



Australian Government

**MBR**  
Modernising Business Registers

# MBR Program

# Standardised error reporting

V0.3



## Background

The MBR program will be provisioning a significant number of APIs. There needs to be consideration and agreement on the preferred, standardised error and exception reporting approach. This preferred approach will be co-designed with DSPs as per normal ATO engagement patterns.

This document currently focuses on error, informational and warning reporting from the MBR business services initiated by the APIs that the MBR program will be provisioning.

The ATO has approved changes to the organisation REST Service standards - [IT S95 REST Service](#). The approved ATO's REST standard uses the Whole of Government API standards as its basis, with the ATO REST standard document calling out any ATO Specific requirements.

The recommendation for MBR is to use the WofG standard as the base for error attributes. Other ATO standards will be considered as part of this activity, for example the SBR error message reporting standard. We would extend the WofG standard as required and potentially bring across any SBR features when they add value.

This document focuses on three areas:

1. Discussing the attributes that could be used when reporting an exception event – considering the WofG standard attributes and other ATO approaches.
2. Discussing the attributes in detail, and their OOTB support by the COTS product at the heart of the MBR solution.
3. Discussing the error code structures returned from the COTS product to support a standard error code structure for MBR.

The DSP Technical Focus Group has discussed this subject and the ATO recommendation has been confirmed. The updated document is designed to facilitate discussion and further co-design with DSPs. This document has been updated after wider discussion within the ATO.

## Error attributes

### WHOLE OF GOVERNMENT API STANDARD

The following table shows the attributes of the error object from the [WofG API standard](#):

*For some errors returning the HTTP status code is enough to convey the response. Additional error information can be supplemented in the response body. For example; HTTP 400 Bad request is considered to be too generic for a validation error and more information must be provided in the response body.*

Error attributes	Description	Mandatory?	Example
<i>id</i>	Identifier of the specific error	Optional	"86032cbe-a804-86ce-ec3041e3effc"
<i>detail</i>	A human-readable explanation specific to this occurrence of the problem.	Mandatory	"Suburb contains repeated numerics "
<i>code</i>	An application-specific error code	Mandatory	address.suburb.repeatednumeric error.dataConstraint.Country.invalid
<i>source</i>	An object containing references to the source of the error, optionally including any of the following members: pointer: a JSON Pointer to the associated entity in the request document parameter: a string indicating which URI query parameter caused the error	Optional	n/a
<i>source &gt; pointer</i>	JSON Pointer to the associated entity in the request document	Optional	"pointer": "Director.Address.Line1" "pointer": "Associate[0].DOB"
<i>source &gt; parameter</i>	A string indicating which URI query parameter caused the error.	Optional	"parameter": "LastName"



Example of possible MBR error attributes:

```
{  
  "errors": [{  
    "detail": "Suburb contains repeated numerics",  
    "code" : "address.suburb.repeatednumeric",  
    "source": {  
      "pointer": "Director.Address.Suburb"  
    },  
    {  
      "detail": "Input value(s) exceeded maximum length",  
      "code" : "name.familyname.toolong",  
      "source": {  
        "pointer": "Director.Name.famillyname "  
      }  
    }  
  ]  
}
```

DRAFT

## OTHER ATO SYSTEM ERROR MESSAGE PATTERNS

Standard Business Reporting or SBR is a standard approach to online or digital record-keeping that was introduced by government in 2010 to simplify business reporting obligations. The ATO provides a wide range of digital wholesale services for DSPs through SBR – including those related to:

- tax and business accounting
- payroll
- super
- business registration and other reporting services.

SBR uses Message Events to support reporting the outcome of an API call, the details of this structure are outlined in the SBR [Web services Implementation Guide \(WIG\)](#) – section 4.6 Message Events:

*In order that every message exchange has an explicit indication of its result, every response to a service request MUST include one MessageEvent as part of the Standard Business Document Header. An event MUST include at least one MessageEventItem. Note that items MAY not necessarily be ordered by severity within a MessageEvent.*

The following table shows the 'message events' that SBR provide that may add value for reporting errors from MBR APIs – and if they are covered by the WofG API error object attributes.

Error attributes	Description	Comments
<i>error code</i>	A code to uniquely identify the condition that has occurred	SBR format {Jurisdiction}.{Agency}.{Function}.{Id} SBR.GEN.FAULT.TOOMANYINSTANCES <b>Covered by WofG 'code' attribute.</b>
<i>severity code</i>	The attribute that categorises the message event.	Items are categorised by severity, the options being Information, Warning or Error. <b>Not covered by WofG attributes, but seen as adding significant value.</b>
<i>maximum severity code</i>	The attribute that shows the most server 'category code' of all of the MessageEvents.	<b>Not covered by WofG attributes, not seen as adding significant value. HTTP response code will indicate success/failure of API call.</b>
<i>short description</i>	A concise, human readable description of the condition that has occurred. It is intended for use in visual components such as tool tips.  Recommended < 100 chars	Supports use of markup to support inclusion of a hyperlink – markup vocabulary only to use XHTML.  <b>Covered by WofG 'detail' attribute. However, this does not mention the use of markup.</b>
<i>detailed description</i>	To support a more extensive explanation of a condition than can be reasonably contained within the short description	An event item MAY include a detailed description. Where no detailed information is available, the detailed description SHOULD NOT be provided.  Supports use of markup to support inclusion of a hyperlink.



		<b>Not covered by WofG attributes, not seen required at this point in time for lightweight REST services at this point in time.</b>
<i>Parameters</i>	Support the insertion of dynamic information into descriptions.	Parameters are provided in 'Identifier' and 'Text' pairs. The location in the description where a parameter SHOULD be inserted is represented as the identifier for the parameter, surrounded by curly braces. <b>Not covered by WofG attributes, seen as adding value.</b>
<i>location</i>	An item may have one location object	To support user interface design to highlight fields in which errors have occurred, which would allow client software to intelligently indicate the information affected by the item.
<i>sequence number</i>	Indicates to which payload document in the incoming request the event item applies	Each location must include a sequence number. <b>Not covered by WofG attributes, should not be applicable for MBR services. MBR lightweight REST services will not support multiple document payloads.</b>
<i>location path text</i>	To identify the element in the incoming payload document to which the event item occurs.	The location path field is included in the location to indicate, via an XPath expression, the element in the incoming payload document to which the event item refers. <b>Covered by WofG attributes, seen as adding significant value</b>

The following is an example XML Message event from the SBR WIG, showing the various attributes discussed above.

```
<MessageEvent>
  <Message.Event.MaximumSeverity.Code>Error</Message.Event.MaximumSeverity.Code>
  <MessageEventItems>
    <MessageEventItem>
      <Message.Event.Item.Error.Code>SBR.GEN.GEN.INVALIDABN</Message.Event.Item.Error.Code>
      <Message.Event.Item.Severity.Code>Error</Message.Event.Item.Severity.Code>
      <Message.Event.Item.Short.Description>ABN {abn} is not valid</Message.Event.Item.Short.Description>
      <Message.Event.Item.Detailed.Description>Please check the ABN to confirm that the number is correct. See &lt;a
        href="{url}"&gt;www.ato.gov.au&lt;/a&gt; for more information</Message.Event.Item.Detailed.Description>

      <Parameters>
        <Parameter>
          <Message.Event.Item.Parameter.Identifier>url</Message.Event.Item.Parameter.Identifier>
          <Message.Event.Item.Parameter.Text>http://www.ato.gov.au/path/to/information</Message.Event.Item.Parameter.Text>
        </Parameter>
        <Parameter>
          <Message.Event.Item.Parameter.Identifier>abn</Message.Event.Item.Parameter.Identifier>
          <Message.Event.Item.Parameter.Text>12345678901</Message.Event.Item.Parameter.Text>
        </Parameter>
      </Parameters>

      <Locations>
        <Location>
          <BusinessDocument.Sequence.Number>1</BusinessDocument.Sequence.Number>
          <Message.Event.Item.Location.Path.Text>/xbrli:xbrl/... </Message.Event.Item.Location.Path.Text>
        </Location>
      </Locations>
    </MessageEventItem>
  </MessageEventItems>
</MessageEvent>
```

## RECOMMENDED ATTRIBUTES

The following table lists the attributes from the WofG attribute list and the SBR message attribute list showing:

- Recommendation for including/excluding the attribute from the MBR API error object
- If there is a match between the WofG and SBR attributes.
- If the attribute is recommended, if the COTS product supports this OOTB.

NOTE: If the attribute is recommended and the COTS product does not support this OOTB, the updated MBR program will need to investigate the options available to include the attribute (if possible).

This table has been updated to reflect the latest proposal from an ATO perspective, please see the blue text any changes. Also the order has changed to show all recommended attributes first.

Error attributes	Recommended for inclusion?	Supported by COTS product?	Comments
<i>id</i>	Yes	No.	<p>The previous version of this document called out this ID as being returned from the COTS product as part of the error block. While the COTS platform does return a suitable crash/transaction ID which can be supplied to MBR support staff to enable crash investigation and resolution, crash ID will be returned as part of a response header for Internal error HTTP responses.</p> <p>This attribute will not be populated for the foreseeable future.</p>
<i>detail</i>	Yes	Yes	<p>The COTS product will support substitution of values into this detail line and perform the substitution - rather than return 'paramater' attributes and allow the consumer to perform the substitution.</p> <p>EG: ABN {ABN} IS NOT VALID i.e. the {ABN} would be replaced by the COTS product with the actual ABN value that caused the error. This is done before the error is returned to the consumer.</p> <p><b>SBR attribute: short description</b></p>



<b>Error attributes</b>	<b>Recommended for inclusion?</b>	<b>Supported by COTS product?</b>	<b>Comments</b>
<i>code</i>	Yes	Yes	The COTS product will return codes from business services. The underlying COTS product will also return system codes and other validation error codes.  <b>SBR attribute: error code</b>
<i>source</i>	Yes	No	This higher level object is not supported by the COTS product OOTB  <b>SBR Attribute: Location</b>
<i>source &gt; pointer</i>	Yes	Yes	The COTS product supports this OOTB  <b>SBR attribute: Location Path Text</b>
<i>source &gt; parameter</i>	Yes	No	<a href="#">This will be supported by the ABRS solution. To identify the actual parameter in error.</a>  <b>SBR Attribute: None.</b>
<i>Severity Code</i>	Yes	Yes	The COTS product supports this OOTB. EG: error, warning or information DSG and COTS vendor are keen for this to be included.
<i>extended detail</i>	No	No	<a href="#">In keeping with lightweight API approach, this attribute would not be considered for inclusion.</a> COTS product does not support this extended detail attribute OOTB.  <b>SBR attribute: detailed description</b>
<i>maximum severity code</i>	No	N/A	Attribute is considered not required in the MBR lightweight services. This attribute could also be determined from severity code of individual error and the HTTP response.



Error attributes	Recommended for inclusion?	Supported by COTS product?	Comments
<i>Parameters</i>	No	Partial	<p>COTS product does not support providing Parameter data explicitly as separate attributes.</p> <p>The COTS product does support substitution of 'field value' information in the details attribute.</p> <p>EG: 'The ABN {abn} is invalid, where {abn} should be the actual value of the ABN field will be inserted into the {abn} parameter before being returned from the COTS product.</p> <p>For the URL substitution example, where the URL is just value add into the detailed description, the COTS product does not support this.</p>
<i>Location</i>	No	Yes	<p>The Location object and the underlying location path text attribute is covered by the WofG source &gt; pointer attribute.</p>
<i>Location &gt; Sequence Number</i>	No	N/A	<p>The sequence number attribute was not seen as appropriate outside of the SBR solution.</p>
<i>Location &gt; Path Text</i>	No	Yes	<p>The WofG attribute 'source pointer' supports this requirement to be able to identify the actual field that caused the error item to be returned.</p>

## STANDARD ERROR CODE PATTERNS

MBR program is working towards a consistent format for error codes from the APIs the program delivers.

The program is trying to conform to the emerging standards that the ATO have around error codes. The SBR error code format has been examined as part of this discussion.

At the heart of the MBR program is a COTS product that is both a platform and a host for the numerous configured business services that process APIs requests. Errors returned from the COT product include those built as part of these business services; errors from the underlying platform and errors from a number of consumed pre-built components.

The current proposal is to return the various types of error codes as outlined below. Without standardising the format of the error codes from the platform layer or the Pre-built components. This is proposed as:

- Any transformation of error codes values would be 'post action' from the COTS service.
- Would involve additional logic to scan for each API call to check for error codes that require transformation
- Requires custom logic to perform the transformation.

Source	Error pattern	Comment
Business Services	{object}.{attribute}.{error} EG address.suburb.repeatednumerics	These types of error codes are specified at design time and configured by the MBR build team.  As a default the object would be the appropriate 'data object' – EG Address.  This object <u>could</u> be extended in the COTS product to support object class to support more fine grained errors if desired. EG: postalAddress.suburb.repeatednumerics
Platform layer errors	attribute.number.notNumeric error.dataConstraint.{constraintname}.invalid Eg: error.dataConstraint.Country.invalid	These errors are returned from the platform that underpins the COTS product. EG Java layer errors, or an attribute data definition.  This error format is part of the platform itself. To transform this error code would require custom logic.
Pre-built component errors	attribute.email.invalid	The COTS product business services and platform make use of a number of pre-built components. The errors from these components can be returned by MBR business services and APIs.  Transforming of the error code would require custom logic.

Existing ATO error code formats:

SBR has an error code that follows this format:

**{Jurisdiction}.{Agency}.{Function}.{Id}**

EG:

SBR.GEN.FAULT.TOOMANYINSTANCES

CMN.ATO.TFN.OK

This level error report breakdown was not seen as required for MBR error codes.

DRAFT

## ERROR TRANSFORMATION EXAMPLES

The error, warning and informational messages returned by the COTS APIs will be transformed before being returned to external consumers of the MBR APIs.

This includes attribute names and values. The table below is intended to show an example of mapping outcomes. The colour coding is intended to show the mapping between structures.

The MBR and wider ATO preference for the error block is to align to the whole of Government API standard, the only additional attribute to be included is the severityCode to support the return of warning and informational messages.

Error object returned by COTS APIs	Transformed error block returned externally.	Notes
<pre>"messages": [   {     "path": "Address.Line1",     "code": "address.line1.mandatory",     "message": "Address Line 1 is mandatory",     "level": "error"   },   {     "path": "Address.Suburb",     "code": "address.suburb.corrected",     "message": "Suburb has been corrected",     "level": "warn"   },   {     "path": "Address.Line2",     "code": "address.Line2.merged",     "message": "Line 2 merged with Line 1",     "level": "info"   },   Etc etc</pre>	<pre>"errors": [   { "source": {     "pointer": "CurrentPostalAddress.Line1"   },     "code" : "address.line1.mandatory",     "detail": "Address Line 1 is mandatory",     "severityCode": "error"   },   { "source": {     "pointer": "CurrentPostalAddress.Locality"   },     "code" : "address.suburb.corrected",     "detail": "Suburb has been corrected",     "severityCode": "warning"   },   { "source": {     "pointer": "CurrentPostalAddress.Line2"   },     "code" : "address.line2.merged",     "detail": "Line 2 merged with Line 1",     "severityCode": "information"   },   Etc etc</pre>	<p>The 'path' structure to be transformed.</p> <p>Path attribute value to be transformed.</p> <p>'Message' attribute name to be transformed.</p> <p>COTS level attribute name and values to be transformed.</p>

NOTE: the COTS error, informational and warning structure are still being finalised by the COTS vendor. The structure shown in the left cell is one possible structure, they are also considering the option of returning three different blocks, one for errors, one for warnings and one for informational messages.

**However**, the use of the severityCode in the middle lane is the ATO's proposed structure.

## HTTP RESPONSES AND RETURN OF ERROR, WARNING AND INFORMATIONAL MESSAGES

The 'errors' block is expected to be populated only for certain HTTP responses. I.E. 2xx, 400 and 422 HTTP responses. The table below highlights the combination of errors, warnings and informational messages versus HTTP responses.

HTTP Response	Error codes returned	Warning codes could be returned	Informational codes could be returned
Success – 200, 201 etc	✘	✓	✓
400 – Bad Request	✓	✓	✓
422 – Unprocessable Entity	✓	✓	✓

When an API (EG a POST) supports a validation method, on successful call, the resource will not be created. It is possible that the API could return error, warning and informational codes.