# 3. Indicating Use of WS-Addressing

This specification supports a mechanism for indicating, in a WSDL description, that the endpoint conforms to the WS-Addressing specification. That mechanism uses WS-Policy Framework [<u>WS Policy 1.5 - Framework</u>].

## **3.1 WS-Policy Assertions**

The mechanism for indicating that a binding or endpoint conforms to the WS-Addressing specification is through the use of the Web Services Policy -Framework [<u>WS Policy 1.5 - Framework</u>] and Web Services Policy - Attachment [<u>WS Policy 1.5 - Attachment</u>] specifications. This specification defines <u>one policy</u> assertion, and two policy parameters.

For WSDL 1.1, these assertions may be attached to wsdll:port or wsdll:binding. For WSDL 2.0, they may be attached to wsdl20:endpoint or wsdl20:binding.

### 3.1.1 Addressing Assertion

The wsam:Addressing policy assertion <u>may contain policy parameters.</u> The meaning of this assertion, when present in a policy alternative, is that WS-Addressing is required to communicate with the subject. In order to indicate that the subject supports WS-Addressing but does not require its use, an additional policy alternative should be provided which does not contain this assertion. This may be done in WS-Policy compact form by adding the attribute wsp:Optional="true" to the wsam:Addressing assertion.

#### 3.1.2 AnonymousResponses Parameter

The wsam:AnonymousResponses element MAY be used as a policy <u>parameter</u> nested within the wsam:Addressing assertion in accordance with the rules laid down by WS-Policy Framework 1.5 section 4.3.2.

The appearance of this element within a policy alternative indicates that the endpoint expresses explicit support for request messages with response endpoint EPRs that contain the anonymous URI

("http://www.w3.org/2005/08/addressing/anonymous") as the value of [address]. In other words, the endpoint guarantees support for anonymous responses.

The absence of the wsam:AnonymousResponses policy <u>parameter</u> within a policy alternative does **not** indicate that the endpoint will not accept request messages with response endpoint EPRs that contain the anonymous URI as an address; it simply indicates the lack of any affirmation of support for anonymous URIs.

The None URI ("http://www.w3.org/2005/08/addressing/none") may appear as the value of [address] in place of the anonymous URI; this value MUST be accepted.

Deleted: is a nested policy container

Deleted: Assertion

Deleted: three

assertion

Deleted: assertion

Deleted: assertion

Deleted: Assertion

Deleted: assertion

Deleted: assertion

### 3.1.3 NonAnonymousResponses Parameter

The wsam:NonAnonymousResponses element MAY be used as a policy <u>parameter</u> nested within the Addressing assertion in accordance with the rules laid down by WS-Policy Framework 1.5 section 4.3.2.

**v**\_ \_ \_

The appearance of this element within a policy alternative indicates that the endpoint expresses explicit support for request messages with response endpoint EPRs that contain something other than the anonymous URI as the value of [address]. In other words, the endpoint guarantees support for non-

anonymous responses. This <u>parameter definition</u> is deliberately vague; its presence indicates that some non-anonymous addresses will be accepted but doesn't constrain what such an address might look like. A receiver can still reject a request that contains an address that it doesn't understand or that requires a binding it doesn't support.

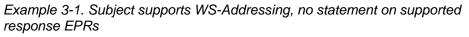
As with the other policy parameter, the absence of the

wsam:NonAnonymousResponses policy <u>parameter</u> within a policy alternative does **not** indicate that the endpoint will not accept request messages with response endpoint EPRs that contain something other than the anonymous URI address; it simply indicates the lack of any affirmation of support for them.

The None URI ("http://www.w3.org/2005/08/addressing/none") may appear as the value of [address] in place of a non-anonymous address; this value MUST be accepted.

### 3.1.4 Examples (Compact Form)

cwsp:Policy>



Example 3-2. Subject requires WS-Addressing, no statement on supported response EPRs

<pre><wsp.rollcy></wsp.rollcy></pre>	
<wsam:addressing></wsam:addressing>	
	Deleted:
	<wsp:policy></wsp:policy> ¶
Example 3-3. Subject supports WS-Addressing, explicitly and supports	Deleted: (and optionally)
anonymous and non-anonymous response EPRs	
<wsp:policy></wsp:policy>	
<wsam:addressing wsp:optional="true"></wsam:addressing>	
<pre><wsam:anonymousresponses></wsam:anonymousresponses></pre>	<b>Deleted:</b> <wsp:policy>¶</wsp:policy>

<wsam:anonymousresponses></wsam:anonymousresponses>		Deleted:	<wsp:policy>¶</wsp:policy>
<wsam:nonanonymousresponses 🏑=""></wsam:nonanonymousresponses>	[ ] ] ] ]	Deleted: wsp:0	Optional="true"
	12-		<u> </u>
		Deleted: wsp:0	Optional="true"
	Ì	Deleted: ¶	
		<th>p:Policy&gt;</th>	p:Policy>

Deleted: assertions
Deleted: assertion

Example 3-4. Subject requires WS-Addressing, supports anonymous and nonanonymous response EPRs

<wsp:policy></wsp:policy>	
<wsam:addressing></wsam:addressing>	
<wsam:anonymousresponses></wsam:anonymousresponses>	
<wsam:nonanonymousresponses></wsam:nonanonymousresponses>	
Example 3-5. Subject requires WS-Addressing and support of non-anonymous response EPRs	~~~

<wsp:policy></wsp:policy>		
<wsam:addressing></wsam:addressing>		Deleted: ¶
<wsam:nonanonymousresponses></wsam:nonanonymousresponses>		<w:< th=""></w:<>
	[	Deleted: ¶
		1</td

#### 3.1.5 Examples (Normal Form)

Example 3-6. Subject supports WS-Addressing, no statement on supported response EPRs

76	esponse Erins	1	
	<pre><wsp:policy>     <wsp:exactlyone></wsp:exactlyone></wsp:policy></pre>		¶
	<wsp:all></wsp:all> <wsp:all></wsp:all>	1	Deleted: ¶
1 I	<wsp:all> <wsam:addressing></wsam:addressing></wsp:all>	i i	<wsp:policy>¶</wsp:policy>
I		' ;	<wsp:exactlyone>¶</wsp:exactlyone>
	 		<wsp:all></wsp:all> ¶
			¶
E	Example 3-7. Subject requires WS-Addressing, no statement on supported		
	esponse EPRs		Deleted: (and optionally)
	<pre><wsp:policy></wsp:policy></pre>	į	Deleted: ¶
	<wsp:exactlyone></wsp:exactlyone>	- 1 /	<pre>vwsp:Policy&gt;¶</pre>
T	<wsp:all></wsp:all>	-1/j	<wsp:exactlyone>¶</wsp:exactlyone>
I	<wsam:addressing> </wsam:addressing>	' ;;;	
			<wsp:all></wsp:all> ¶
			¶
	Example 3-8. Subject supports WS-Addressing, explicitly and supports		
			Deleted: ¶
а	nonymous and non-anonymous response EPRs		<wsp:all>¶</wsp:all>
	<wsp:policy></wsp:policy>		<wsam:addressing>¶</wsam:addressing>
	<pre><wsp:exactlyone></wsp:exactlyone></pre>	- 14	<wsp:policy>¶</wsp:policy>
	<wsp:all></wsp:all> <wsp:all></wsp:all>		<wsp:exactlyone>¶</wsp:exactlyone>
1	<wsp:all> <wsam:addressing></wsam:addressing></wsp:all>	-1j	<wsp:all>¶</wsp:all>
	<pre><wsam:anonymousresponses></wsam:anonymousresponses></pre>	-1j	<wsp.all>1</wsp.all>
	<pre><wsam:nonanonymousresponses></wsam:nonanonymousresponses></pre>		<wsam:anonymousresponses></wsam:anonymousresponses> ¶
		i i	¶
1			

Deleted: requires explicit

</wsp:ExactlyOne>¶

Deleted: explicit

Deleted: ¶

<wsp:All/>¶

<wsp:ExactlyOne>¶

</wsp:ExactlyOne>¶

... [1]

<wsp:Policy>¶

</wsp:Policy>

<wsp:Policy>

</wsp:Policy>

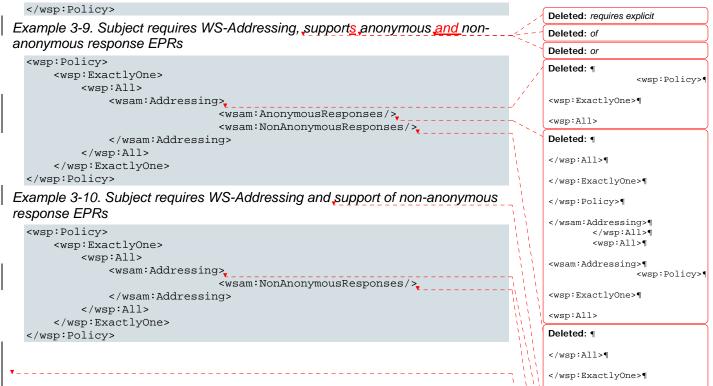
<wsp:ExactlyOne>

<wsp:Policy>¶

Deleted: of Deleted: or Deleted: ¶

Deleted: ¶

</wsp:All>



### </wsp:Policy>

#### Deleted: explicit

#### Deleted: ¶

<wsp:Policy>¶

<wsp:ExactlyOne>¶

#### <wsp:All>

Deleted: ¶

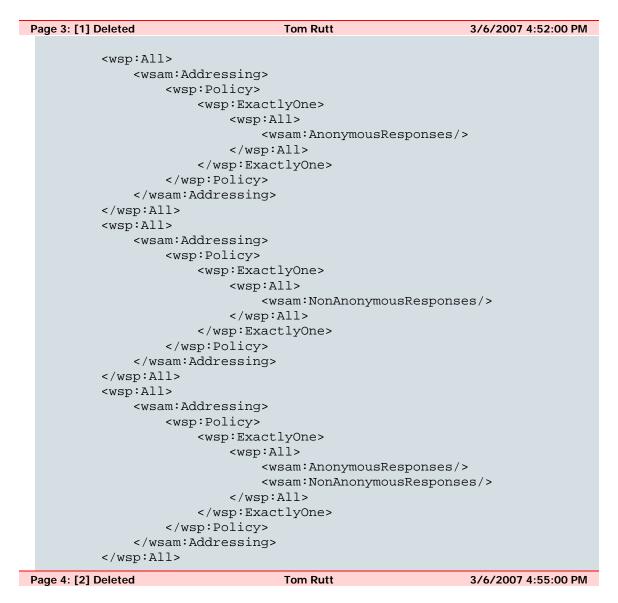
</wsp:All>¶

</wsp:ExactlyOne>¶

#### </wsp:Policy>

Deleted: 3.1.6 Finding Compatible Policies¶ When a client is looking for an endpoint with compatible policy, one common method used is to take the policy intersection between the policy which the client is looking for, and the policy asserted in the WSDL document; a non-empty intersection is sought. The policy used by the client must be written carefully to avoid unexpected results. This is most obvious when the client is not

looking for explicit support of a particular kind of response; f  $[\dots [2]]$ 



# 3.1.6 Finding Compatible Policies

When a client is looking for an endpoint with compatible policy, one common method used is to take the policy intersection between the policy which the client is looking for, and the policy asserted in the WSDL document; a non-empty intersection is sought. The policy used by the client must be written carefully to avoid unexpected results. This is most obvious when the client is not looking for explicit support of a particular kind of response; failing to take care could mean missing a compatible policy.

Consider the following example, where we have a client who does not care whether the endpoint explicitly supports anonymous responses, and a WSDL which states that the endpoint does explicitly support anonymous responses. Example 3-11. Client looking for an endpoint which supports Addressing, WSDL states explicit support for anonymous responses

<wsp:Policy>

<wsam:Addressing>

<wsp:Policy/>

</wsam:Addressing>

</wsp:Policy>

The client's policy (above) states the requirement for Addressing, but no requirement for explicit support of responses.

<wsp:Policy>

<wsam:Addressing>

<wsp:Policy>

<wsam:AnonymousResponses/>

</wsp:Policy>

</wsam:Addressing>

</wsp:Policy>

The policy attached to the endpoint in the WSDL (above) states explicit support for anonymous responses. The intersection of this policy with the client's policy will be empty, so the client will miss a compatible endpoint.

<wsp:Policy>

<wsam:Addressing>

<wsp:Policy>

<wsam:AnonymousResponses wsp:Optional="true"/>

</wsp:Policy>

</wsam:Addressing>

</wsp:Policy>

This is what the client's policy could be; by stating that the wsam:AnonymousResponses assertion is optional, there will be a nonempty intersection with endpoint policies that do and do not contain this assertion.

Now let us consider a variation on this same situation, where the WSDL marks its explicit support of anonymous responses as ignorable.

Example 3-12. Client looking for an endpoint which supports Addressing, WSDL states (ignorable) explicit support for anonymous responses

<wsp:Policy>

<wsam:Addressing>

<wsp:Policy/>

</wsam:Addressing>

</wsp:Policy>

The client's policy (above) states the requirement for Addressing, but no requirement for explicit support of responses.

<wsp:Policy>

<wsam:Addressing>

<wsp:Policy>

<wsam:AnonymousResponses wsp:Ignorable="true"/>

</wsp:Policy>

</wsam:Addressing>

</wsp:Policy>

The policy attached to the endpoint in the WSDL (above) states explicit support for anonymous responses, but marks that as an ignorable assertion. Now the result of the policy intersection with the client's policy will depend on whether the client is using lax or strict intersection. The strict intersection of this policy with the client's policy will still be empty. The lax intersection, on the other hand, will not be empty, so the client will find a compatible endpoint. These two examples show the use of wsp:Optional and wsp:Ignorable, and how they can be used to produce non-empty intersections between client and endpoint policies. For more detailed descriptions of the use of wsp:Optional, wsp:Ignorable, and strict and lax intersection, please refer to the WS-Policy Primer [*WS Policy 1.5 - Primer*].