



RESERVE BANK OF AUSTRALIA

Securitisation System B2B API Technical Implementation Notes

April 2015

Version Control

Version	1.2
Date	28 April 2015
Document Owner	Reserve Bank of Australia

Contents

1.	Purpose and Overview	1
2.	Role of B2B API Web Services	1
	2.1 B2B API Web Services Certificate and API UserID	1
	2.2 B2B API SOAP Web Services Details	1
	2.3 B2B API Web Service Methods	1
3.	B2B API Web Services Methods Usage and Detail	2
	3.1 Starting/Ending Submissions and Getting Status	2
	3.2 StartSubmission	4
	3.3 SubmitStream	4
	3.4 EndSubmission	5
	3.5 AbortSubmission	5
	3.6 GetSubmissionStatus	5
	3.7 GetResultsSummary	6
	3.8 GetSubmissionEventSummary	7
4.	B2B API Web Services Description Language (WSDL)	9
5.	Appendix A: Document Control	13

1. Purpose and Overview

This document describes how Information Provider (IP) users can use the B2B API web services to programmatically upload submission files to the Securitisation System. Submissions must comply with the published RBA submission XML schemas (XSD) and Reporting Guidelines in order to be validated.

The B2B API is a standard set of basic functions to allow for automation of the file submission process by IPs.

2. Role of B2B API Web Services

The B2B API Web SOAP service provides a mechanism for IP users to upload submission files programmatically to the Securitisation System. The module is designed as a web service, exposing a SOAP version 1.1 endpoint for use.

The service allows IP users to perform the following functions programmatically:

- upload files (from the IP to the RBA)
- retrieve submission metadata to determine a submission's status and/or result details

2.1 B2B API Web Services Certificate and API UserID

Before an IP user can use the web services API's, the IP must submit an API access request via the RBA Securitisation Support Centre (email ssc@rba.gov.au or phone **1800 919 211**). The RBA will issue a **client certificate** and a **new API-only UserID** approved for submissions. The IP can then connect to the RBA API Web site programmatically using the issued certificate.

(Please note that the API-only UserID provided to the IP will be required as an input parameter to relevant API function calls as an additional layer of identity verification.)

2.2 B2B API SOAP Web Services Details

The web services use the following message transfer technology:

Communication protocol	HTTPS
SOAP Version	1.1
Transfer Mode	Streamed
Message Encoding	Text

2.3 B2B API Web Service Methods

The B2B API web services expose the following methods:

Name	Description
StartSubmission	Initiates a submission session from an IP.
SubmitStream	Uploads file(s) as binary stream
AbortSubmission	Aborts the submission session
EndSubmission	Indicates the end of submission session
GetSubmissionStatus	Retrieves the pass/fail status for a submission
GetResultsSummary	Retrieves summary results for all files within a submission
GetSubmissionEventSummary	Retrieves event-level details for a single file within a submission
GetSubmissionRuleInfo	Retrieves rule-level details for a single file within a submission

3. B2B API Web Services Methods Usage and Detail

The following sections describe the web service methods usage and parameters that are used by the web services methods, and what each method returns.

Key Mandatory Development Principles to implement for successful API usage:

1. Session Management - cookies enabled on the client.
The first POST request (StartSubmission) will create the session via the F5 APM.

Following is an example of the session cookies:

cookie-sec

!E4BwpFmfdKiBf991jXFpS+V038EI++7T7b3i9yXcTNGKIC3TzB4RIecPL7mUJhfSQz2NpXkCQCzQ

LastMRH_Session 6755c2b9

MRHSession 52b934ca5f4c67b3a0f05d676755c2b9

2. The subsequent POST requests (SubmitStream, GetSubmissionStatus, EndSubmission) **MUST** include these cookies for the session to remain active and allow the server-side transfer to complete.
The session management is IMPORTANT - If it is not maintained, the APM will automatically create a new session for each request and kill the previous session with the client. The new client session will receive a "200 OK" message HOWEVER the previous session gets deleted, and the server-side data transfer is stopped. Resulting in a failed upload because the data stream is truncated – due to the kill previous session forced by new client request.
3. The "SubmitStream" POST request should include a custom HTTP header, this is mandatory and if not provided upload attempts via the API will fail.

The custom HTTP Header that **MUST** be included is:

"Turn-Off-Clientless-Mode: Yes"

3.1 Starting/Ending Submissions and Getting Status

1. Specify the B2B Web service URL. **https://securitisation-api.rba.gov.au/app/API.svc**
2. Connect and authenticate to service using the client certificate issued by the RBA

Post a successful **Step 2**, there are two execution path options:

Path 1: Get results of a submission using **"GetSubmissionStatus"**, **"GetResultsSummary"**, **"GetSubmissionEventSummary"** or **"GetSubmissionRuleInfo"**

3. Retrieve Submission Results/Status Details using the **GetSubmissionStatus()** method.
(Request and receive metadata for submissions using a valid passed GUID parameter. Submission metadata comprises information about files uploaded as well as the processing status of the submission files.) Further details about the submission can be retrieved using the **GetResultsSummary()**, **GetSubmissionEventSummary()** and **GetSubmissionRuleInfo()** methods.

Path 2: Upload submission file or files

3. Each submission session must begin with a **“StartSubmission”** request. Call the **StartSubmission()** method to start a submission batch session, passing API UserID (issued by the RBA) as a parameter to the **StartSubmission()** method. A GUID (representing the session) is returned on a successful call *(The application will confirm that the API UserID corresponds to the Organisation, and that the specified user is permitted to submit files through the B2B API web service. If an IP fails authorisation, the upload process is stopped immediately)*
4. Upload one or more files using multiple calls of the **SubmitStream()** method. As files are uploaded, they are checked for viruses, malicious VBA code and invalid XSD structure.
*(If a file fails to upload as indicated by an **InvalidOperationException** being thrown by the **SubmitStream()** method, the client application may retry upload for that file. The retry can take place at any point prior to the **EndSubmission()** method, and only the individual file which failed to upload needs to be re-sent. It is not necessary to resend files which were successfully received by the B2B service, as they are cached in the system. There is no limit to the number of retries that can be performed, and likewise the system presently does not impose a time limit for completing a submission. In the case of a multiple file submission, files can be sent in any order, it is not necessary to submit them in a particular sequence)*
5. End submission batch using the **EndSubmission()** method, or Abort submission batch using the **AbortSubmission()** method.
*(“**EndSubmission**” triggers an Upload Confirmation email to the client, and proceeds with further background data validation processing, unless any of the files in the submission had failed validation. If any files within the submission failed validation (were identified as containing a virus, malicious code, or malformed XML), the entire submission is aborted. “**AbortSubmission**” triggers an Upload Confirmation email to the client and Aborts the submission batch and ends all processing)*
6. Receive Upload Confirmation email, if either **“EndSubmission”** or **“AbortSubmission”** used.
7. Retrieve Submission Results/Status Details using the **GetSubmissionStatus()** method.
(Request and receive metadata for submissions using a valid passed GUID parameter. Submission metadata comprises information about files uploaded as well as the processing status of the submission files.) Further details about the submission can be retrieved using the **GetResultsSummary()** , **GetSubmissionEventSummary()** and **GetSubmissionRuleInfo()** methods.

Key Additional Information:

*Uploaded files are cached in the system between the calls to **StartSubmission()** and **EndSubmission()**.*

Upload rejections are caused when the submitted file:

- Does not have the proper file extension (“zip”, “xml”). “zip” files can only contain a single submission “xml” file, you cannot submit multiple xml files within a “zip” file
- Was identified as containing a virus
- Does not conform to the XML Schema Definition (XSD) provided by the RBA
- Contains potentially malicious VBA code

After a submission has been ended or aborted, the client receives the following notifications:

- The **Upload confirmation email** contains the list of files in the submission batch and their respective status.
- The **Submissions Receipt** contains the final validation results so that the client knows which files have been accepted by the system and which were rejected.

3.2 StartSubmission

StartSubmission – initiates the submission session.

Parameters

Name	Data type	Description
User ID	String	The RBA supplied API UserID who has been given the privilege to submit.

Returns

Unique identifier (guid) - if submission initialisation is successful.

Unique identifier (guid) with empty value - if submission initialisation fails.

Submission failures primarily occur when the IP organisation or the submitting user is not currently authorised for API submissions.

The client gets an `InvalidOperationException` with the appropriate error message for any failures.

3.3 SubmitStream

SubmitStream – uploads the file as binary stream.

Parameters

Name	Data type	Description
RemoteFileInfo	MessageContract	<p>Headers</p> <p>Guid – returned by the successful StartSubmission request FileName – the location of the source file on the client side Length – file size</p> <p>Body</p> <p><i>FileByteStream</i> – file content</p>

Returns

No return value – if success

`InvalidOperationException` with the appropriate error message-if failure

In the event of upload failure, the `SubmitStream` method can be called again to retry sending the file.

3.4 EndSubmission

EndSubmission – indicates the end of the submission.

Parameters

Name	Data type	Description
Guid	String	Guid – returned by the successful StartSubmission request

Returns

True - if success

False and `InvalidOperationException` with the appropriate error message– if failure

3.5 AbortSubmission

AbortSubmission – aborts the submission.

Parameters

Name	Data type	Description
Guid	String	Guid – returned by the successful StartSubmission request

Returns

True - if success

False and `InvalidOperationException` with the appropriate error message – if failure

3.6 GetSubmissionStatus

GetSubmissionStatus - returns the number of uploaded files and rejected files for a submission.

Parameters

Name	Data type	Description
Guid	String	Guid – returned by the successful StartSubmission request or from a previous valid submission session

Returns

Name	Data type	Description
SubmissionSummary	Object	Properties Guid – returned by the successful StartSubmission request Total_file_number – number of the uploaded files Rejected_file_number – number of rejected files

3.7 GetResultsSummary

GetResultsSummary – returns results summary for each of the files within the submission.

Parameters

Name	Data type	Description
Guid	String	Guid – returned by the successful StartSubmission request or from a previous valid submission session
User ID	String	The RBA supplied API UserID who has been given the privilege to submit.

Returns

Name	Data type	Description
ResultSummary	Object	<p>Properties</p> <p>Guid – returned by the successful StartSubmission request</p> <p>Err_message – error message</p> <p>Processing_status – indicates if the submission is “Completed” or still “In Process”</p> <p>Submissions – array containing submission information for each of the uploaded files. Note that Submissions is only populated once all files within the submission have been processed.</p>

Each member of the *Submission* list returned by GetResultsSummary is a *SubmissionResult* object:

Name	Data type	Description
SubmissionResult	object	<p>Properties</p> <p>Date – date when submission has been executed</p> <p>deal_businesskey – deal business key</p> <p>report_date – report date</p> <p>submission_id – ID assigned to each of the files uploaded within a submission</p> <p>collateral_flags – collateral flags</p> <p>coupon_month – true/false value</p> <p>submission_status – Pass,Fail or Pass(Warning)</p> <p>validation_results – for each of the submitted files contains the results of validations</p>

Validation_results list contains *SubmissionEventResult* objects:

Name	Data type	Description
SubmissionEventResult	object	<p>Properties</p> <p>event_name – name of the validation event</p> <p>event_status – Pass,Fail or Pass(Warning)</p> <p>failure_count – number of events where event_status = Fail</p> <p>warning_count – number of events where event_status = Pass(Warning)</p>

3.8 GetSubmissionEventSummary

GetSubmissionEventSummary – retrieves event-level details for a single file within a submission.

Parameters

Name	Data type	Description
User ID	string	The RBA supplied API UserID who has been given the privilege to submit.
Submission_ID	long	ID assigned to each of the files uploaded within a submission
EventName	string	Name of the validation event

Returns

Name	Data type	Description
SubmissionEventSummary	object	<p>Properties</p> <p>date – date when submission has been executed</p> <p>deal_businesskey – deal business key</p> <p>report_date – report date</p> <p>submission_id – ID assigned to each of the files uploaded within a submission</p> <p>err_message – error message</p> <p>event_name – name of the validation event</p> <p>event_status – Pass,Fail or Pass(Warning)</p> <p>failure_count – number of events where event_status = Fail</p> <p>warning_count – number of events where event_status = Pass(Warning)</p> <p>rule_summary – contains information for each of the validation rules which completed with Fail or Pass (Warning)</p> <p>status</p>

The *GetSubmissionEventSummary* request will return information only for events which completed with Fail or Pass (Warning) status. If the method is called for submissions which completed in “Pass” status, the *err_message* property of the returned object indicates that further event information is not available.

Rule_summary list is populated with *ValidationRuleSummary* objects for each rule which completed with Fail or Pass (Warning) status.

Name	Data type	Description
ValidationRuleSummary	object	<p>Properties</p> <p>event_detail – details of the validation event</p> <p>description – event description</p> <p>rule_id – validation rule</p> <p>rule_name – validation rule name</p> <p>severity – Fail or Pass(Warning)</p>

3.9 GetSubmissionRuleInfo

GetSubmissionRuleInfo – Retrieves rule-level details for a single file within a submission.

Parameters

Name	Data type	Description
User ID	string	The RBA supplied API UserID who has been given the privilege to submit.
Submission_ID	long	ID assigned to each of the files uploaded within a submission
EventName	string	Name of the validation event
Rule_ID	long	ID assigned to Rule

Returns

Name	Data type	Description
SubmissionValidationRuleInfo	object	<p>Properties</p> <ul style="list-style-type: none"> date – date when submission has been executed deal_businesskey – deal business key description – description err_message – error message event_detail – event detail event_name – name of the validation event formula_variables – list of variables used in rule report_date – report date rule_details – contains information for each rule which completed with Fail or Pass (Warning) status rule_formula – formula being applied by rule rule_name – name allocated to rule in RBA system (can change) submission_id – ID assigned to each of the files uploaded within a submission

The *GetSubmissionRuleInfo* request will return information only for events which completed with Fail or Pass (Warning) status. If the method is called for submissions which completed in “Pass” status, the *err_message* property of the returned object indicates that further event information is not available.

Rule_details list is populated with *ValidationRuleDetails* objects for each rule which completed with Fail or Pass (Warning) status.

Name	Data type	Description
ValidationRuleDetails	object	<p>Properties</p> <ul style="list-style-type: none"> identifier – respective submission record from submission node rule executed on node – respective xml submission data node rule executed on node_result – Fail or Pass(Warning) values – actual values and result from respective submission record used in rule execution

4. B2B API Web Services Description Language (WSDL)

The following Web Services Description Language (WSDL) shows the definitions used by the B2B API Web services.

```
<?xml version="1.0" encoding="UTF-8"?>
<wSDL:definitions xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="https://www.rba.gov.au/securitisation/submission/2014-12"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsa10="http://www.w3.org/2005/08/addressing"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:i0="http://tempuri.org/" xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="https://www.rba.gov.au/securitisation/submission/2014-12" name="SubmissionAPI">
  <wSDL:import location="https://securitisation-api.rba.gov.au/app/API.svc?wSDL=wSDL0"
namespace="http://tempuri.org/" />
  <wSDL:types>
    <xsd:schema targetNamespace="https://www.rba.gov.au/securitisation/submission/2014-12/Imports">
      <xsd:import namespace="https://www.rba.gov.au/securitisation/submission/2014-12"
schemaLocation="https://securitisation-api.rba.gov.au/app/API.svc?xsd=xsd0" />
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/System"
schemaLocation="https://securitisation-api.rba.gov.au/app/API.svc?xsd=xsd2" />
      <xsd:import namespace="http://schemas.microsoft.com/2003/10/Serialization/"
schemaLocation="https://securitisation-api.rba.gov.au/app/API.svc?xsd=xsd1" />
      <xsd:import namespace="http://schemas.microsoft.com/Message"
schemaLocation="https://securitisation-api.rba.gov.au/app/API.svc?xsd=xsd3" />
      <xsd:import namespace="http://schemas.datacontract.org/2004/07/SubmissionsWCFServiceLib"
schemaLocation="https://securitisation-api.rba.gov.au/app/API.svc?xsd=xsd4" />
    </xsd:schema>
  </wSDL:types>
  <wSDL:message name="ISubmissionAPI_StartSubmission_InputMessage">
    <wSDL:part name="parameters" element="tns:StartSubmission" />
  </wSDL:message>
  <wSDL:message name="ISubmissionAPI_StartSubmission_OutputMessage">
    <wSDL:part name="parameters" element="tns:StartSubmissionResponse" />
  </wSDL:message>
  <wSDL:message name="ISubmissionAPI_StartSubmission_InvalidOperationExceptionFault_FaultMessage">
    <wSDL:part name="detail" element="q1:InvalidOperationException"
xmlns:q1="http://schemas.datacontract.org/2004/07/System" />
  </wSDL:message>
  <wSDL:message name="RemoteFileInfo">
    <wSDL:part name="FileByteStream" element="tns:FileByteStream" />
  </wSDL:message>
  <wSDL:message name="RemoteFileInfo_Headers">
    <wSDL:part name="FileName" element="tns:FileName" />
    <wSDL:part name="Guid" element="tns:Guid" />
    <wSDL:part name="Length" element="tns:Length" />
  </wSDL:message>
  <wSDL:message name="ISubmissionAPI_SubmitStream_OutputMessage">
  </wSDL:message>
  <wSDL:message name="ISubmissionAPI_SubmitStream_InvalidOperationExceptionFault_FaultMessage">
    <wSDL:part name="detail" element="q2:InvalidOperationException"
xmlns:q2="http://schemas.datacontract.org/2004/07/System" />
  </wSDL:message>
  <wSDL:message name="ISubmissionAPI_EndSubmission_InputMessage">
    <wSDL:part name="parameters" element="tns:EndSubmission" />
  </wSDL:message>
</wSDL:definitions>
```

```

</wsdl:message>
<wsdl:message name="ISubmissionAPI_EndSubmission_OutputMessage">
  <wsdl:part name="parameters" element="tns:EndSubmissionResponse"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_EndSubmission_InvalidOperationExceptionFault_FaultMessage">
  <wsdl:part name="detail" element="q3:InvalidOperationException"
    xmlns:q3="http://schemas.datacontract.org/2004/07/System"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_AbortSubmission_InputMessage">
  <wsdl:part name="parameters" element="tns:AbortSubmission"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_AbortSubmission_OutputMessage">
  <wsdl:part name="parameters" element="tns:AbortSubmissionResponse"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_AbortSubmission_InvalidOperationExceptionFault_FaultMessage">
  <wsdl:part name="detail" element="q4:InvalidOperationException"
    xmlns:q4="http://schemas.datacontract.org/2004/07/System"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionStatus_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSubmissionStatus"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionStatus_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSubmissionStatusResponse"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionStatus_InvalidOperationExceptionFault_FaultMessage">
  <wsdl:part name="detail" element="q5:InvalidOperationException"
    xmlns:q5="http://schemas.datacontract.org/2004/07/System"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetResultsSummary_InputMessage">
  <wsdl:part name="parameters" element="tns:GetResultsSummary"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetResultsSummary_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetResultsSummaryResponse"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetResultsSummary_InvalidOperationExceptionFault_FaultMessage">
  <wsdl:part name="detail" element="q6:InvalidOperationException"
    xmlns:q6="http://schemas.datacontract.org/2004/07/System"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionEventSummary_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSubmissionEventSummary"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionEventSummary_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSubmissionEventSummaryResponse"/>
</wsdl:message>
<wsdl:message
  name="ISubmissionAPI_GetSubmissionEventSummary_InvalidOperationExceptionFault_FaultMessage">
  <wsdl:part name="detail" element="q7:InvalidOperationException"
    xmlns:q7="http://schemas.datacontract.org/2004/07/System"/>
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionRuleInfo_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSubmissionRuleInfo" />
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionRuleInfo_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSubmissionRuleInfoResponse" />
</wsdl:message>
<wsdl:message name="ISubmissionAPI_GetSubmissionRuleInfo_InvalidOperationExceptionFault_FaultMessage">
  <wsdl:part name="detail" element="q8:InvalidOperationException"
    xmlns:q8="http://schemas.datacontract.org/2004/07/System" />
</wsdl:message>
<wsdl:portType name="ISubmissionAPI">
  <wsdl:operation name="StartSubmission">
    <wsdl:input message="tns:ISubmissionAPI_StartSubmission_InputMessage"
      wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-

```

```

12/ISubmissionAPI/StartSubmission"/>
<wsdl:output message="tns:ISubmissionAPI_StartSubmission_OutputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/StartSubmissionResponse"/>
<wsdl:fault name="InvalidOperationExceptionFault"
message="tns:ISubmissionAPI_StartSubmission_InvalidOperationExceptionFault_FaultMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/StartSubmissionInvalidOperationExceptionFault"/>
</wsdl:operation>
<wsdl:operation name="SubmitStream">
<wsdl:input name="RemoteFileInfo" message="tns:RemoteFileInfo"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/SubmitStream"/>
<wsdl:output message="tns:ISubmissionAPI_SubmitStream_OutputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/SubmitStreamResponse"/>
<wsdl:fault name="InvalidOperationExceptionFault"
message="tns:ISubmissionAPI_SubmitStream_InvalidOperationExceptionFault_FaultMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/SubmitStreamInvalidOperationExceptionFault"/>
</wsdl:operation>
<wsdl:operation name="EndSubmission">
<wsdl:input message="tns:ISubmissionAPI_EndSubmission_InputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/EndSubmission"/>
<wsdl:output message="tns:ISubmissionAPI_EndSubmission_OutputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/EndSubmissionResponse"/>
<wsdl:fault name="InvalidOperationExceptionFault"
message="tns:ISubmissionAPI_EndSubmission_InvalidOperationExceptionFault_FaultMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/EndSubmissionInvalidOperationExceptionFault"/>
</wsdl:operation>
<wsdl:operation name="AbortSubmission">
<wsdl:input message="tns:ISubmissionAPI_AbortSubmission_InputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/AbortSubmission"/>
<wsdl:output message="tns:ISubmissionAPI_AbortSubmission_OutputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/AbortSubmissionResponse"/>
<wsdl:fault name="InvalidOperationExceptionFault"
message="tns:ISubmissionAPI_AbortSubmission_InvalidOperationExceptionFault_FaultMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/AbortSubmissionInvalidOperationExceptionFault"/>
</wsdl:operation>
<wsdl:operation name="GetSubmissionStatus">
<wsdl:input message="tns:ISubmissionAPI_GetSubmissionStatus_InputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/GetSubmissionStatus"/>
<wsdl:output message="tns:ISubmissionAPI_GetSubmissionStatus_OutputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/GetSubmissionStatusResponse"/>
<wsdl:fault name="InvalidOperationExceptionFault"
message="tns:ISubmissionAPI_GetSubmissionStatus_InvalidOperationExceptionFault_FaultMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/GetSubmissionStatusInvalidOperationExceptionFault"/>
</wsdl:operation>
<wsdl:operation name="GetResultsSummary">
<wsdl:input message="tns:ISubmissionAPI_GetResultsSummary_InputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/GetResultsSummary"/>
<wsdl:output message="tns:ISubmissionAPI_GetResultsSummary_OutputMessage"
wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
12/ISubmissionAPI/GetResultsSummaryResponse"/>

```

```

    <wsdl:fault name="InvalidOperationExceptionFault"
      message="tns:ISubmissionAPI_GetResultsSummary_InvalidOperationExceptionFault_FaultMessage"
      wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
        12/ISubmissionAPI/GetResultsSummaryInvalidOperationExceptionFault"/>
  </wsdl:operation>
  <wsdl:operation name="GetSubmissionEventSummary">
    <wsdl:input message="tns:ISubmissionAPI_GetSubmissionEventSummary_InputMessage"
      wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
        12/ISubmissionAPI/GetSubmissionEventSummary"/>
    <wsdl:output message="tns:ISubmissionAPI_GetSubmissionEventSummary_OutputMessage"
      wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
        12/ISubmissionAPI/GetSubmissionEventSummaryResponse"/>
    <wsdl:fault name="InvalidOperationExceptionFault"
      message="tns:ISubmissionAPI_GetSubmissionEventSummary_InvalidOperationExceptionFault_FaultMessage"
      wsaw:Action="https://www.rba.gov.au/securitisation/submission/2014-
        12/ISubmissionAPI/GetSubmissionEventSummaryInvalidOperationExceptionFault"/>
  </wsdl:operation>
  <wsdl:operation name="GetSubmissionRuleInfo">
    <wsdl:input wsaw:Action="http://www.rba.gov.au/securitisation/submission/2014-
      12/ISubmissionAPI/GetSubmissionRuleInfo"
      message="tns:ISubmissionAPI_GetSubmissionRuleInfo_InputMessage" />
    <wsdl:output wsaw:Action="http://www.rba.gov.au/securitisation/submission/2014-
      12/ISubmissionAPI/GetSubmissionRuleInfoResponse"
      message="tns:ISubmissionAPI_GetSubmissionRuleInfo_OutputMessage" />
    <wsdl:fault wsaw:Action="http://www.rba.gov.au/securitisation/submission/2014-
      12/ISubmissionAPI/GetSubmissionRuleInfoInvalidOperationExceptionFault"
      name="InvalidOperationExceptionFault"
      message="tns:ISubmissionAPI_GetSubmissionRuleInfo_InvalidOperationExceptionFault_FaultMessage" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:service name="SubmissionAPI">
  <wsdl:port name="basic_rba" binding="i0:basic_rba">
    <soap:address location="https://securitisation-api.rba.gov.au/app/API.svc"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

5. Appendix A: Document Control

Version	Date	Details
1.2	28 April 2015	Added ZIP file submission capability and additional API function, Section 3.9 (GetSubmissionRuleInfo). Section 2.3, 3.1 and 4 have been updated to reflect additional API function, GetSubmissionRuleInfo.
1.1	23 January 2015	Further information added to the opening paragraph in Section 3 outlining the Key Mandatory Development Principles to implement for successful API usage .
1.0	8 December 2014	The Securitisation System B2B API Technical Implementation Notes (version 1.0) published