ResourceLifetime-1.2-draft-01.xsd">
744  <!--
745
746    =============== Resource Property Related  ==================
```

```
747        -->
748   <!--
749
750    ==== Resource Properties for ScheduledResourceTermination ====
751      -->
752     <xsd:element name="CurrentTime" type="xsd:dateTime" />
753     <xsd:element name="TerminationTime" nillable="true"
754   type="xsd:dateTime" />
755   <!--
756
757    ============= Notification Message Related  ==================
758
759      -->
760   <xsd:element name="TerminationNotification">
761      <xsd:complexType>
762      <xsd:sequence>
763            <xsd:element name="TerminationTime" type="xsd:dateTime"
764   minOccurs="1" maxOccurs="1" />
765            <xsd:element name="TerminationReason" type="xsd:anyType"
766   minOccurs="0" maxOccurs="1" />
767      </xsd:sequence>
768      </xsd:complexType>
769    </xsd:element>
770    </xsd:schema>
```

# Appendix C. WSDL 1.1

771

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

```xml
774  <?xml version="1.0" encoding="utf-8"?>
775  <!—
776  OASIS takes no position regarding the validity or scope of any
777  intellectual property or other rights that might be claimed to pertain
778  to the implementation or use of the technology described in this
779  document or the extent to which any license under such rights might or
780  might not be available; neither does it represent that it has made any
781  effort to identify any such rights. Information on OASIS's procedures
782  with respect to rights in OASIS specifications can be found at the
783  OASIS website. Copies of claims of rights made available for
784  publication and any assurances of licenses to be made available, or the
785  result of an attempt made to obtain a general license or permission for
786  the use of such proprietary rights by implementors or users of this
787  specification, can be obtained from the OASIS Executive Director.
788
789  OASIS invites any interested party to bring to its attention any
790  copyrights, patents or patent applications, or other proprietary rights
791  which may cover technology that may be required to implement this
792  specification. Please address the information to the OASIS Executive
793  Director.
794
795  Copyright (C) OASIS Open (2004). All Rights Reserved.
796
797  This document and translations of it may be copied and furnished to
798  others, and derivative works that comment on or otherwise explain it or
799  assist in its implementation may be prepared, copied, published and
800  distributed, in whole or in part, without restriction of any kind,
801  provided that the above copyright notice and this paragraph are
802  included on all such copies and derivative works. However, this
803  document itself may not be modified in any way, such as by removing the
804  copyright notice or references to OASIS, except as needed for the
805  purpose of developing OASIS specifications, in which case the
806  procedures for copyrights defined in the OASIS Intellectual Property
807  Rights document must be followed, or as required to translate it into
808  languages other than English.
809
810  The limited permissions granted above are perpetual and will not be
811  revoked by OASIS or its successors or assigns.
812
813  This document and the information contained herein is provided on an
814  "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED,
815  INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE
816  INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
817  WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
818
819  -->
820
821  <wsdl:definitions name="WS-ResourceLifetime"
822    xmlns="http://schemas.xmlsoap.org/wsdl/"
823    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
824    xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
825    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

```
826        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
827        xmlns:wsbf=
828          "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-
829      draft-01.xsd"
830      xmlns:wsrp=
831        "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-
832      1.2-draft-01.xsd"
833        xmlns:wsrl=
834          "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-
835      ResourceLifetime-1.2-draft-01.xsd"
836      xmlns:wsrlw=
837          "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-
838      ResourceLifetime-1.2-draft-01.wsdl"
839        targetNamespace=
840          "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-
841      ResourceLifetime-1.2-draft-01.wsdl">
842
843      <!-- ==================== Types Definitions ===================== -->
844        <wsdl:types>
845          <xsd:schema
846            xmlns="http://www.w3.org/2001/XMLSchema"
847            targetNamespace=
848          http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-
849      1.2-draft-01.xsd
850      elementFormDefault="qualified" attributeFormDefault="unqualified"
851        >
852
853            <xsd:include
854               schemaLocation=
855          "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-
856      ResourceLifetime-1.2-draft-01.xsd"
857            />
858
859            <xsd:import
860                 namespace=
861                 "http://schemas.xmlsoap.org/ws/2003/03/addressing"
862                 schemaLocation=
863                 "http://schemas.xmlsoap.org/ws/2003/03/addressing"
864            />
865
866             <xsd:import
867                 namespace=
868             "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-
869      draft-01.xsd"
870                 schemaLocation=
871          "http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-
872      draft-01.xsd"
873            />
874
875      <!-- ==== Resource Properties for ScheduledResourceTermination ==== -->
876            <xsd:element name="ScheduledResourceTerminationRP" >
877                <xsd:complexType>
878                   <xsd:sequence>
879                      <xsd:element ref="wsrl:CurrentTime"
880                                   minOccurs="1" maxOccurs="1" />
881                      <xsd:element ref="wsrl:TerminationTime"
882                                   minOccurs="1" maxOccurs="1" />
883                   </xsd:sequence>
884                </xsd:complexType>
885            </xsd:element>
886
```

```
887     <!-- ====== Message Types for ImmediateResourceTermination  ======= -->
888
889             <xsd:element name="Destroy">
890                <xsd:complexType />
891             </xsd:element>
892
893             <xsd:element name="DestroyResponse" >
894                <xsd:complexType />
895             </xsd:element>
896
897             <xsd:complexType name="ResourceUnknownFaultType">
898                <xsd:complexContent>
899                   <xsd:extension base="wsbf:BaseFaultType"/>
900                </xsd:complexContent>
901             </xsd:complexType>
902             <xsd:element name="ResourceUnknownFault"
903                        type="wsrl:ResourceUnknownFaultType"/>
904
905             <xsd:complexType name="ResourceNotDestroyedFaultType">
906                <xsd:complexContent>
907                   <xsd:extension base="wsbf:BaseFaultType"/>
908                </xsd:complexContent>
909             </xsd:complexType>
910             <xsd:element name="ResourceNotDestroyedFault"
911                        type="wsrl:ResourceNotDestroyedFaultType"/>
912
913     <!-- ====== Message Types for ScheduledResourceTermination  ======= -->
914
915             <xsd:element name="SetTerminationTime">
916                <xsd:complexType>
917                   <xsd:sequence>
918                      <xsd:element name="RequestedTerminationTime"
919                              nillable="true"
920                              type="xsd:dateTime" />
921                   </xsd:sequence>
922                </xsd:complexType>
923             </xsd:element>
924
925             <xsd:element name="SetTerminationTimeResponse">
926                <xsd:complexType>
927                   <xsd:sequence>
928                      <xsd:element name="NewTerminationTime"
929                              nillable="true"
930                              type="xsd:dateTime" />
931                      <xsd:element name="CurrentTime"
932                              type="xsd:dateTime" />
933                   </xsd:sequence>
934                </xsd:complexType>
935             </xsd:element>
936
937             <xsd:complexType name="UnableToSetTerminationTimeFaultType">
938                <xsd:complexContent>
939                   <xsd:extension base="wsbf:BaseFaultType"/>
940                </xsd:complexContent>
941             </xsd:complexType>
942             <xsd:element name="UnableToSetTerminationTimeFault"
943                        type="wsrl:UnableToSetTerminationTimeFaultType"/>
944
945             <xsd:complexType name="TerminationTimeChangeRejectedFaultType">
946                <xsd:complexContent>
947                   <xsd:extension base="wsbf:BaseFaultType"/>
948                </xsd:complexContent>
```

```
949          </xsd:complexType>
950          <xsd:element name="TerminationTimeChangeRejectedFault"
951                       type="wsrl:TerminationTimeChangeRejectedFaultType"/>
952
953
954        </xsd:schema>
955      </wsdl:types>
956
957    <!-- ============= Message Definitions for Destroy ====================
958      Destroy()
959      returns: empty
960    -->
961      <wsdl:message name="DestroyRequest">
962         <wsdl:part name="DestroyRequest"
963               element="wsrl:Destroy" />
964      </wsdl:message>
965
966      <wsdl:message name="DestroyResponse">
967         <wsdl:part name="DestroyResponse"
968               element="wsrl:DestroyResponse" />
969      </wsdl:message>
970
971      <wsdl:message name="ResourceUnknownFault">
972         <wsdl:part name="ResourceUnknownFault"
973               element="wsrl:ResourceUnknownFault" />
974      </wsdl:message>
975
976      <wsdl:message name="ResourceNotDestroyedFault">
977         <wsdl:part name="ResourceNotDestroyedFault"
978               element="wsrl:ResourceNotDestroyedFault" />
979      </wsdl:message>
980
981    <!-- ===================== SetTerminationTime ===================
982      SetTerminationTime(xsd:dateTime)
983      returns: xsd:dateTime
984    -->
985
986      <wsdl:message name="SetTerminationTimeRequest">
987         <wsdl:part name="SetTerminationTimeRequest"
988               element="wsrl:SetTerminationTime" />
989      </wsdl:message>
990
991      <wsdl:message name="SetTerminationTimeResponse">
992         <wsdl:part name="SetTerminationTimeResponse"
993               element="wsrl:SetTerminationTimeResponse" />
994      </wsdl:message>
995
996      <wsdl:message name="UnableToSetTerminationTimeFault">
997         <wsdl:part name="UnableToSetTerminationTimeFault"
998               element="wsrl:UnableToSetTerminationTimeFault" />
999      </wsdl:message>
1000
1001      <wsdl:message name="TerminationTimeChangeRejectedFault">
1002         <wsdl:part name="TerminationTimeChangeRejectedFault"
1003               element="wsrl:TerminationTimeChangeRejectedFault" />
1004      </wsdl:message>
1005
1006    <!-- =================== PortType Definitions ===================== -->
1007      <wsdl:portType name="ImmediateResourceTermination">
1008         <wsdl:operation name="Destroy">
1009           <wsdl:input message="wsrlw:DestroyRequest" />
1010           <wsdl:output message="wsrlw:DestroyResponse" />
```

```
1011            <wsdl:fault name="ResourceUnknownFault"
1012                        message="wsrlw:ResourceUnknownFault" />
1013            <wsdl:fault name="ResourceNotDestroyedFault"
1014                        message="wsrlw:ResourceNotDestroyedFault" />
1015        </wsdl:operation>
1016      </wsdl:portType>
1017
1018      <wsdl:portType name="ScheduledResourceTermination"
1019        wsrp:ResourceProperties ="wsrl:ScheduledResourceTerminationRP">
1020        <wsdl:operation name="SetTerminationTime">
1021          <wsdl:input message="wsrlw:SetTerminationTimeRequest" />
1022          <wsdl:output message="wsrlw:SetTerminationTimeResponse" />
1023          <wsdl:fault name="ResourceUnknownFault"
1024                      message="wsrlw:ResourceUnknownFault" />
1025          <wsdl:fault name="UnableToSetTerminationTimeFault"
1026                      message="wsrlw:UnableToSetTerminationTimeFault" />
1027          <wsdl:fault name="TerminationTimeChangeRejectedFault"
1028                      message="wsrlw:TerminationTimeChangeRejectedFault" />
1029        </wsdl:operation>
1030      </wsdl:portType>
1031
1032    </wsdl:definitions>
1033
```

1034

# Appendix D. Revision History

*[This appendix is optional, but helpful. It should be removed for specifications that are at OASIS Standard level.]*

| Rev | Date | By Whom | What |
|---|---|---|---|
| wd-01 | 2004-05-21 | Latha Srinivasan | Initial version created from submission by contributing companies. Minor modifications made to reflect OASIS formatting and the following issues: WSRF2, WSRF3, WSRF14, WSRF33. |
| wd-02 | 2004-06-01 | Latha Srinivasan | Modification to Acknowledgments section to reflect TC list as per WS-RP draft spec. v 1.2 |
| Wd-03 | 2004-06-08 | Latha Srinivasan | Fixed namespaces to reflect 2004/06; replaced rogue verdana fonts with Arial; updated Acknowledgments section; added ElementFormDefault and attributeFormDefault to schema and XSD files; updated references to point to pdf versions of files; Fixed reference for WS-BaseNotification and replaced references to "lifecycle" with lifetime |

1038

# Appendix E. Notices

1039

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 (C) 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 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.

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.