

Proposal for Issue

http://www.w3.org/Bugs/Public/show_bug.cgi?id=6424

WS-Eventing is not defined using XML Infoset. But we believe Infoset has the following advantages:

- ◆ *More generic and flexible while maintaining the semantics of the specification*
- ◆ *Same specification can apply to XML with different surface forms*
- ◆ *More portable to various implementations, transport protocols and SOAP engines*
- ◆ *More friendly for SOAP/XML compression and aligned with the work of W3C EXI (Efficient XML Interchange) WG and other WGs at W3C.*

Proposed Changes

The proposed changes are described below. To highlight the differences, the added texts are shown in red color.

1. Add the following to **Section 2.4** as the last paragraph.

This specification defines a core set of properties for each operation, but it is also possible for other specifications to extend these and/or add other properties. The semantics and XML Infoset representation for any such extension properties will be described in their defining specifications.

2. Rewrite Section 3 as follows:

3. Subscription Messages

To create, renew, and delete subscriptions, subscribers send request messages to event sources and subscription managers.

When an event source accepts a request to create a subscription, it typically does so for a given amount of time, although an event source may accept an indefinite subscription with no time-based expiration. If the subscription manager accepts a renewal request, it updates that amount of time. During that time, notifications are delivered by the event source to the requested event sink. An event source may support filtering to limit notifications that are delivered to the event sink; if it does, and a subscribe request contains a filter, the event source sends only notifications that match the requested filter. The event source sends notifications until one of the following happens: the subscription manager accepts an unsubscribe request for the subscription; the subscription expires without being renewed; or the event source cancels the subscription prematurely. In this last case, the event source makes a best effort to indicate why the subscription ended.

In the absence of reliable messaging at the application layer (e.g. [\[WS-ReliableMessaging\]](#)), messages defined herein are delivered using the quality of service of the underlying transport(s) and on a best-effort basis at the application layer.

3.1 Infoset Model for Subscribe messages

Subscribe request and response messages are collections of abstract properties described below, where `wsa:[endpoint reference]` refers to the Information Model for Endpoint Reference defined in Section 2.1 of WS-Addressing 1.0 Core.

```
Subscribe Request has the following abstract properties with their
cardinalities specified in Section 3.2:
[event source]: wsa:[endpoint reference]
    Represents the endpoint reference of the Event Source.
[action]: wsa:[action]
    Identifies the semantics of this message.
[endpoint]: wsa:[endpoint reference]
    Represents the endpoint reference for the receiver of SubscriptionEnd
message
[requested expiry]: xsd:dateTime|xsd:duration
    Represents the expiry requested by the Subscriber.
[filter]: wse:[filter expression]
    Represents the filter.
[delivery]: wse:[delivery specification]
    Represents the specification of event delivery.

wse:[filter expression] has the following abstract properties:
[dialect]: xsd:anyURI
    Represents the dialect of filter expression.
[expression]: xsd:any
    Represents the filter expression.

wse:[delivery specification] has the following abstract properties:
[mode]: xsd:anyURI
    Represents the mode of event delivery.
[format]: xsd:anyURI
    Represents the format of event delivery.
[event sink]: wsa:[endpoint reference]
    Represents the endpoint reference of the Event Sink.
[extension]: xsd:any
    Represents the extension point.

Subscribe Response has the following abstract properties:
[subscription manager]: wsa:[endpoint reference]
    Represents the endpoint reference of the Subscription Manager.
[assigned expiry]: xsd:dateTime|xsd:duration
    Represents the subscription expiry assigned by the Subscription Manager.
[extension]: xsd:any
    Represents the extension point.
```

The XML Infoset representation and semantics of each property is described in Section 3.2.

Section 3.2 Subscribe XML Infoset Representation

To create a subscription, a subscriber sends a request message of the following form to an event source:

```
<s:Envelope ...>
  <s:Header ...>
    <wsa:To>"xs:anyURI"</wsa:To>?
    <wsa:Action>
      http://schemas.xmlsoap.org/ws/2004/08/eventing/Subscribe
    </wsa:Action>
    ...
  </s:Header>
  <s:Body ...>
    <wse:Subscribe ...>
      <wse:EndTo>endpoint-reference</wse:EndTo> ?
      <wse:Delivery Mode="xs:anyURI"? >xs:any</wse:Delivery>
      <wse:Expires>[xs:dateTime | xs:duration]</wse:Expires> ?
      <wse:Filter Dialect="xs:anyURI"? > xs:any </wse:Filter> ?
      ...
    </wse:Subscribe>
  </s:Body>
</s:Envelope>
```

The following describes additional, normative constraints on the outline listed above:

/s:Envelope/s:Header/wsa:To

The REQUIRED [event source] property provides the value of the WS-Addressing wsa:To element.

/s:Envelope/s:Header/wsa:Action

This REQUIRED element provides the value of the [action] property. If a SOAP Action URI is used in the binding for SOAP, the value indicated herein MUST be used for that URI.

/s:Envelope/s:Body/*/wse:EndTo

This OPTIONAL element provides the value of the [endto endpoint] property. Where to send a SubscriptionEnd message if the subscription is terminated unexpectedly. (See [Section 3.5 Subscription End](#).) If present, this element MUST be of type wsa:EndpointReferenceType. Default is not to send this message.

/s:Envelope/s:Body/*/wse:Delivery

This REQUIRED element provides the value of the [delivery specification] property. A delivery destination for notification messages, using some delivery mode. See [Section 1.2 Delivery Modes](#) for details.

/s:Envelope/s:Body/*/wse:Delivery/@Mode

This OPTIONAL attribute provides the value of the [mode] property. The delivery mode to be used for notification messages sent in relation to this subscription. Implied value is "http://schemas.xmlsoap.org/ws/2004/08/eventing/DeliveryModes/Push", which indicates that Push Mode delivery should be used. See [Section 1.2 Delivery Modes](#) for details.

If the event source does not support the requested delivery mode, the request MUST fail, and the event source MAY generate a wse:DeliveryModeRequestedUnavailable fault indicating that the requested delivery mode is not supported.

```
/s:Envelope/s:Body/*/wse:Delivery/@Mode="http://schemas.xmlsoap.org/ws/2004/08/eventing/DeliveryModes/Push"
```

Value of /s:Envelope/s:Body/*/wse:Delivery is a single element, wse:NotifyTo, that contains the endpoint reference to which notification messages should be sent.

```
/s:Envelope/s:Body/*/wse:Expires
```

This OPTIONAL element provides the value of the [requested expiry] property. Requested expiration time for the subscription. (No implied value.) The event source defines the actual expiration and is not constrained to use a time less or greater than the requested expiration. The expiration time may be a specific time or a duration from the subscription's creation time. Both specific times and durations are interpreted based on the event source's clock.

If this element does not appear, then the request is for a subscription that will not expire. That is, the subscriber is requesting the event source to create a subscription with an indefinite lifetime. If the event source grants such a subscription, it may be terminated by the subscriber using an Unsubscribe request, or it may be terminated by the event source at any time for reasons such as connection termination, resource constraints, or system shut-down.

If the expiration time is either a zero duration or a specific time that occurs in the past according to the event source, then the request MUST fail, and the event source MAY generate a wse:InvalidExpirationTime fault indicating that an invalid expiration time was requested.

Some event sources may not have a "wall time" clock available, and so are only able to accept durations as expirations. If such a source receives a Subscribe request containing a specific time expiration, then the request MAY fail; if so, the event source MAY generate a wse:UnsupportedExpirationType fault indicating that an unsupported expiration type was requested.

```
/s:Envelope/s:Body/*/wse:Filter
```

This OPTIONAL element provides the value of the [filter expression] property. A Boolean expression in some dialect, either as a string or as an XML fragment (see [/s:Envelope/s:Body/*/wse:Filter/@Dialect](#)). If the expression evaluates to false for a notification, the notification MUST NOT be sent to the event sink. Implied value is an expression that always returns true. If the event source does not support filtering, then a request that specifies a filter MUST fail, and the event source MAY generate a wse:FilteringNotSupported fault indicating that filtering is not supported.

If the event source supports filtering but cannot honor the requested filtering, the request MUST fail, and the event source MAY generate a wse:FilteringRequestedUnavailable fault indicating that the requested filter dialect is not supported.

`/s:Envelope/s:Body/*/wse:Filter/@Dialect`

This OPTIONAL attribute provides the value of the [dialect] property. Implied value is "http://www.w3.org/TR/1999/REC-xpath-19991116".

While an XPath predicate expression provides great flexibility and power, alternate filter dialects may be defined. For instance, a simpler, less powerful dialect might be defined for resource-constrained implementations, or a new dialect might be defined to support filtering based on data not included in the notification message itself. If desired, a filter dialect could allow the definition of a composite filter that contained multiple filters from other dialects.

`/s:Envelope/s:Body/*/wse:Filter/@Dialect="`
<http://www.w3.org/TR/1999/REC-xpath-19991116>`"`

Value of `/s:Envelope/s:Body/*/wse:Filter` is an XPath [[XPath 1.0](#)] predicate expression (PredicateExpr); the context of the expression is:

- Context Node: the SOAP Envelope containing the notification.
- Context Position: 1.
- Context Size: 1.
- Variable Bindings: None.
- Function Libraries: Core Function Library [[XPath 1.0](#)].
- Namespace Declarations: The [in-scope namespaces] property [[XML Infoset](#)] of `/s:Envelope/s:Body/*/wse:Filter`.

Other message information headers defined by WS-Addressing [[WS-Addressing](#)] MAY be included in the request and response messages, according to the usage and semantics defined in WS-Addressing.

Other components of the outline above are not further constrained by this specification.

If the event source accepts a request to create a subscription, it MUST reply with a response of the following form:

```
<s:Envelope ...>
  <s:Header ...>
    <wsa:Action>
      http://schemas.xmlsoap.org/ws/2004/08/eventing/SubscribeResponse
    </wsa:Action>
    ...
  </s:Header>
  <s:Body ...>
    <wse:SubscribeResponse ...>
      <wse:SubscriptionManager>
        wsa:EndpointReferenceType
      </wse:SubscriptionManager>
      <wse:Expires>[xs:dateTime | xs:duration]</wse:Expires>
      ...
    </wse:SubscribeResponse>
  </s:Body>
</s:Envelope>
```

The following describes additional, normative constraints on the outline listed above:

/s:Envelope/s:Header/wsa:RelatesTo

MUST be the value of the wsa:MessageID of the corresponding request.

/s:Envelope/s:Body/*/wse:SubscriptionManager

This REQUIRED element provides the value of the [subscription manager] property. The EPR of the subscription manager for this subscription.

In some cases, it is convenient for all EPRs issued by a single event source to address a single Web service and use a reference parameter to distinguish among the active subscriptions. For convenience in this common situation, this specification defines a global element, wse:Identifier of type xs:anyURI, that MAY be used as a distinguishing reference parameter if desired by the event source.

/s:Envelope/s:Body/*/wse:Expires

This OPTIONAL element provides the value of the [assigned expiry] property. The expiration time assigned by the event source. The expiration time MAY be either an absolute time or a duration but SHOULD be of the same type as the requested expiration (if any).

If this element does not appear, then the subscription will not expire. That is, the subscription has an indefinite lifetime. It may be terminated by the subscriber using an Unsubscribe request, or it may be terminated by the event source at any time for reasons such as connection termination, resource constraints, or system shut-down.

Other components of the outline above are not further constrained by this specification.

If the event source chooses not to accept a subscription, the request **MUST** fail, and the event source **MAY** generate a `wse:EventSourceUnableToProcess` fault indicating that the request was not accepted.

Table 4 lists another hypothetical request to create a subscription.

Table 4: Second hypothetical request to create a subscription

```
(01) <s12:Envelope
(02)   xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
(03)   xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
(04)   xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
(05)   xmlns:ew="http://www.example.com/warnings" >
(06)   <s12:Header>
(07)     <wsa:Action>
(08)       http://schemas.xmlsoap.org/ws/2004/08/eventing/Subscribe
(09)     </wsa:Action>
(10)     <wsa:MessageID>
(11)       uuid:e1886c5c-5e86-48d1-8c77-fc1c28d47180
(12)     </wsa:MessageID>
(13)     <wsa:ReplyTo>
(14)       <wsa:Address>http://www.example.com/MyEvEntsink</wsa:Address>
(15)     <wsa:ReferenceProperties>
(16)       <ew:MySubscription>2597</ew:MySubscription>
(17)     </wsa:ReferenceProperties>
(18)   </wsa:ReplyTo>
(19)   <wsa:To>http://www.example.org/oceanwatch/EventSource</wsa:To>
(20) </s12:Header>
(21) <s12:Body>
(22)   <wse:Subscribe>
(23)     <wse:EndTo>
(24)       <wsa:Address>
(25)         http://www.example.com/MyEventSink
(26)       </wsa:Address>
(27)       <wsa:ReferenceProperties>
(28)         <ew:MySubscription>2597</ew:MySubscription>
(29)       </wsa:ReferenceProperties>
(30)     </wse:EndTo>
(31)     <wse:Delivery>
(32)       <wse:NotifyTo>
(33)         <wsa:Address>
(34)           http://www.other.example.com/OnStormWarning
(35)         </wsa:Address>
(36)         <wsa:ReferenceProperties>
(37)           <ew:MySubscription>2597</ew:MySubscription>
(38)         </wsa:ReferenceProperties>
(39)       </wse:NotifyTo>
(40)     </wse:Delivery>
(41)     <wse:Expires>2004-06-26T21:07:00.000-08:00</wse:Expires>
(42)     <wse:Filter xmlns:ow="http://www.example.org/oceanwatch"
(43)       Dialect="http://www.example.org/topicFilter" >
(44)       weather.storms
```

```

(45)      </wse:Filter>
(46)      </wse:Subscribe>
(47)      </s12:Body>
(48) </s12:Envelope>

```

Like the request in [table 1](#) , Lines (07-09) of [table 4](#) indicate the message is a request to create a subscription. Line (19) indicates that it is sent to a hypothetical event source of ocean events.

Lines (13-18) indicate where to send the response to this request, Lines (23-30) indicate where to send a SubscriptionEnd message if necessary, and Lines (31-34) indicate how and where to send notifications.

Line (41) indicates the event sink would prefer to have the subscription expire on 26 June 2004 at 9:07 PM Pacific time.

Lines (42-45) indicate the event sink only wants weather reports where topic is storms, using a custom filter dialect.

Table 5 lists a hypothetical response to the request in [table 4](#) .

Table 5: Hypothetical response to second subscribe request

```

(01) <s12:Envelope
(02)   xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
(03)   xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
(04)   xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
(05)   xmlns:ew="http://www.example.com/warnings"
(06)   xmlns:ow="http://www.example.org/oceanwatch" >
(07)   <s12:Header>
(08)     <wsa:Action>
(09) http://schemas.xmlsoap.org/ws/2004/08/eventing/SubscribeResponse
(10)   </wsa:Action>
(11)     <wsa:RelatesTo>
(12)       uuid:e1886c5c-5e86-48d1-8c77-fc1c28d47180
(13)     </wsa:RelatesTo>
(14)     <wsa:To>http://www.example.com/MyEventSink</wsa:To>
(15)     <ew:MySubscription>2597</ew:MySubscription>
(16)   </s12:Header>
(17)   <s12:Body>
(18)     <wse:SubscribeResponse>
(19)       <wse:SubscriptionManager>
(20)         <wsa:Address>
(21)           http://www.example.org/oceanwatch/SubscriptionManager
(22)         </wsa:Address>
(23)         <wsa:ReferenceParameters>
(24)           <wse:Identifier>
(25)             uuid:22e8a584-0d18-4228-b2a8-3716fa2097fa
(26)           </wse:Identifier>
(27)         </wsa:ReferenceParameters>
(28)       </wse:SubscriptionManager>
(29)       <wse:Expires>2004-07-01T00:00:00.000-00:00</wse:Expires>

```



```
(30)    </wse:SubscribeResponse>
(31)    </s12:Body>
(32) </s12:Envelope>
```

Like the response in [table 2](#) , Lines (08-10) of [table 5](#) indicate this message is a response to a request to create a subscription, and Lines (11-13) indicate that it is a response to the request in [table 4](#) . Lines (14-15) indicate the response is sent to the event sink indicated in Lines (13-18) of [table 4](#) . Lines (19-28) provide the address of the subscription manager for this subscription; note that this particular response uses the global wse:Identifier element defined by this specification. Finally, Line (29) indicates the subscription will expire on 1 July 2004 unless renewed; there is no requirement that this time be necessarily longer or shorter than the requested expiration (Line (41) of [table 4](#)).