**WS-Enumeration Interoperability Scenario**

Date: December 8, 2010

Editor: Nathan Burkhart (nathan.burkhart@microsoft.com)

***Note:*** *This document is written under the assumption that it will be integrated into a larger document that covers interoperability scenarios for all specifications listed as Second Last Call Working Drafts here:* [*http://www.w3.org/2002/ws/ra/*](http://www.w3.org/2002/ws/ra/)*. The exact wording of many sections will need to change when merged with the rest of the interoperability scenarios.*

# Abstract

The following scenario is designed to provide a framework in which to test the interoperability of various WS-Enumeration implementations. Because this scenario and the tests defined within it will be used to judge which features of WS-Enumeration are implemented and which are not, the feature coverage is intended to be complete.

# Dependencies

## Scope

The following specification and technologies are in scope for this scenario:

* SOAP 1.1
* WS-Enumeration
* WSDL 1.1

## Namespaces

The following table defines the namespaces used in this document:

|  |  |  |
| --- | --- | --- |
| **Prefix** | **Namespace** | **Specification** |
| xsd | http://www.w3.org/2001/XMLSchema | XML Schema |
| wsdl | http://schemas.xmlsoap.org/wsdl/ | WSDL 1.1 |
| soap11 | http://schemas.xmlsoap.org/soap/envelope/ | SOAP 1.1 |
| wsoap11 | http://schemas.xmlsoap.org/wsdl/soap/ | WSDL 1.1 |
| wsa  | http://www.w3.org/2005/08/addressing | WS-Addressing 1.0 |
| wsen  | http://www.w3.org/2010/08/ws-enu | WS-Enumeration |

# Scenario Description

This scenario presupposes a cetacean tracking system in which a number of animals have been “tagged” with devices that track their location. A central database is used to maintain an inventory of all animals that have been tagged. External systems can consume the stored information by using WS-Enumeration to enumerate the tagged animals and retrieve the stored data that describes them.

A set of properties is maintained for each animal that is tagged. The following properties are stored:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Example** |
| ID | A GUID that uniquely identifies the animal | 13c76450-de3d-11df-85ca-0800200c9a66 |
| Name | The name given to the animal | Howard |
| Birthdate | The day on which the animal was born | 12/8/2006 |
| Gender | The gender of the animal | Male |
| Family | TheFamily of which the animal is a member | Eschrichtiidae |
| Genus | TheGenus of which the animal is a member | Eschrichtius |
| Species | TheSpecies of which the animal is a member | robustus |

# Tests

The following sub-sections describe tests designed to exercise all the mandatory and optional features of WS-Enumeration. Each of these subsections is organized into four parts:

* An overview that describes the purpose of the test and the salient features of the messages that are exchanged.
* A sequence diagram that illustrates the sequence of events in the test.
* A list of criteria used to judge the success of the test.
* A conformance section that enumerates the conditions under which conforming implementations are allowed to either not implement the test or fail one or more of the success criteria.

## Basic Test

This test verifies the ability to enumerate instances of a resource using the default behavior. The initial Enumerate request has the following salient features:

* NewContext element

The subsequent Enumerate requests have the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Basic Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of one or more valid Enumerate messages with EnumerationContext element by the data source
* Receipt of one or more valid EnumerateResponse messages with one item by the data consumer
* Receipt of a valid EnumerateResponse message with EndOfSequence element by the data consumer

### Conformance

A conforming data source MAY NOT be capable of implementing this test due to its inability to return items in response to an Enumerate request that also created a new enumeration.

A conforming data source MAY respond to the initial Enumerate request with a wsen:MaxElementsMustBeZero fault.

## New Empty Enumeration Test

This test verifies the ability to create a new enumeration without retrieving any of the data items in the first EnumerateResponse message. The initial Enumerate request has the following salient features:

* NewContext element
* MaxElements element equal to 0

The subsequent Enumerate requests have the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the New Empty Enumeration Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element and MaxElements equal to 0 by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with no items by the data consumer
* Receipt of one or more valid Enumerate messages with EnumerationContext element by the data source
* Receipt of one or more valid EnumerateResponse messages with one item by the data consumer
* Receipt of a valid EnumerateResponse message with EndOfSequence element by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:MaxElements element.

Because this test involves only operations and elements that are required to be supported by the data source, there are no allowable failure cases.

## Optimized Enumeration Test

This test verifies the ability to enumerate an entire set of instances of a resource with just one Enumerate message. The Enumerate request has the following salient features:

* NewContext element
* MaxElements element with large value

### Sequence

The following diagram illustrates the sequence of messages for the Optimized Enumeration Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element by the data source
* Receipt of a valid EnumerateResponse message with all Items and EndOfSequence by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:MaxElements element.

A conforming data source MAY NOT be capable of implementing this test due to its inability to return items in response to an Enumerate request that also created a new enumeration.

A conforming data source MAY respond to the initial Enumerate request with a wsen:MaxElementsMustBeZero fault.

## MaxCharacters Test

This test verifies the ability of a data consumer to limit the size of the items returned from an enumeration using the wsen:MaxCharacters element. The initial Enumerate request has the following salient features:

* NewContext element
* MaxElements element with a large value
* MaxCharacters element with a value slightly bigger than one returned item

The subsequent Enumerate requests have the following salient features:

* EnumerationContext element that was provided by the data source
* MaxElements element with a large value
* MaxCharacters element with a value slightly bigger than one returned item

### Sequence

The following diagram illustrates the sequence of messages for the MaxCharacters Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element, MaxElements element and MaxCharacters element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of one or more valid Enumerate messages with EnumerationContext element, MaxElements element and MaxCharacters element by the data source
* Receipt of one or more valid EnumerateResponse messages with one item by the data consumer
* Receipt of a valid EnumerateResponse message with EndOfSequence element by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:MaxElements element or the optional wsen:MaxCharacters element.

Because this test involves only operations and elements that are required to be supported by the data source, there are no allowable failure cases.

## MaxTime Test

This test verifies the ability of a data consumer to limit the amount of time it takes for the data source to assemble an EnumerateResponse using the wsen:MaxTime element. The initial Enumerate request has the following salient features:

* NewContext element

The second Enumerate request has the following salient features:

* EnumerationContext element that was provided by the data source
* MaxTime element with a very low value

The subsequent Enumerate requests have the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the MaxTime Test. Note that the data source is not able to assemble the second Enumerate response before the duration specified in the MaxTime element has passed.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of one valid Enumerate message with EnumerationContext element and MaxTime element by the data source
* Receipt of a wsen:TimedOut fault by the data consumer
* Receipt of one or more valid Enumerate messages with EnumerationContext element by the data source
* Receipt of one or more valid EnumerateResponse messages with one item by the data consumer
* Receipt of a valid EnumerateResponse message with EndOfSequence element by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:MaxTime element.

Because this test involves only operations and elements that are required to be supported by the data source, there are no allowable failure cases.

## Duration Expiration Test

This test verifies the correct implementation of the expiration feature on the data source. The initial Enumerate message has the following salient features:

* NewContext element
* Expires element with a short expiration time as xs:duration

The second Enumerate message has the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Duration Expiration Test. Note that the data source waits until the expiration time has passed before sending the second request.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element and Expires element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of valid Enumerate message with EnumerationContext element by the data source
* Receipt of a wsen:InvalidEnumerationContext fault by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:Expires element with a value of type xs:duration.

A conforming data source MAY NOT be capable of implementing this test due to its inability to grant an expiration time that matches the specified value in the Enumerate request.

A conforming data source MAY respond to the initial Enumerate request with a wsen:UnsupportedExpirationValue fault.

## Specific Time Expiration Test

This test verifies the correct implementation of the expiration feature on the data source. The initial Enumerate message has the following salient features:

* NewContext element
* Expires element with a short expiration time as xs:dateTime

The second Enumerate message has the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Specific Time Expiration Test. Note that the data source waits until the expiration time has passed before sending the second request.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element and Expires element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of valid Enumerate message with EnumerationContext element by the data source
* Receipt of a wsen:InvalidEnumerationContext fault by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:Expires element with a value of type xs:dateTime.

A conforming data source MAY NOT be capable of implementing this test due to its inability to grant an expiration time that matches the specified value in the Enumerate request, or to its inability to support specific expiration time values.

A conforming data source MAY respond to the initial Enumerate request with a wsen:UnsupportedExpirationValue fault or a wsen:UnsupportedExpirationType fault.

## Best Effort Expiration Test

This test verifies the correct implementation of the “best effort” expiration feature on the data source. The initial Enumerate message has the following salient features:

* NewContext element
* Expires element with a short expiration time as xs:duration with @BestEffort=”true”

The second Enumerate message has the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Best Effort Expiration Test. Note that the data source waits until the expiration time has passed before sending the second request.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element and Expires element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of valid Enumerate message with EnumerationContext element by the data source
* Receipt of a wsen:InvalidEnumerationContext fault by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:Expires element with a value of type xs:duration or with the @BestEffort attribute.

Because this test involves only operations and elements that are required to be supported by the data source, there are no allowable failure cases.

## GetStatus Test

This test verifies the ability of a data consumer to get the status of an existing enumeration. The initial Enumerate request has the following salient features:

* NewContext element

The GetStatus message has the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the GetStatus Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element by the data consumer
* Receipt of valid GetStatus message with EnumerationContext element by the data source
* Receipt of valid GetStatusResponse message by the data source

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the GetStatus operation.

Because this test involves only operations and elements that are required to be supported by the data source, there are no allowable failure cases.

## Renew Test

This test verifies the ability of a data consumer to renew an existing enumeration. The initial Enumerate request has the following salient features:

* NewContext element

The Renew message has the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Renew Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element by the data consumer
* Receipt of valid Renew message with EnumerationContext element by the data source
* Receipt of valid RenewResponse message by the data source

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the Renew operation.

A conforming data source MAY choose not to renew the enumeration and instead respond to the Renew request with a SOAP 1.1 Server fault or a SOAP 1.2 Receiver fault.

## Release Test

This test verifies the ability of a data consumer to release an existing enumeration before its data items have all been retrieved. The initial Enumerate request has the following salient features:

* NewContext element

The Release message has the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Release Test.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element by the data consumer
* Receipt of valid Release message with EnumerationContext element by the data source
* Receipt of valid ReleaseResponse message by the data source

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the Release operation.

Because this test involves only operations and elements that are required to be supported by the data source, there are no allowable failure cases.

## EnumerationEnd Test

This test verifies the ability of a data source to send a notification to a data consumer if it terminates the enumeration unexpectedly. The initial Enumerate request has the following salient features:

* NewContext element
* EndTo element containing an EPR for the data consumer

### Sequence

The following diagram illustrates the sequence of messages for the EnumerationEnd Test. Note that the data source is shut down in a controlled manner before sending the EnumerationEnd message.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element by the data consumer
* Receipt of valid EnumerationEnd message by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:EndTo element or the EnumerationEndPortType portType.

A conforming data source MAY NOT be capable of implementing this test due to its inability to support the use of the EndTo EPR.

A conforming data source MAY respond to the initial Enumerate request with a wsen:EndToNotSupported fault.

A conforming data source MAY respond to the initial Enumerate request with a wsen:UnusableEPR fault if it checks the validity of the EndTo EPR and detects a problem.

## Filter Test – XPath 1.0

This test verifies the ability of the data source to correctly implement XPath 1.0 filters. The initial Enumerate request has the following salient features:

* NewContext element
* Filter element with @Dialect equal to “http://www.w3.org/2010/08/ws-enu/Dialects/XPath10” (actual filter expression TBD)

The subsequent Enumerate requests have the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Filter Test – XPath 1.0.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element and Filter element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of one or more valid Enumerate messages with EnumerationContext element by the data source
* Receipt of one or more valid EnumerateResponse messages with one item by the data consumer
* Receipt of a valid EnumerateResponse message with EndOfSequence element by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:Filter element or the @Dialect “http://www.w3.org/2010/08/ws-enu/Dialects/XPath10”.

A conforming data source MAY NOT be capable of implementing this test due to its inability to support filtering, to process the filter dialect “http://www.w3.org/2010/08/ws-enu/Dialects/XPath10”, or to process the filter content.

A conforming data source MAY respond to the initial Enumerate request with a wsen:FilteringNotSupported fault, a wsen:FilterDialectRequestedUnavailable fault, or a wsen:CannotProcessFilter fault..

A conforming data source MAY respond to the initial Enumerate request with a wsen:EmptyFilter fault if it detects that the filter will never evaluate to true for the lifetime of the enumeration.

## Filter Test – XPath 2.0

This test verifies the ability of the data source to correctly implement XPath 2.0 filters. The initial Enumerate request has the following salient features:

* NewContext element
* Filter element with @Dialect equal to “http://www.w3.org/2010/08/ws-enu/Dialects/XPath20” (actual filter expression TBD)

The subsequent Enumerate requests have the following salient features:

* EnumerationContext element that was provided by the data source

### Sequence

The following diagram illustrates the sequence of messages for the Filter Test – XPath 2.0.

**SEQUENCE DIAGRAM TBD**

### Success Criteria

* Receipt of valid Enumerate message with NewContext element and Filter element by the data source
* Receipt of a valid EnumerateResponse message with EnumerationContext element and with one item by the data consumer
* Receipt of one or more valid Enumerate messages with EnumerationContext element by the data source
* Receipt of one or more valid EnumerateResponse messages with one item by the data consumer
* Receipt of a valid EnumerateResponse message with EndOfSequence element by the data consumer

### Conformance

A conforming data consumer MAY NOT be capable of implementing this test due to its inability to support the optional wsen:Filter element or the @Dialect “http://www.w3.org/2010/08/ws-enu/Dialects/XPath20”.

A conforming data source MAY NOT be capable of implementing this test due to its inability to support filtering, to process the filter dialect “http://www.w3.org/2010/08/ws-enu/Dialects/XPath20”, or to process the filter content.

A conforming data source MAY respond to the initial Enumerate request with a wsen:FilteringNotSupported fault, a wsen:FilterDialectRequestedUnavailable fault, or a wsen:CannotProcessFilter fault..

A conforming data source MAY respond to the initial Enumerate request with a wsen:EmptyFilter fault if it detects that the filter will never evaluate to true for the lifetime of the enumeration.

# WSDL

TBD

# Schemas

TBD