

REFERENCE
DOCUMENT
FORMAT

EXTERNAL
AUDIENCE

10 DECEMBER 2013
DATE

UNCLASSIFIED
CLASSIFICATION



Australian Government
Australian Taxation Office

FILE REF: CD 01

DATA AND PAYMENT STANDARDS - ERROR CODE MANAGEMENT

Published by the Commissioner of Taxation
Version 1.2
Release date: 10 December 2013
Applies from: 1 July 2013



UNCLASSIFIED



For further information or
questions, email
SuperStreamStandards@ato.gov.au

VERSION CONTROL

Version	Release date	Operative from	Operative up to and including	Description of changes
1.0	9 January 2013	1 July 2013	11 June 2013	N/A
1.1	12 June 2013	1 July 2013	9 December 2013	<p>Updates to incorporate the RSA Data and Payment Standards 2013.</p> <p>Technical updates to schedules incorporating industry and internal review.</p> <p>Updates to align with release of new versions of taxonomy and schematron.</p>
1.2	10 December 2013	1 July 2013	Open	<p>Technical updates to schedules incorporating industry and internal review.</p> <p>Updates to align with release of new versions of taxonomy and schematron.</p>

TABLE OF CONTENTS

1. Purpose.....	4
2. Error codes.....	4
3. Communicating errors	5
3.1 Message event content model.....	5
3.2 Message event Items content model.....	6
3.3 Parameters content model	7
3.4 Locations content model	7
3.5 Applying and interpreting maximum severity codes.....	8
4. Generic SBR error codes	11
5. Standard specific error codes	12
6. Organisation specific error codes.....	14
Appendix A: Message Event Schema.....	15

Note on terminology

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119 <http://www.ietf.org/rfc/rfc2119.txt>. The use of the word “mandatory” is to be read as “MUST”.

1. PURPOSE

This document specifies the error codes that entities must use to comply with the Superannuation Data and Payment Standards 2012 and the RSA¹ Data and Payment Standards 2013 (the Standards).

Note: A reference to 'the Standard' in this document is a reference to whichever Standard is applicable to the entity for the transaction.

This document also allows that an entity may use additional error codes not covered by this document.

2. ERROR CODES

- (a) The rules set out for error codes in this document **MUST** be complied with to comply with the Standard.
- (b) An entity may implement organisation specific error codes that cover errors not included in this document at section 4 (Generic SBR Error Codes) and section 5 (Standard Specific Error Codes).
- (c) If an entity implements organisation specific error codes for errors not included in this document at section 4 (Generic SBR Error Codes) and section 5 (Standard Specific Error Codes), the error codes **MUST** adhere to the convention set out in section 6 of this document.

¹ Retirement Savings Accounts.

3. COMMUNICATING ERRORS

- (a) Error messages **MUST** be communicated through the use of the Event structure. The XML schema describing this structure is at Appendix A.
- (b) The Event structure **MUST** be included as a separate part in the message envelope.
- (c) At least one EventItem **MUST** be provided within the Event. Items are not required to be ordered by severity within an Event.
- (d) Error messages **MUST** conform with the specifications in the tables in sections 3.1 through to 3.5 of this document.
- (e) If there is any confusion about how to resolve ambiguities between Sections 3.1 and 3.5 in applying MaximumSeverity, Section 3.1 should take precedence.

3.1 MESSAGE EVENT CONTENT MODEL

ELEMENT	PURPOSE	OPTIONALITY
MaximumSeverity.Code	<p>Indicates the most severe level of error present in the associated EventItems.</p> <p>Allowable values listed in descending order of importance (most important listed first) are:</p> <ul style="list-style-type: none"> ▪ Progressive ▪ Error ▪ Partial ▪ Warning ▪ Information <p>Where more than one code might apply the code with the higher importance MUST be selected.</p>	MANDATORY
EventItems	Provides specific information about individual message events.	MANDATORY

3.2 MESSAGE EVENT ITEMS CONTENT MODEL

ELEMENT	PURPOSE	OPTIONALITY
Error.Code	Uniquely identifies the condition that has occurred.	MANDATORY
Severity.Code	Categorises the condition that has occurred by severity. Allowable values are (in decreasing order of severity): <ul style="list-style-type: none"> ▪ Error ▪ Warning ▪ Information 	MANDATORY
Short.Description	Concise description of the condition that has occurred. This SHOULD be included for all message event items ² and is recommended to be no longer than 100 characters.	OPTIONAL
Detailed.Description	Extensive description of the condition that has occurred. There is no maximum length for a long description.	OPTIONAL
Parameters	Used to support the insertion of dynamic information into descriptions.	OPTIONAL
Locations	Indicates via an XPath expression the element in the identified business document to which the message event item applies.	MANDATORY

² In the common event of a successful transaction, this element is not required.

3.3 PARAMETERS CONTENT MODEL

ELEMENT	PURPOSE	OPTIONALITY
Parameter.Identifier	Uniquely identifies the parameter used in descriptors. Will appear in a descriptor in conjunction with curly braces (for example {abn}). Maximum length is 80 characters.	MANDATORY
Parameter.Text	The information that is intended to be dynamically inserted into the descriptor in place of the parameter. Maximum length is 4096 characters.	MANDATORY

3.4 LOCATIONS CONTENT MODEL

ELEMENT	PURPOSE	OPTIONALITY
Location.Instance.Identifier	This must be the value of the Property element with attribute @name equal to 'PartID' of the PartProperties section for the business document affected by the error. Every Location.Instance.Identifier within the same Event MUST refer to the same PartID. That is, there is a strict one-to-one relationship between an XBRL business document and the associated Event XML business document.	MANDATORY
Location.Path.Text	Indicates, via an XPath expression, the element to which the Event item refers. It needs to be interpreted in conjunction with the instance identifier field.	OPTIONAL

3.5 APPLYING AND INTERPRETING MAXIMUM SEVERITY CODES

MAXIMUM SEVERITY CODE	RULES AND INTERPRETATION
Error	<ol style="list-style-type: none"> 1. This code MUST only be used when a single event is sent once ALL processing has completed for the associated request message. 2. This code SHALL be interpreted to mean that none of the request message could be successfully processed. 3. All errors MUST be reported through the use of separate EventItems in the Event for this response. 4. Any subsequent events received with respect to the original request message SHALL be ignored.
Partial	<ol style="list-style-type: none"> 1. This code MUST only be used when a single event is sent once ALL processing has completed for the associated request message. 2. This code SHALL be interpreted to mean that one or more components of the associated request message were able to be successfully processed, but that errors were found in the remaining components. 3. All errors discovered MUST be reported through the use of separate EventItems in the Event for this response. 4. EventItems are not required to be reported for successfully processed components. 5. This code SHALL be interpreted to mean that processing was successful for any components for which an error was not declared within the Event. 6. Any subsequent events received with respect to the original request message SHALL be ignored.

Warning	<ol style="list-style-type: none">1. This code MUST only be used when a single event is sent once ALL processing has completed for the associated request message.2. This code SHALL be interpreted to mean that processing was successful for all components included in the associated request message, however one or more warnings were encountered during processing.3. All warnings encountered MUST be reported through the use of separate EventItems in the Event for this response.4. EventItems are not required to be reported for successfully processed components.5. Any subsequent events received with respect to the original request message SHALL be ignored.
Information	<ol style="list-style-type: none">1. This code MUST only be used when a single event is sent once ALL processing has completed for the associated request message.2. This code SHALL be interpreted to mean that processing was successful for all components included in the associated request message.3. A single EventItem MUST be reported with appropriate response code to indicate that processing was successful. Separate EventItems SHOULD NOT be reported for individual components.4. Any subsequent events received with respect to the original request message SHALL be ignored.

Progressive	<ol style="list-style-type: none">1. This code MUST only be used when multiple response messages will be sent for the associated request message (that is, responses commence prior to finalising processing of all components within the associated request message).2. Once electing to use this code to respond to a request message, the sender of the response message MUST NOT send any subsequent response messages with the maximum severity code 'Error', 'Partial', 'Warning' or 'Information'.3. On receiving a response with this maximum severity code the receiver MUST NOT assume that processing is complete for the associated request message.4. Any subsequent response messages received with respect to the original request messages that have a maximum severity code of 'Error', 'Partial', 'Warning' or 'Information' SHALL be ignored.5. All errors discovered and warnings encountered, for the subset of components intended to be addressed in this response message, MUST be reported through the use of separate EventItems in the Event for this response.5. Separate EventItems MUST be reported for each component intended to be addressed in this response message that was successfully processed.
-------------	--

4. GENERIC SBR ERROR CODES

- (a) Other than as provided for in section 5 (standard specific error codes) and section 6 (organisation specific error codes), the error codes used **MUST** be sourced from the generic SBR error codes that are available on the SBR website at <http://www.sbr.gov.au>.

Note: The following generic SBR error codes are directly referenced by the documents *Data and Payment Standards - Contributions Message Implementation Guide* and *Data and Payment Standards - Rollover Message Implementation Guide*.

Error	Severity	Short description	
1	SBR.GEN.GEN.1	Error	ABN {abn} is not valid
2	SBR.GEN.GEN.14	Error	Invalid Entity Identifier Scheme {entityIdScheme}

5. STANDARD SPECIFIC ERROR CODES

- (a) For the errors identified in the table the associated error code as listed MUST be used.

Error Code	Severity	Short description
1 SUPER.GEN.GEN.1	Error	TFN quoted indicator does not match Entity ID scheme.
2 SUPER.GEN.GEN.2	Error	Unique Superannuation Identifier {usi} not known to Superannuation entity ABN {abn}.
3 SUPER.GEN.GEN.3	Error	Employer supplied Member ID used in Entity ID is not unique.
4 SUPER.GEN.GEN.4	Error	Mandatory data element not supplied.
5 SUPER.GEN.GEN.5	Error	Data element contained an unexpected value.
6 SUPER.GEN.GEN.6	Error	Missing context declaration.
7 SUPER.GEN.GEN.7	Error	Conditional data element rule failure.
8 SUPER.GEN.GEN.8	Error	Tuple has too many occurrences for a context declaration.
9 SUPER.GEN.GEN.9	Warning	TFN failed the TFN algorithm check.
10 SUPER.GEN.GEN.10	Error	Too many instances of a context declaration.
11 SUPER.GEN.GEN.11	Error	ABN {abn} not known to the Message Receiver.
12 SUPER.GEN.GEN.12	Error	Payment Reference Number cannot be reconciled to a payment.
13 SUPER.GEN.GEN.13	Error	Unknown Biller Code.
14 SUPER.GEN.GEN.14	Error	Unknown Customer Reference Number.
15 SUPER.GEN.GEN.15	Error	Bank State Branch {bsb} is invalid or not known.
16 SUPER.GEN.GEN.16	Error	Account Number {acntno} is invalid or not known.
17 SUPER.GEN.GEN.17	Error	Account Name {acntname} is invalid or not known.
18 SUPER.GEN.GEN.20	Error	Unknown property within PartProperties.
19 SUPER.GEN.GEN.21	Error	Member not found with supplied information.
20 SUPER.GEN.GEN.22	Error	No Longer a member of Superannuation entity.
21 SUPER.GEN.RLVR.1	Error	SuperannuationRollover.Requested.Amount MUST be provided if SuperannuationRollover.TransferWholeBalance.Indicator is "false".
22 SUPER.GEN.RLVR.2	Information	Rollover Process successful.
23 SUPER.GEN.RLVR.3	Error	ABN defined within the Member Rollover Transaction context was not defined within the Rollover Payment context.
24 SUPER.GEN.RLVR.4	Error	Product ID defined within the Member Rollover Transaction context was not defined within the Rollover Payment context.
25 SUPER.GEN.RLVR.5	Error	Rollover Process unsuccessful.
26 SUPER.GEN.RLVR.6	Error	Rollover could not be processed due to rules within Superannuation entity. Contact Superannuation entity for details.
27 SUPER.GEN.RLVR.7	Error	Rollover could not be processed due to a pending claim.

Error Code	Severity	Short description	
28 ³		Reserved	
29	SUPER.GEN.RLVR.9	Error	The account for the provided member identifier has been closed.
30	SUPER.GEN.CNTRBTN.1	Information	Member registration request message was successfully processed.
31 ⁵		Reserved	
32 ⁶		Reserved	
33	SUPER.GEN.CNTRBTN.4	Error	Contributions cannot be accepted from this Contribution Provider.
34	SUPER.GEN.CNTRBTN.5	Error	Member TFN required for this Contribution.
35	SUPER.GEN.CNTRBTN.6	Error	Payment is less than what has been specified with Contribution Transaction Request Message.
36	SUPER.GEN.CNTRBTN.7	Information	Payment is more than what has been specified with Contribution Transaction Request Message.
37	SUPER.GEN.CNTRBTN.8	Error	Eligibility issue preventing the contribution being processed. Contact Superannuation entity for details.

³ This sequence number was originally intended for SUPER.GEN.RLVR.8 however this error code is no longer to be used.

⁵ This sequence number was originally intended for SUPER.GEN.CNTRBTN.2 however this error code is no longer to be used.

⁶ This sequence number was originally intended for SUPER.GEN.CNTRBTN.3 however this error code is no longer to be used.

6. ORGANISATION SPECIFIC ERROR CODES

(a) Organisation specific error codes MUST adhere to the following convention:

{Sector}.{Organisation}.{Function}.{Id}

Sector = SUPER
Organisation = the ABN of the defining organisation
Function = GEN | RLVR | CNTRBTN
Id = function specific identifier (format may vary across organisations)

Function value descriptions:

- GEN represents general errors
- RLVR represents errors specific to rollovers
- CNTRBTN represents errors specific to registrations or contributions.

APPENDIX A: MESSAGE EVENT SCHEMA

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://sbr.gov.au/comn/event.02.data"
targetNamespace="http://sbr.gov.au/comn/event.02.data" elementFormDefault="qualified"
version="02.01">
  <!-- Reusable types -->
  <xs:simpleType name="Event.80CharString.TextType">
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="80"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Event.4096CharString.TextType">
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="4096"/>
    </xs:restriction>
  </xs:simpleType>
  <!-- Event -->
  <xs:element name="Event" type="tns:EventType"/>
  <xs:complexType name="EventType">
    <xs:sequence>
      <xs:element name="MaximumSeverity.Code" type="tns:MaximumSeverity.CodeType"/>
      <xs:element ref="tns:EventItems"/>
    </xs:sequence>
  </xs:complexType>
  <!-- Event Item -->
  <xs:element name="EventItems" type="tns:EventItemsType"/>
  <xs:complexType name="EventItemsType">
    <xs:sequence>
      <xs:element name="EventItem" type="tns:EventItemType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="EventItemType">
    <xs:sequence>
      <xs:element name="Error.Code" type="tns:Error.CodeType"/>
      <xs:element name="Severity.Code" type="tns:Severity.CodeType"/>
      <xs:element name="Short.Description" type="xs:string" minOccurs="0"/>
      <xs:element name="Detailed.Description" type="xs:string" minOccurs="0"/>
      <xs:element name="Parameters" type="tns:EventItemParametersType" minOccurs="0"/>
      <xs:element name="Locations" type="tns:EventItemLocationsType"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="Error.CodeType">
    <xs:restriction base="tns:Event.80CharString.TextType">
      <xs:pattern value="([A-Z0-9])+.([A-Z0-9])+.([A-Z0-9]).([A-Z0-9])+"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Severity.CodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Error"/>
      <xs:enumeration value="Warning"/>
      <xs:enumeration value="Information"/>
    </xs:restriction>
  </xs:simpleType>

```

```

    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="MaximumSeverity.CodeType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Error"/>
      <xs:enumeration value="Partial"/>
      <xs:enumeration value="Warning"/>
      <xs:enumeration value="Information"/>
      <xs:enumeration value="Progressive"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="EventItemParametersType">
    <xs:sequence>
      <xs:element name="Parameter" type="tns:EventItemParameterType"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="EventItemParameterType">
    <xs:sequence>
      <xs:element name="Parameter.Identifier" type="tns:Event.80CharString.TextType"/>
      <xs:element name="Parameter.Text" type="tns:Event.4096CharString.TextType"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="EventItemLocationsType">
    <xs:sequence>
      <xs:element name="Location" type="tns:EventItemLocationType" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="EventItemLocationType">
    <xs:sequence>
      <xs:element name="Location.Instance.Identifier"
type="tns:Event.4096CharString.TextType"/>
      <xs:element name="Location.Path.Text" type="tns:Event.4096CharString.TextType"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <!-- Event Item Code Lists -->
  <xs:element name="EventItemCodeLists" type="tns:EventItemCodeListsType"/>
  <xs:complexType name="EventItemCodeListsType">
    <xs:sequence>
      <xs:element name="EventItemCodeList" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="tns:EventItems"/>
          </xs:sequence>
          <xs:attribute name="source" type="xs:string" use="required"/>
          <xs:attribute name="version" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```