

**Schema Release**  
**AseXML Schema Working Group**  
**Release r32**

**Draft Release Date: 11/02/2013**

**Final Release Date: 08/03/2013**

## Document History

Version	Date	Authors	Comments
0.1	20/2/13	Andrew Screen	Initial draft
1.0	5/3/13	Andrew Screen	Updated with fixes for enumeration spelling. Sample files added

## Contents

1. Introduction .....	4
2. Change Requests .....	4
3. Impact Summary .....	5
4. File Change Summary.....	6
4.1. aseXML .....	6
4.2. Changes .....	6
4.2.1 Schema change description.....	7
5. Schema Manifest .....	13
6. Schema Test.....	14
6.1. Test Platforms .....	14
6.2. Test.....	14
6.2.1 Test Platforms.....	14
6.2.2 Test Cases .....	14
6.2.3 Test Process .....	14
6.2.4 Test Results .....	15
6.2.5 Character Classification.....	15
7. ASWG Endorsement.....	16
8. AEMO Approval .....	16

## 1. Introduction

Version r32 of the aseXML schema has been developed from r31. This schema release is presented to aseXML Subscribers and Industry Participants for review, and to AEMO for approval, in accordance with the ASWG Terms of Reference.

## 2. Change Requests

The following ASWG Change Requests have been included in this schema release:

CR#	Description
51 v1.1	Provide a new report response format to provide feedback to participants about the current status of their processing by MSATS systems.
52 v1.3	<ul style="list-style-type: none"> <li>• Modifications to the Customer Details Request (Auto/Man) See: QC776- SD - Customer and Site Details - Pack 2 v2.00 -ws.docx, Solution 776-1-N7</li> <li>• Sensitive Load Reconciliation Only: See QC776- SD - Customer and Site Details - Pack 1 v2.00.docx, Solution 776-5-N2.</li> </ul>

These Change Requests are available from the ASWG or the aseXML website ([http://au.groups.yahoo.com/group/aseXMLTech/files/aseXML\\_Schemas/Schema\\_Change\\_Requests/](http://au.groups.yahoo.com/group/aseXMLTech/files/aseXML_Schemas/Schema_Change_Requests/)) for review by all impacted parties.

### 3. Impact Summary

This table identifies the files, transactions and versioned types that are changed in this schema, where:

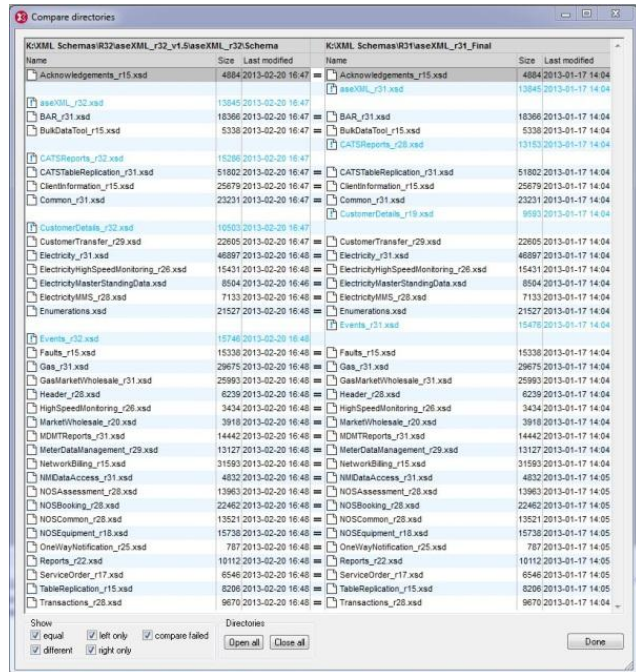
- Modified types - is a full list of types changed in this schema
- Derived types – is a list of any types that are derived from a modified type, and are therefore also modified by default
- Versioned types affected – is a list of all versioned types that will need to have the version attribute updated to use this schema
- Transactions potentially affected – is a list of all transactions that contain a modified type, either directly or via a type substitution
- Schema files affected – is a list of schema files that have been changed in some way for this schema.

Modified types	Derived types	Versioned types affected	Transactions potentially affected	Schema files affected
RequestReason		CustomerDetailsRequest	CustomerDetailsRequest	CustomerDetails_r32.xsd
MovementType	CustomerData	CustomerDetailsNotification	CustomerDetailsNotification	
			ReportResponse	CATSReports_r32.xsd

## 4. File Change Summary

The following file changes are implemented to create the r32 schema version as shown in the comparison of directories in the picture below taken from XMLSpy:

- asexml
- CATSReports
- CustomerDetails
- Events



### 4.1. aseXML

- Changed aseXML namespace and file references to reflect r32 schema

### 4.2. Changes

Chg #	Item #	Description of change	Filename
CR 51 & 52	1	Updated schema namespace to urn:aseXML:r32 for xmlns, TargetNamespace and schema location. File version changed from r31 to r32.	asexml_r32.xsd
	2	Updated Event schema to Events_r32 in the aseXML_r32.	
CR 51	3	Updated CATSReports schema to CATSReports_r32;	asexml_r32.xsd
CR 52	4	Updated CustomerDetails schema to CustomerDetails_r32	asexml_r32.xsd
CR 51 & 52	5	Added event (release Identifier) for R32	Events_r32.xsd
CR 51	6	Added "CATSParticipantSystemStatusReportFormat" types; File version changed from r30 to r32.	CATSReports_r32
CR 52	7	Amend simple type RequestReason.	CustomerDetails_r32.xsd
CR 52	8	Amend simple type MovementType	CustomerDetails_r32.xsd
CR 52	9	Amend complex type CustomerData	CustomerDetails_r32.xsd
CR 52	10	Change the version attribute of versioned types impacted	CustomerDetails_r32.xsd

		by change to RequestReason and MovementType.	
--	--	--	--

Table 4-1 Change Log

## 4.2.1 Schema change description

### 4.2.1.1 asexml\_r32.xsd

Normal changes associated with updating the schema to reflect the namespaces and filename changes in the schema.

Updated:

```
<xsd:schema xmlns="urn:aseXML:r31"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  targetNamespace="urn:aseXML:r31" version="r31"
  xsi:schemaLocation="urn:aseXML:r31 aseXML_r31.xsd">
```

To

```
<xsd:schema xmlns="urn:aseXML:r32"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  targetNamespace="urn:aseXML:r32" version="r32"
  xsi:schemaLocation="urn:aseXML:r32 aseXML_r32.xsd">
```

- Modified include file references for modified files to be:

- Events\_r32.xsd
- CATSReports\_r32.xsd
- CustomerDetails\_r32.xsd

### 4.2.1.2 event\_r32.xsd

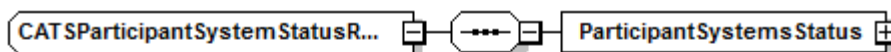
Normal changes associated with updating the schema to reflect the new release identifier in the schema.

Added:

```
<xsd:simpleType name="r32">
  <xsd:annotation>
    <xsd:documentation>Purpose - Release r32 identifier.</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="ReleasIdentifier">
    <xsd:enumeration value="r32"/>
  </xsd:restriction>
</xsd:simpleType>
```

### 4.2.1.3 CATSReports\_r32.xsd

- Add new complex type “CATSParticipantSystemStatusReportFormat” derived from “BaseReportFormat” with mandatory element “ParticipantSystemsStatus” of type “ParticipantSystemsStatus”.

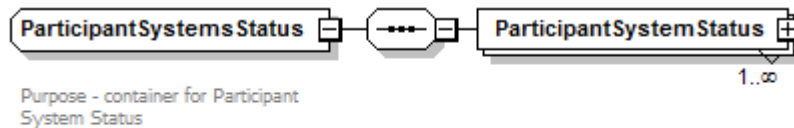


Purpose - Format for a CATS Participant  
System Status Report  
Report Name - ParticipantSystemStatus

Generated by XMLSpy

www.altova.com

- Add new complex type “ParticipantSystemsStatus” with mandatory element “ParticipantSystemStatus” of type “ParticipantSystemStatus” which can appear more than once.



Generated by XMLSpy

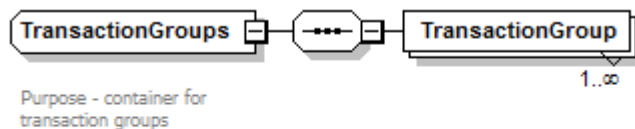
www.altova.com

```

<xsd:complexType name="ParticipantSystemsStatus">
  <xsd:annotation>
    <xsd:documentation>
      Purpose - container for Participant System Status
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="ParticipantSystemStatus" type="ParticipantSystemStatus"
      maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

```

- Add new complex type “TransactionGroups” with mandatory element “TransactionGroup” of type “TransactionGroup” which can appear more than once.



Generated by XMLSpy

www.altova.com

```

<xsd:complexType name="TransactionGroups">
  <xsd:annotation>
    <xsd:documentation>
      Purpose - container for transaction groups
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="TransactionGroup" type="TransactionGroup"
      maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

```

- Add new complex type “ParticipantSystemStatus” with
  - Mandatory element “System” of type “xsd:string” which can appear only once.
  - Mandatory element “ModeType” of type “xsd:string” which can appear only once.
  - Optional element “TransactionGroups” of type “TransactionGroups” which can appear only once.
  - Optional element “Box” of type “xsd:string” which can appear only once.
  - Mandatory element “Status” of type “xsd:string” which can appear only once.
  - Optional element “Description” of type “xsd:string” which can appear only once.
  - Optional element “Reason” of type “xsd:string” which can appear only once.
  - Optional element “Resolution” of type “xsd:string” which can appear only once.
  - Mandatory element “StartTime” of type “xsd:dateTime” which can appear only once.
  - Mandatory element “HeartbeatTime” of type “xsd:dateTime” which can appear only once.



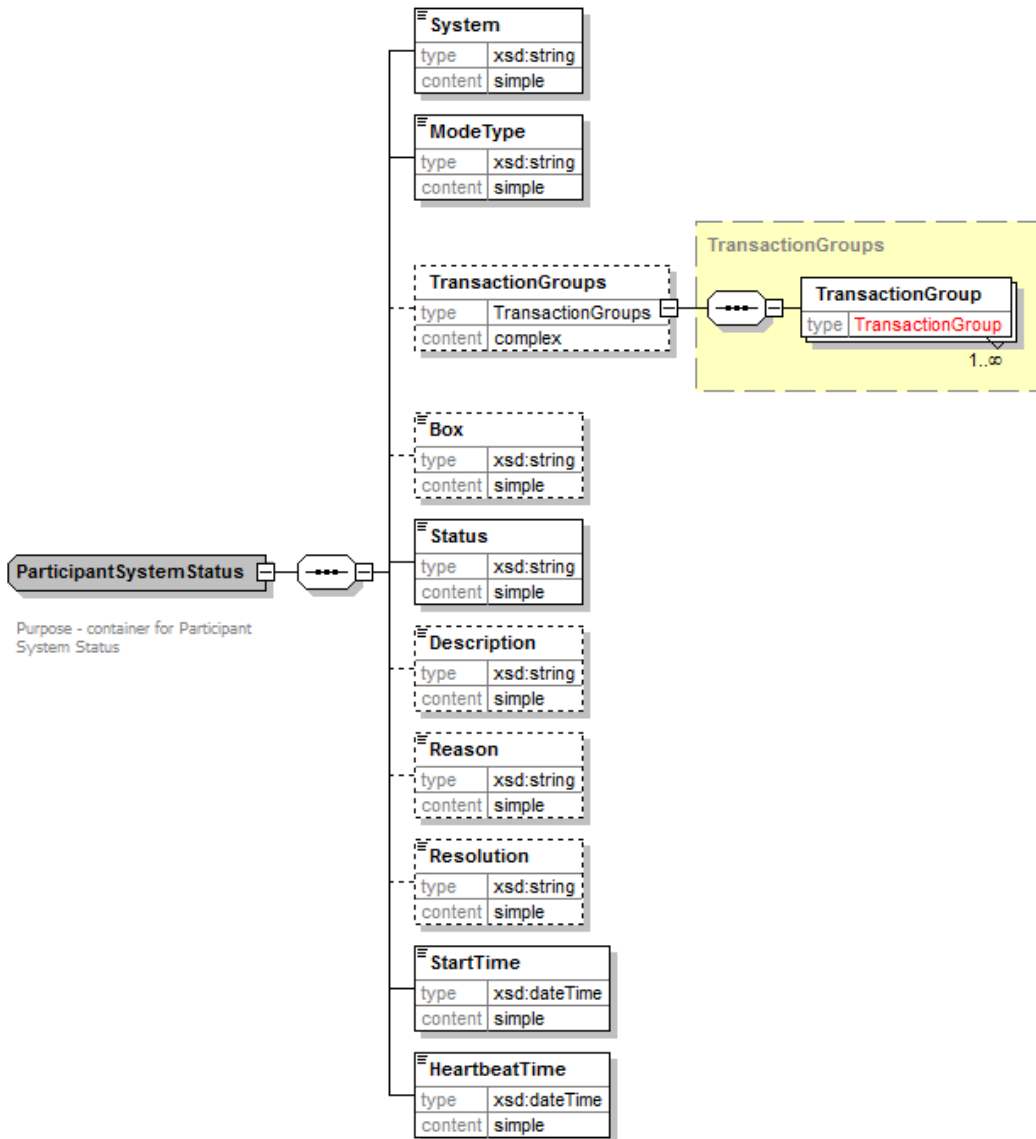


Fig 4.2.1.3.1 Model View for ParticipantSystemStatus

Comment		Complex Types for Participant System Status Report	
xsd:complexType name=TransactionGroups			
xsd:complexType			
name		ParticipantSystemStatus	
xsd:annotation			
xsd:sequence			
maxOccurs		1	
xsd:element name=System type=xsd:string			
xsd:element name=ModeType type=xsd:string			
xsd:element			
name		TransactionGroups	
type		TransactionGroups	
minOccurs		0	
maxOccurs		1	
xsd:element			
name		Box	
type		xsd:string	
minOccurs		0	
maxOccurs		1	
xsd:element			
name		Status	
type		xsd:string	
maxOccurs		1	
minOccurs		1	
xsd:element			
name		Description	
type		xsd:string	
minOccurs		0	
maxOccurs		1	
xsd:element			
name		Reason	
type		xsd:string	
minOccurs		0	
maxOccurs		1	
xsd:element			
name		Resolution	
type		xsd:string	
minOccurs		0	
maxOccurs		1	
xsd:element			
name		StartTime	
type		xsd:dateTime	
maxOccurs		1	
minOccurs		1	
xsd:element			
name		HeartbeatTime	
type		xsd:dateTime	
minOccurs		1	
maxOccurs		1	

Fig 4.2.1.3.2 Grid View for ParticipantSystemStatus

**XML Fragment for ParticipantSystemStatus Complex Type**

```
<!-- Complex Types for Participant System Status Report-->
<xsd:complexType name="TransactionGroups">
```

```

<xsd:annotation>
  <xsd:documentation>
    Purpose - container for transaction groups
  </xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element name="TransactionGroup" type="TransactionGroup" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ParticipantSystemStatus">
  <xsd:annotation>
    <xsd:documentation>
      Purpose - container for Participant System Status
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence maxOccurs="1">
    <xsd:element name="System" type="xsd:string"/>
    <xsd:element name="ModeType" type="xsd:string"/>
    <xsd:element name="TransactionGroups" type="TransactionGroups" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="Box" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="Status" type="xsd:string" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="Description" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="Reason" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="Resolution" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="StartTime" type="xsd:dateTime" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="HeartbeatTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

```

#### 4.2.1.4 CustomerDetails\_r32.xsd

##### Sensitive Load changes

- Add new enumeration values to simple type *MovementType*:

<i>MovementType</i> Code	Description
'RecCustConf'	Reconciliation - Confirmed with Customer. Means that the Retailer has confirmed with the Customer that the <i>SensitiveLoad</i> value is correct.
'RecPerRetailer'	Reconciliation - As per Retailer. Means that the Retailer reasonably believes that the <i>SensitiveLoad</i> value provided is correct.
'RecPerDNSP'	Reconciliation - As per DNSP. Means that the Retailer agrees that the <i>SensitiveLoad</i> value should be 'None', as inferred by the DNSP.

Current *MovementType* values are:

- Update
- Reconciliation
- Site Vacant

##### Changes for Customer Request Details Transaction:

- Add a new value 'Rec - confirm no SensitiveLoad' to the simple type RequestReason for the CustomerDetailsRequest as part of sensitive load reconciliation changes

"Rec - confirm no SensitiveLoad" means DNSP believes a NMI has a SensitiveLoad value other than 'None' and was not included in the Reconciliation Process.

Current Reason values as defined by simple type RequestReason are:

- Returned Mail
- Missing Customer Details
- Confirm Life Support
- Other

Add the following new enumeration values to the simple type RequestReason as part of the changes for customer details requests:

- Transfer complete, no CDN received
- New Connection, no CDN received
- No response to rejected CDN
- Data Quality Issue

Current simple type RequestReason enumeration values are:

- Returned Mail
- Missing Customer Details
- Confirm Life Support
- Other

### Add new enumeration values to simple type MovementType.

Amend simple type MovementType as follows:

```
<xsd:simpleType name="MovementType">
  <xsd:annotation>
    <xsd:documentation>
      Purpose - Define the fixed Special Condition Code used in customer details notification where:
      RecCustConf: Reconciliation - Confirmed with Customer
      RecPerRetailer: Reconciliation - As per Retailer
      RecPerDNSP: Reconciliation - As per DNSP.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Update"/>
    <xsd:enumeration value="Reconciliation"/>
    <xsd:enumeration value="Site Vacant"/>
    <xsd:enumeration value="RecCustConf"/>
    <xsd:enumeration value="RecPerRetailer"/>
    <xsd:enumeration value="RecPerDNSP"/>
  </xsd:restriction>
</xsd:simpleType>
```

Change the version attribute of CustomerDetailsNotification type impacted by changing of MovementType :

Change from : <xsd:attribute name="version" type="r18" use="required"/>

Change to : <xsd:attribute name="version" type="r32" use="optional" default="r32"/>

Amend simple type RequestReason as follows:

```
<xsd:simpleType name="RequestReason">
  <xsd:annotation>
    <xsd:documentation>
      Purpose - Define the fixed Reasons used to request customer details
    </xsd:documentation>
  </xsd:annotation>
</xsd:simpleType>
```

```

</xsd:annotation>
<xsd:restriction base="xsd:string">
  <xsd:enumeration value="Returned Mail"/>
  <xsd:enumeration value="Missing Customer Details"/>
  <xsd:enumeration value="Confirm Life Support"/>
  <xsd:enumeration value="Other"/>
  <xsd:enumeration value="Transfer complete, no CDN received"/>
  <xsd:enumeration value="New Connection, no CDN received"/>
  <xsd:enumeration value="No response to rejected CDN"/>
  <xsd:enumeration value="Data Quality Issue"/>
  <xsd:enumeration value="Rec - confirm no SensitiveLoad"/>
</xsd:restriction>
</xsd:simpleType>

```

Change the version attribute of CustomerDetailsRequest type impacted by changing of RequestReason:

Change from : <xsd:attribute name="version" type="r17" use="required"/>

Change to : <xsd:attribute name="version" type="r32" use="optional" default="r32"/>

## 5. Schema Manifest

The table below shows the schema files included in this release. Files that have been added, removed or modified for this release are marked.

File	Modified
Acknowledgements_r15.xsd	N
aseXML_r32.xsd	Y
BAR_r31.xsd	N
BulkDataTool_r15.xsd	N
CATSReports_r32.xsd	Y
CATSTableReplication_r31.xsd	N
ClientInformation_r15.xsd	N
Common_r31.xsd	N
CustomerDetails_r32.xsd	Y
CustomerTransfer_r29.xsd	N
ElectricityHighSpeedMonitoring_r31.xsd	N
ElectricityMasterStandingData.xsd	N
ElectricityMMS_r28.xsd	N
Electricity_r25.xsd	N
Enumerations.xsd	N
Events_r32.xsd	Y
Faults_r15.xsd	N
GasMarketWholesale_r31.xsd	N
Gas_r29.xsd	N
Header_r28.xsd	N
HighSpeedMonitoring_r26.xsd	N
MarketWholesale_r20.xsd	N
MDMTRReports_r31.xsd	N
MeterDataManagement_r29.xsd	N
NetworkBilling_r15.xsd	N
NMIDataAccess_r31.xsd	N
NOSAssessment_r28.xsd	N
NOSBooking_r28.xsd	N
NOSCommon_r28.xsd	N
NOSEquipment_r18.xsd	N
OneWayNotification_r25.xsd	N
Reports_r22.xsd	N
ServiceOrder_r17.xsd	N

File	Modified
TableReplication_r15.xsd	N
Transactions_r28.xsd	N

## 6. Schema Test

### 6.1. Test Platforms

The new schema has been tested using the following platforms:

- XMLSpy 2010
- MSXML4
- MSXML6
- Xerces 2.2.1 and 2.9.1

### 6.2. Test

The ASWG ensures that all recommended parsers on relevant platforms can successfully validate the proposed schema.

#### 6.2.1 Test Platforms

Supplied samples have been tested using the following parsers:

- MSXML 6.0
- MSXML 4.0
- Xerces 2.2.1
- Xerces 2.9.1
- XMLSpy 2010

#### 6.2.2 Test Cases

New Specific Test Cases:

- NEMMSATS\_ParticipantSystemStatus\_ReportRequest\_01.xml
- NEMMSATS\_ParticipantSystemStatus\_ReportResponse\_01.xml
- NEMMSATS\_ParticipantSystemStatus\_ReportResponse\_02.xml
- NEMMSATS\_ParticipantSystemStatus\_ReportResponse\_03.xml
- NEMMSATS\_CustomerDetailsNotification\_MovementType\_r32.xml
- NEMMSATS\_CustomerDetailsRequest\_RequestReason\_r32.xml

#### 6.2.3 Test Process

1. Obtain a copy of the 5 existing regression test suite XML files
  - a. nem\_samples\_r31 zip.zip
  - b. nem\_wholesale\_samples\_r31 zip.zip
  - c. nemb2b\_samples\_r31 zip.zip
  - d. sawa\_samples\_r31 zip.zip
  - e. vicgas\_samples\_r31 zip.zip
2. Obtain a copy of the new R32 test suite XML files
3. Unzip all test XML files a folder
4. Replace “xmlns:ase=urn:aseXML:r31” with “xmlns:ase=urn:aseXML:r32”
5. Replace “xsi:schemaLocation=urn:aseXML:r31” with [http://www.nemmco.com.au/aseXML/schemas/r31/aseXML\\_r31.xsd](http://www.nemmco.com.au/aseXML/schemas/r31/aseXML_r31.xsd) with

- “xsi:schemaLocation=”urn:aseXML:r32  
[http://www.nemmco.com.au/aseXML/schemas/r32/aseXML\\_r32.xsd](http://www.nemmco.com.au/aseXML/schemas/r32/aseXML_r32.xsd)”
6. Check files with XMLSpy bypassing the url referencing and point specifically to the schema using project properties.
  7. Check every single test XML file individually to detect variances in the samples.
  8. Updated samples so that they validate
  9. Copy then update sample files by replace “xsi:schemaLocation=”urn:aseXML:r32  
[http://www.nemmco.com.au/aseXML/schemas/r32/aseXML\\_r32.xsd](http://www.nemmco.com.au/aseXML/schemas/r32/aseXML_r32.xsd)” with  
 “xsi:schemaLocation=”urn:aseXML:r32 C:/samples\_a/aseXML\_r32.xsd”
  10. Check every single test XML file individually to detect variances in the above xsi:schemaLocation approach, e.g. hardcoded “O:/<filename>” instances and other occurrences not picked up by the standard search/replace above need to be manually fixed. It would be good if a single search/replace could be used for this step, and the test XML files had consistent headers.
  11. Run the test process using the 4 supported XML Toolkits.
  12. Check output log for any successful parse results, as well as expected or unexpected errors.

## 6.2.4 Test Results

All OK on all Test Platforms, see section 6.2.1

- To make the review easier sample files that had invalid in the name but were just responses to invalid were renamed. Renaming changed Invalid to response.
- All test XML files intended to fail, typically with “Invalid” as part of the file name, failed as expected.
- All other test files passed with no parse error.
- Comparison of the enumerations in the schema to what is in the change notes was made to confirm the



enumeration  
comparison v1\_6.xlsx

changes. See:

## 6.2.5 Character Classification

Pattern restriction is enforced by regular expressions in some places in aseXML schema. That makes knowing the precise set of classification of characters important. It is particularly important in the testing process. In some quick tests using simple Pattern class in JAVA – not fully fledged JAVA parsers – some differences vis-à-vis XML Spy were reported. These tests were performed to validate the data stored in database against aseXML type definitions. Some characters that were treated differently between JAVA Pattern class and XML Spy were \$ + |. XML Spy accepted them as punctuation characters but JAVA Pattern class rejected. Note, these three characters are only a few examples of difference, not an exhaustive list. As explained below, further investigations revealed that XML Spy is correct as per the XML standard.

The XML standards depend upon Unicode specifications for the purpose of this classification. The complete list of classification of Unicode characters in various classes can be found at <http://www.unicode.org/>.

To download the classification for any particular version of classification, start from directory listing at <http://www.unicode.org/Public> and traverse down the tree of the concerned version to download the zip (usually named ucd.zip) which contains all the documents for that version. The zip for version 6.2.0 resides at <http://www.unicode.org/Public/6.2.0/ucd/>. The document in this zip, usually named UnicodeData.txt, contains entire classification of all Unicode characters, having a line per Unicode character, with semi-colon delimited fields in each line. Explanation of fields can be found in the documentation in the zip (UCD.html). To view this UnicodeData.txt document conveniently, start Excel and open the document. While opening the document Excel will ask for information about the file contents. In response specify the document to be semicolon delimited with each column of the document being text. The column C of the Excel spreadsheet specifies the character classification. Note, while opening the file in Excel if each column is not specified to be text then Excel may format some information incorrectly.

The classification of the extended ASCII characters for version 6.2.0 of Unicode is provided here in an Excel spreadsheet :



UnicodeData\_6.2.0\_  
ASCII\_Character\_cla:

The classifications starting with L signify letters, starting with N signify digits, starting with P signify punctuations, starting with S signify symbols and those starting with C signify control characters. Note, the spread sheet does not contain all the fields (columns) from the UnicodeData.txt file.

## 7. ASWG Endorsement

With a quorum established the ASWG voted to endorse schema rX, with the included aseXML Change Requests. The voting results are forwarded to AEMO for approval. When 75% of those ASWG members vote to endorse a schema it represents an ASWG recommendation for that schema. AEMO will not reject an ASWG recommendation without first consulting with the ASWG.

The results of the ASWG vote for the final schema to be released are as follows:

Date of Vote: DD/MM/YYYY

Option	# Votes	% Vote
For		
Against		
Abstained		
Total Members Present		100%

## 8. AEMO Approval

The schema approval and approval date are identified below

Status	Date	Authorised by
<b>Approved</b> – The schema has been approved by AEMO and is formally released for use	26/03/2013	Fred Dykman