

The Derivatives Service Bureau (DSB)

REST Rules of Engagement

Author: The Derivatives Service Bureau (DSB), Technical support team
Email: technical.support@anna-dsb.com
Date: 30 August 2017
Version: 2.02 RC1
Designation: Rules of Engagement

Proprietary Information

Derivatives Service Bureau (DSB) provides this publication “as is” without warranty of any kind, either express or implied, including but not limited to warranties of merchantability. Occasionally, the author may apply changes to the information within this document and these changes will be incorporated to subsequent revisions of this document. These periodic changes may include revisions to typographical errors and technical inaccuracies.

Copyright Derivatives Service Bureau (DSB) © 2017

All company, product, and service names are hereby acknowledged

Contents

Proprietary Information	2
Preface	5
Change History	5
1 Introduction	5
1.1 Document Purpose	5
1.2 Intended Audience	5
1.3 Scope	5
1.4 Contact Information	5
1.5 Functional Summary	5
1.6 Activity Diagram Summary	6
1.7 Document Structure	7
2 Site Preparation	7
2.1 Introduction	7
2.2 JSON Product Definitions Representation as JSON Schema	9
3 REST API methods	11
3.1 Introduction	11
3.2 Obtain JSON schemas	11
3.3 Retrieve or Create ISIN Record by Attribute	12
3.4 Retrieve ISIN Record by Attribute	13
3.5 Retrieve ISIN Record by ISIN	14
3.6 Search for ISIN Records by attributes	15
3.7 Download historical ISINs	17
4 Methods Reference	18
4.1 Introduction	18
4.2 Obtain schema names	18
4.3 Obtain single schema	18
4.4 Retrieve or Create ISIN Record by Attribute Retrieve ISIN Record by Attribute	19
4.5 Retrieve ISIN Record by ISIN	20
4.6 Search for ISIN Records by attributes	21
4.7 Download Historical ISINs	22
5 REST Methods Samples	23
5.1 Introduction	23
5.2 Obtain schema names	23
5.3 Obtain single schema	23

5.4	Retrieve or Create ISIN Record by Attribute	24
5.5	Retrieve ISIN Record by Attribute	25
5.6	Retrieve ISIN Record by ISIN	26
5.7	Search for ISIN Records by attributes	27
5.8	Download Historical ISINs	28
About Derivatives Service Bureau (DSB)		29

Preface

Change History

Date	Change	Version	Author	Revision Details
7 June 2017	Creation	1.00 RC1	Yuval Cohen	Initial Version
26 June 2017	Additions	1.01 RC1	Yuval Cohen	Added: File download workflows
18 July 2017	Corrections and amendments	1.02 RC1	Yuval Cohen	Titles of section 5.4 and 5.5 got mismatched. Update /search to be a GET method Added clarifications in section 3 and 4
27 July 2017	Additions	2.00 RC1	Yuval Cohen	Clarification about token Added workflow: Test for ISIN existence
3 August 2017	Additions	2.01 RC1	Yuval Cohen	Clarifications about status codes and maintenance schedule
30 August 2017	Corrections and amendments	2.02 RC1	Hannah Cabuyao	Updated the file download URL for "Download historical ISINs" method Updated the GitHub Product Definition URL Updated the product definition examples Corrected the sample method for GET /schemas?SchemaName on section 3.2 Corrected the sample method for POST /records&create=false on section 5.5

1 Introduction

1.1 Document Purpose

This specification defines the implementation of the Representational State Transfer (REST or RESTful) Web Service by the Derivatives Service Bureau (DSB) Limited for determining the ISIN for a financial instrument. In doing so it aims to provide a comprehensive reference guide to any such institutions who wish to automate the usage of the DSB REST API Web service.

1.2 Intended Audience

Anyone with an interest in creating or querying ISINs for financial OTC derivatives instruments using an automated REST API web service.

1.3 Scope

This document focuses on the use of the DSB REST API to define and query for ISINs for OTC derivative financial products.

1.4 Contact Information

Please direct your questions on the REST API via email technical.support@anna-dsb.com

1.5 Functional Summary

The DSB REST API provides a near real time service to determine ISINs for OTC derivative financial instruments. Each financial instrument supported by the DSB service is defined by a unique set of attributes verified by industry.

1.6 Activity Diagram Summary

The following diagram illustrates, the DSB REST API Web service:

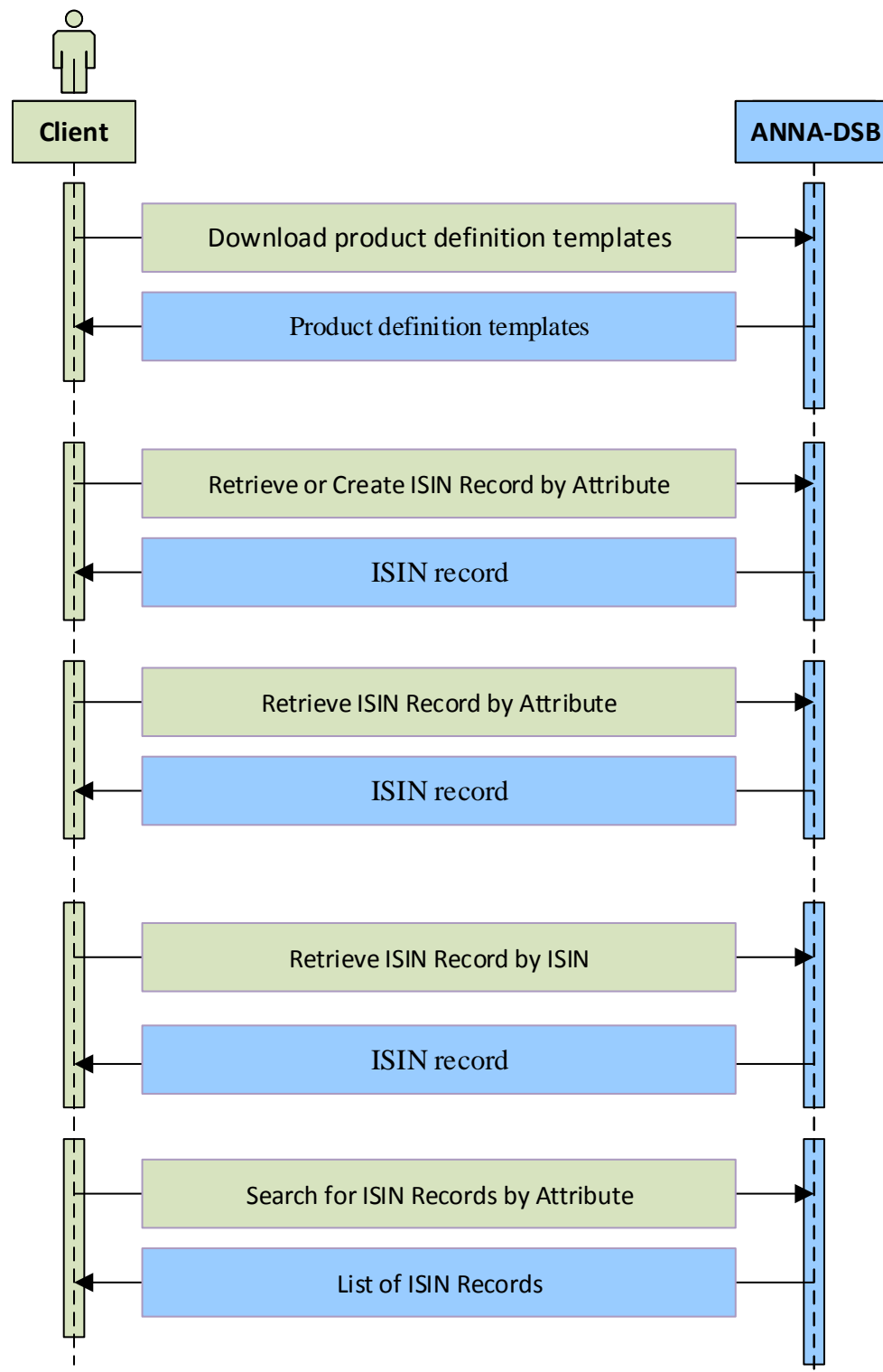


Diagram 1: Activity Diagram

1.7 Document Structure

This document contains the following sections:

Section Number	Title	Description
Section 1	Introduction	A brief introduction to this document providing background to the purpose of the document and the DSB REST API Web service.
Section 2	Site Preparation	Provides details of the REST API implementation and the tasks that are required to use the service.
Section 3	REST API methods	Provides details of the methods and workflows that are supported DSB.
Section 4	REST API methods Reference	Contains the definitions of the methods that are supported by the DSB REST API service including possible arguments and expected response codes.
Section 5	Methods samples	Some sample methods that illustrate the possible contents of the methods as well as their response.

2 Site Preparation

2.1 Introduction

The following preparation is required to use the DSB REST API Web Service:

- Users should follow and adhere to the [REST architectural constraints](#) called RESTful APIs.
- Web service methods are transmitted over the internet using HTTPS (HTTP secure over SSL) protocol through port 443
- DSB operations will provide authentication parameters: username and password
 - Users should attach the authentication parameters to each of the REST methods' header with:
key = "Authorization"
value = base64(username + ":" + password)

The username and password are concatenated with a colon separator and encoded in [base64](#).

- Access to the DSB Web Service is by default restricted to a throughput of one message per minute. Users wishing to increase this throughput rate may ask DSB operations to provide a DSB-Token for use with the REST API. The token increases the message rate for Power Users as per the DSB Acceptable User Policy (currently set at 60 messages per minute).
 - Users should attach the DSB-token to each of the REST methods' header with:
key = "DSB-Token"
value = <token>

DSB Web Service responds with status code 429 (too many request) when a client exceeds its rate limit.
- Users should make any network/firewall configuration changes required to enable connectivity to the base DSB websites (below) over port 443
- The default Base URL for REST methods (except file-download – see [Download Historical ISINs](#)) are:
 - Production environment: <https://www.anna-dsb.com/api>

- UAT environment: <https://uat.anna-dsb.com/api>
- The response to all methods in the REST API contains “responseCode” which follows the [HTTP status codes](#).
 - In case of either client errors (4XX) or Server errors (5XX) the response will contain indicative (human readable) “message”.
- The DSB Web Service schedule weekly maintenance periods which currently commence on Saturday 12:00:00 and finish on Sunday 12:00:00 [mid-day] UTC. During these hours, all REST methods are expected to be responded to with HTTP status code 503 (Service unavailable).
- Common HTTP status codes:
 - 200: OK, Success.
 - 401: Unauthorized
 - 429: too many request [The user has sent too many requests in a given amount of time. Intended for use with rate-limiting schemes]
 - 503: Service unavailable: [The server is currently unavailable because it is down for maintenance]

Additional HTTP status codes are listed in section 4.

2.2 JSON Product Definitions Representation as JSON Schema

2.2.1 Product Definitions

The DSB Product Committee defines the set of Product Definitions for all OTC derivatives in scope and any future changes or additions will be made under the advisement of the Product Committee. Each Product Definition can be identified by:

- Asset Class
- Instrument Type
- Product (Use case)
- Level

Further information about the Product Definition will be available on our web-site.

2.2.2 JSON and JSON Schema

JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. JSON Schema describe existing data format in a clear, human and machine readable documentation and provides complete structural validation which is useful to validate the client submitted data.

JSON standards are available in [ECMA-404](#) as well as in [RFC-7159](#).

JSON Schema standards are available in: [JSON-SCHEMA-04](#)

2.2.3 Product Definition as JSON Schema

The Product Definitions are formatted in a machine readable format in a set of JSON schema files, which are made available by the DSB system. Users are expected to use these JSON schema when requesting an ISIN via the RESTful API.

For each Product Definition there are two JSON schema files:

1	File: Request schema file Description: a schema that defines the attributes in order to request a new ISIN Naming Convention: Request.<AssetClass>.<InstrumentType>.<Product>.<Level>.json Example of a name: Request.Rates.Swap.Fixed_Float_OIS.InstRefDataReporting.json
2	File: Record schema file Description: a schema that defines the attributes in the ISIN Record which is returned from the DSB Naming Convention: <AssetClass>.<InstrumentType>.<Product>.<Level>.<Version>.json Example of a name: Rates.Swap.Fixed_Float_OIS.InstRefDataReporting.V1.json

The example for the name above is for a Product Definition where:

- Asset Class = Rates
- Instrument Type = Swap
- Product (Use Case) = Fixed_Float_OIS
- Level = InstRefDataReporting

The (Template) Version is added for backwards compatibility to the record file only

- Template Version = V1

2.2.4 Using JSON Schema

This subsection lists some of the main tasks and procedures to interact with the DSB system

1. User needs to obtain the up-to-date JSON schema
 - I. Users can download JSON schema using REST API (see [Obtain JSON schemas](#))
 - II. JSON schema are available to download from the DSB web-site (in the file-download section)
 - III. JSON schema are also available through GitHub in: <https://github.com/ANNA-DSB/Product-Definitions/tree/master/UAT>
2. To request for a new ISIN, user needs to:
 - I. Select a 'Request' JSON schema which matches their needs. Each Request JSON Schema can be uniquely identified by four attributes:
 - Asset Class
 - Instrument Type
 - Product (Use Case)
 - Level
 - II. Format the required instrument and supply its attributes as a valid *JSON request for an instrument record* based on the Request JSON Schema
 - III. Embed the *JSON request for an instrument record* within a POST /records (see [Request the ISIN for a financial instrument](#)) method
3. In case the responseCode is continue (100) or successful (200) the response contains a *JSON instrument record*
4. To parse the *JSON instrument record*, the user needs to:
 - I. Extract the TemplateVersion attribute and Header which contains:
 - Asset Class
 - Instrument Type
 - Product (Use Case)
 - Level
 - II. Select the relevant JSON record schema that matches above attribute
 - III. Continue to parse additional attributes in the *JSON instrument record* as it must be a valid record based on the schema found above

3 REST API methods

3.1 Introduction

This section looks at the methods that will be supported by DSB REST API.

All the DSB REST methods are HTTP based RESTful methods that adhere to the REST architectural constraints.

The HTTP Status codes in the response adhere to the standard HTTP/1.1 ([RFC 7231](https://tools.ietf.org/html/rfc7231)).

3.2 Obtain JSON schemas

To obtain the JSON schemas user needs to invoke `GET /schemas?names` that will return the list of schema names. Using each of these names the user may invoke `GET /schemas?schemaName=<NAME>` that will return the JSON schema for that name.

The following diagram illustrates the workflow:

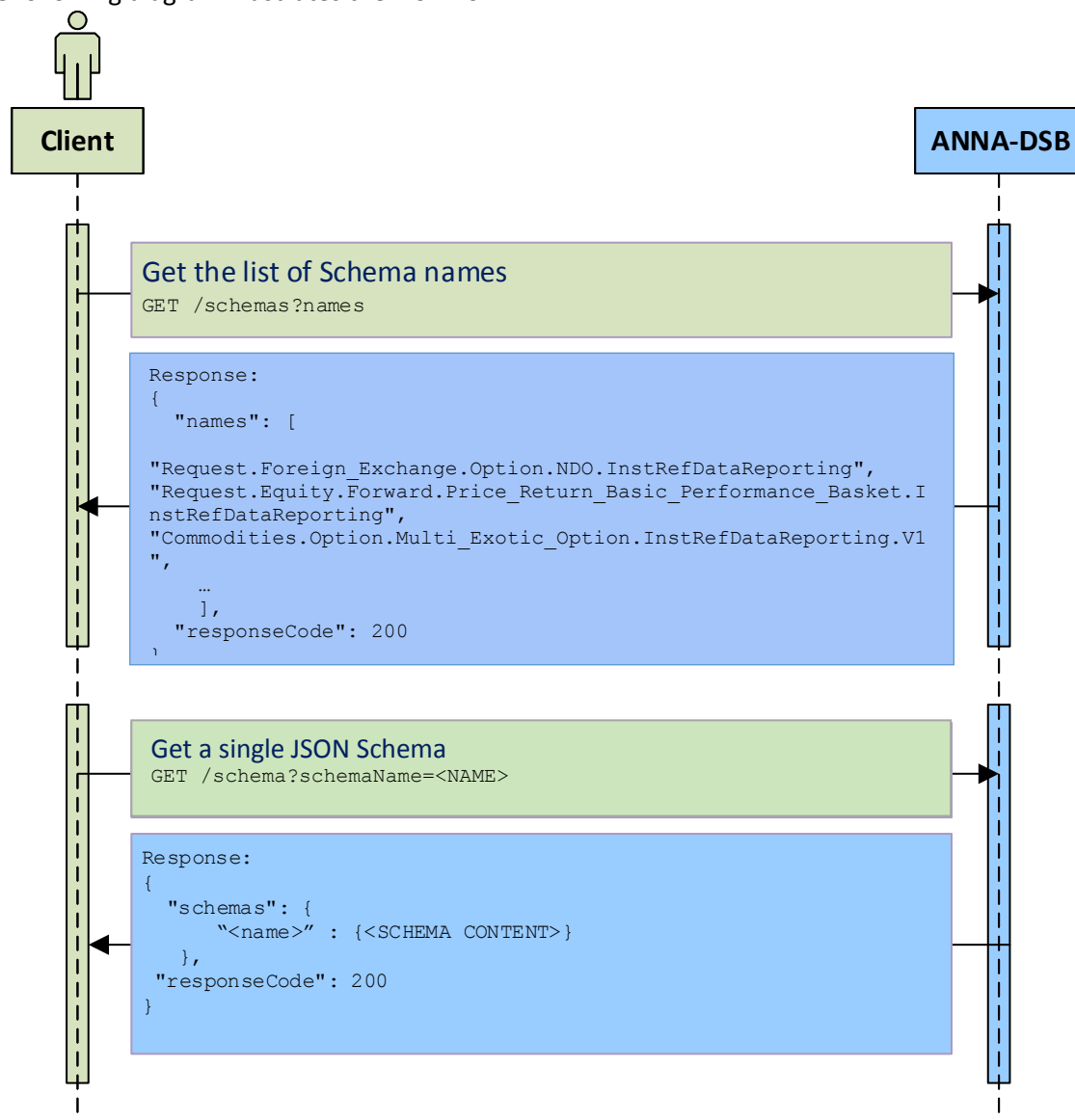


Diagram 2: Obtain JSON schema

3.3 Retrieve or Create ISIN Record by Attribute

The `POST /records` method is used to create the ISIN record for a financial instrument or return the existing record if the ISIN record already exists. The financial instrument is defined by a *JSON instrument request* object provided in the “record” within the body of the request.

The input JSON record must contain a “record” component where the financial instrument is provided. The input may also contain a “requestContext” which will be echoed back in the response and may help users to identify their request.

The REST API service will first search for an existing ISIN that has identical attributes as the record (up to normalization, see section 6 in: [DSB UAT Product Definition.pdf](#)) and if such an ISIN exists, the result record will contain the ISIN.

If such an ISIN does not exist, the DSB will allocate a new ISIN for this request.

In both cases, provided the request is valid, the resulting record will contain an ISIN. The client will not be able to tell if the ISIN was just allocated or if it existed prior to this call.

The following diagram illustrates the workflow:

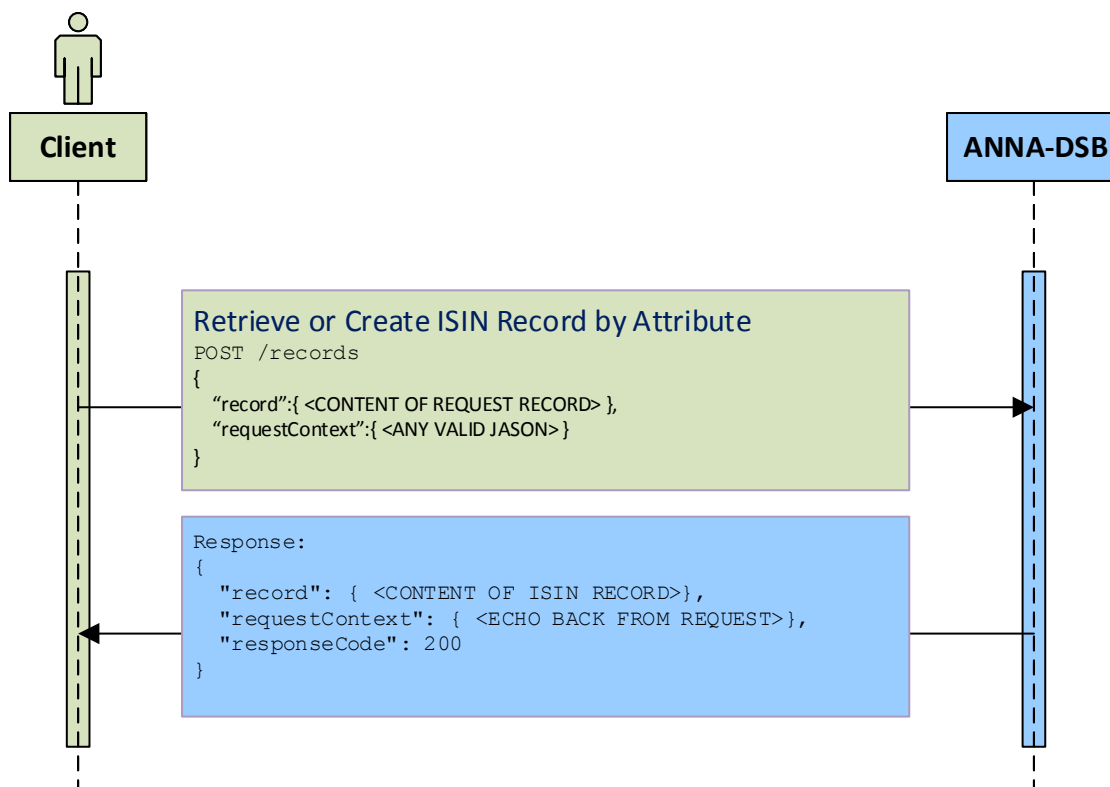


Diagram 3: Retrieve or Create ISIN Record by Attribute

3.4 Retrieve ISIN Record by Attribute

The `POST /records` method that is described in the previous section, can be used with additional parameter: `POST /records?create=false`

The additional parameter: `create=false` guarantees that no new record will be added to the system.

For the input JSON record see the previous section.

The REST API service will search for an existing ISIN that has identical attributes as the supplied record (up to normalization, see section 6 in: [DSB UAT Product Definition.pdf](#)) and if such an ISIN exists, the result record will contain the ISIN record.

If such an ISIN does not exist, the DSB will not allocate a new ISIN, instead the system will calculate all derived values and the output record will contain all attributes except of the ISIN value.

This method can be used to resolve the derived attributes (e.g. CFI code, and FISN) of the record without the need to create the associated ISIN.

The following diagram illustrates the workflow:

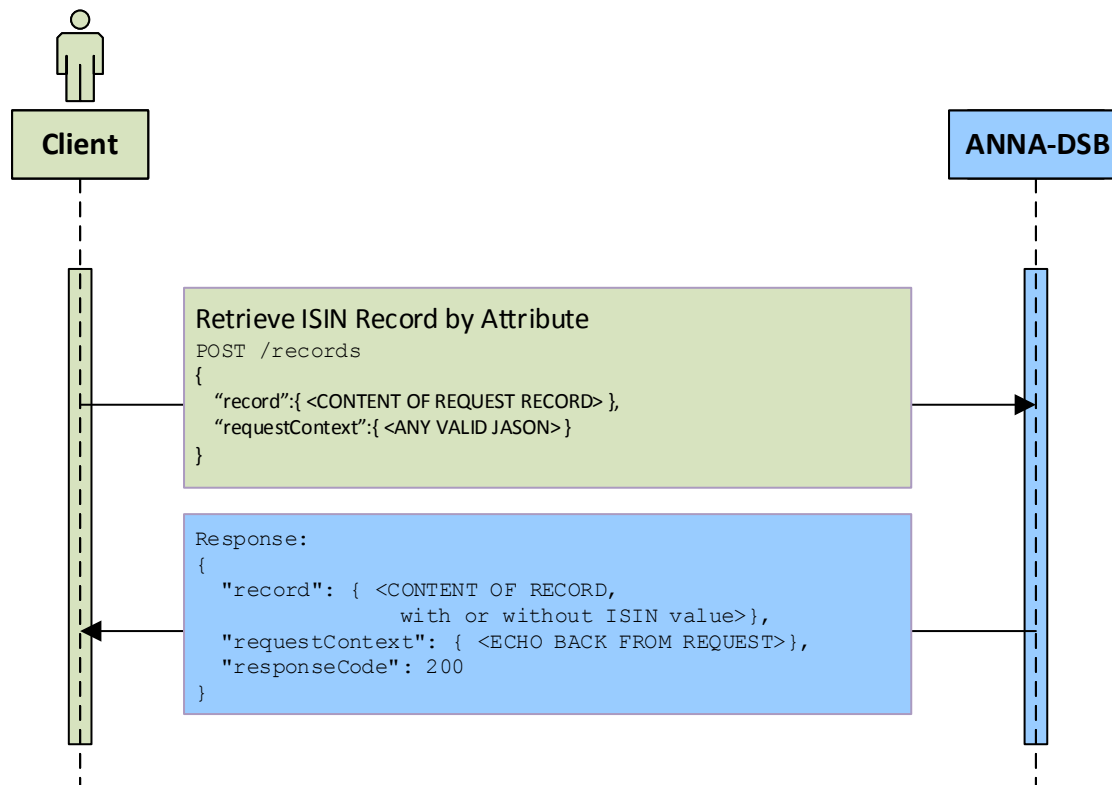


Diagram 4: Retrieve ISIN Record by Attribute

3.5 Retrieve ISIN Record by ISIN

The `GET /records/<ISIN>` method is used to return the ISIN record for an existing ISIN. The ISIN is specified in `<ISIN>`.

The following diagram illustrates the workflow:

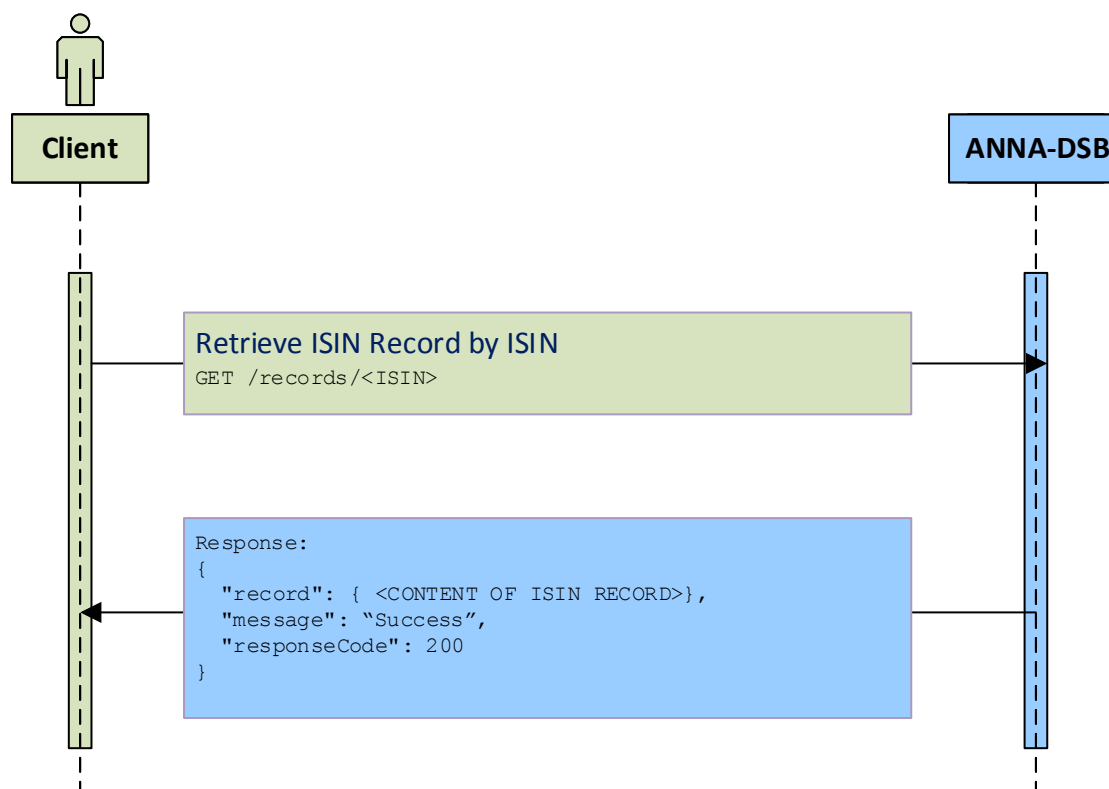


Diagram 5: Retrieve ISIN Record by ISIN

3.6 Search for ISIN Records by attributes

The GET /search/ method can be used to search for ISIN records that match the supplied criteria.

The user must specify the following arguments:

- query: (i.e. a query string). Please refer to DSB Search Query Syntax document.
- pageSize: which is the maximum number of results that are returned in each page
- pageNum: see explanation below
- requestContext: a valid JSON record that may help to identify the query

To retrieve the first 100 records, the user should specify:

- "pageSize" : 100
- "pageNum" : 1

To retrieve the next 100 records, the user should invoke the GET /search/ method again, with the same query but using:

- "pageSize" : 100
- "pageNum" : 2

A valid response contains all the above arguments (in a JSON record) together with:

- Records: which contains the result records
- totalResults: The number of total records that matches the query criteria
- responseCode: Matches the HTTP status code

The following diagram illustrates the workflow:

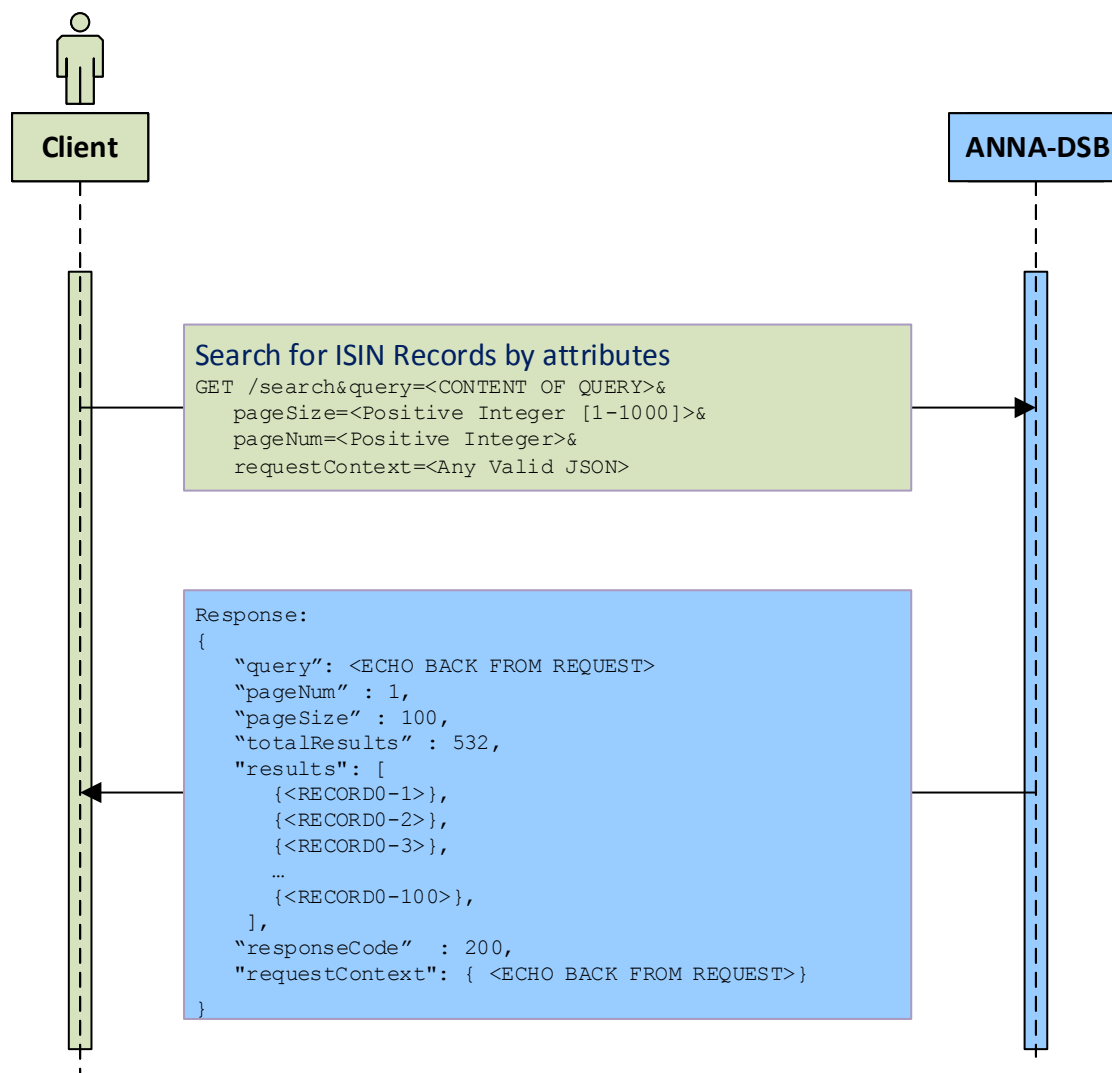


Diagram 6: Search for ISIN Records by attributes

3.7 Download historical ISINs

The `GET /file-download/isin/<YYYYMMDD>/<ASSET-CLASS>/<ASSET-CLASS>-<YYYYMMDD>.records` method is used to download historical ISINs.

The `<YYYYMMDD>` stands for a valid date when ISINs were created.

`<ASSET-CLASS>` is one of:

- Credit
- Commodities
- Equity
- Foreign_Exchange
- Rates

The response contains instrument records separated by newline ('\n') that were either allocated or modified on the `YYYYMMDD` date. Each record is a valid JSON instrument record.

See section 4.6 below for the base URL of this method.

The following diagram illustrates the workflow:

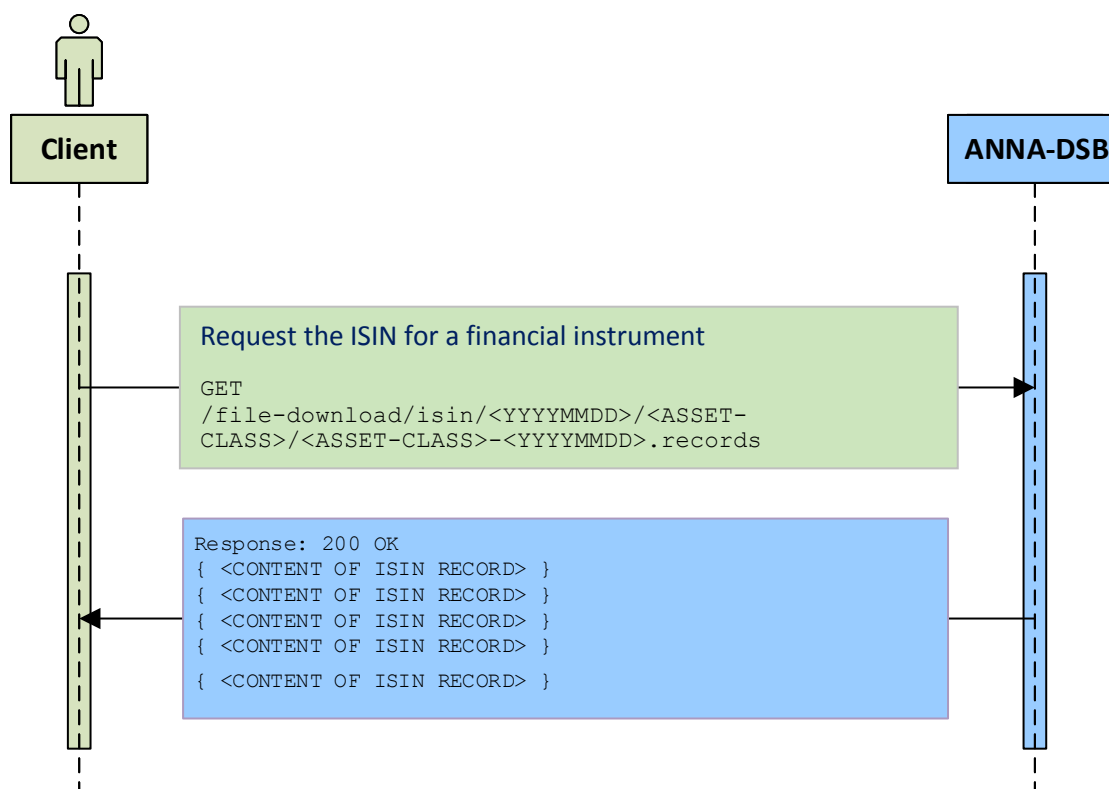


Diagram 7: Download historical ISINs

4 Methods Reference

4.1 Introduction

This section details the REST API methods that are used by DSB Web service.

4.2 Obtain schema names

Method	Request	Required authorization	
GET	/schemas	✖	
Request arguments			
Argument name	Datatype	Required	Note
names	-	✓	No content
Response arguments			
Output JSON record components	Datatype	Required	Note
responseCode	Integer	✓	Matches the HTTP status code: success = 200 Error code = 4XX or 5XX
message	String	✖	Available only if response code indicates an error Contains indicative error message
names	Array of strings	✖	Conditionally required if responseCode=200 Each string indicates a single schema name

4.3 Obtain single schema

Method	Request	Required authorization	
GET	/schemas	✖	
Request arguments			
Argument name	Datatype	Required	Note
schemaName	String	✓	Content is expected to be a valid schema name
Response arguments			
Output JSON record components	Datatype	Required	Note
responseCode	Integer	✓	Matches the HTTP status code: success = 200 Schema Name does not exist = 404 Error code = 4XX or 5XX
message	String	✖	Available only if response code indicates an error Contains indicative error message
schemas	JSON record	✖	Conditionally required if responseCode=200 Contains the name of the schema (as key) and the JSON schema as value.

4.4 Retrieve or Create ISIN Record by Attribute

Retrieve ISIN Record by Attribute

Method	Request	Required authorization	
POST	/records	✓	
Request arguments			
Argument name	Datatype	Required	Note
create	Boolean	✖	The default is: create=true When create=false, the system will not allocate a new ISIN, and the output will not include ISIN.
Input JSON record components	Datatype	Required	Note
record	JSON record	✓	Content of an <i>Instrument Request</i> JSON record which is valid to an <i>Instrument Request schema</i>
requestContext	JSON record	✖	User may place here any information that will help to identify the request
Response arguments			
Output JSON record components	Datatype	Required	Note
responseCode	Integer	✓	Matches the HTTP status code: success = 200 Authorization failed: 401 Invalid record (invalid per schema): 400 Invalid JSON syntax: 500
Message	String	✖	Available only if response code indicates an error Contains indicative error message
Record	JSON record	✖	Available is responseCode=200 Contains a valid <i>instrument JSON record</i> .
requestContext	JSON record	✖	Echo the user input Conditionally required if requestContext was provided and responseCode is one of: - 200 (Success) - 400 (invalid record) - 401 (authorization failed) Note: requestContext is not available if responseCode is 500

4.5 Retrieve ISIN Record by ISIN

Method	Request	Required authorization	
GET	/records/<ISIN>	✓	
Request arguments - none			
Response arguments			
Output JSON record components	Datatype	Required	Note
responseCode	Integer	✓	Matches the HTTP status code: success = 200 Authorization failed: 401 No such ISIN: 404
message	String	✗	Contains indicative error message Contains: “Success” in case responseCode = 200
record	JSON record	✗	Conditionally required if responseCode=200 Contains a valid <i>instrument JSON record</i> .

4.6 Search for ISIN Records by attributes

Method	Request	Required authorization	
GET	/search	✓	
Request arguments			
Argument name	Datatype	Required	Note
query	String	✓	Query string. refer to: DSB Search Query Syntax document.
pageSize	Positive Integer	✗	The maximum number of records that return in each page Default and the maximum values are 1000
pageNum	Positive Integer	✗	The page number Default to 1
requestContext	JSON record	✗	User may place here any information that will help to identify the request
Response arguments			
Output JSON record components	Datatype	Required	Note
responseCode	Integer	✓	Matches the HTTP status code: success = 200 pageSize exceeds maximum = 403 Note: The responseCode does not indicate the case where user is not authorized, instead, in such a case, the totalResults is 0.
message	String	✗	Available only if response code indicates an error Contains indicative error message
totalResults	Integer	✗	Conditionally required if responseCode=200 Indicate the total number of records that match the query
query	String	✗	Echo the user input Conditionally required if responseCode=200
pageSize	Positive Integer	✗	Echo the user input Conditionally required if responseCode=200
pageNum	Positive Integer	✗	Echo the user input Conditionally required if responseCode=200
records	Array of JSON record	✗	Conditionally required if responseCode=200 Contains a valid <i>instrument JSON records</i> that matches the query.
requestContext	JSON record	✗	Echo the user input Conditionally required if requestContext was provided and responseCode is 200 (Success)

4.7 Download Historical ISINs

Method	Request	Required authorization
GET	/file-download/isin/<YYYYMMDD>/<ASSET-CLASS>/<ASSET-CLASS>-<YYYYMMDD>.records	✓
Request notes		
<p>YYYYMMDD – stands for a valid date</p> <p>Asset Class – is one of:</p> <ul style="list-style-type: none"> • Credit • Commodities • Equity • Foreign_Exchange • Rates <p>The base url for this method is:</p> <ul style="list-style-type: none"> • Production environment: https://www.anna-dsb.com • UAT environment: Error! Hyperlink reference not valid. 		
Response notes		
<p>The response is a content of file which contains ISIN records</p> <p>The records are separated by new line</p>		

5 REST Methods Samples

5.1 Introduction

This section contains REST method samples.

Some of the responses are **<truncated>**

5.2 Obtain schema names

GET	/schema?names
Response	
Response: <pre> { "names": ["Request.Foreign_Exchange.Option.NDO.InstRefDataReporting", "Request.Equity.Forward.Price_Return_Basic_Performance_Basket.InstRefDataReporting", "Commodities.Option.Multi_Exotic_Option.InstRefDataReporting.V1", <truncated> "Request.Rates.Swap.Inflation_Fixed_Float_YoY.InstRefDataReporting"], "responseCode": 200 }</pre>	

5.3 Obtain single schema

GET	/schemas?schemaName=<NAME>
GET	/schemas?schemaName=Credit.Swap.ABS.InstRefDataReporting.V1
Response	
<pre> { "schemas": { "Credit.Swap.ABS.InstRefDataReporting.V1": { "\$schema": "http://json-schema.org/draft-04/schema#", "title": "Credit.Swap.ABS.InstRefDataReporting.V1", "LastModifyDateTime": "2017-08-16T17:47:37", "description": "Record template for Credit.Swap.ABS", <truncated> }, "responseCode": 200 } }</pre>	

5.4 Retrieve or Create ISIN Record by Attribute

POST	/records
<pre>POST /records { "record": { "Header": { "AssetClass": "Foreign_Exchange", "InstrumentType": "Forward", "UseCase": "NDF", "Level": "InstRefDataReporting" }, "Attributes": { "NotionalCurrency": "EUR", "ExpiryDate": "2033-06-13", "OtherNotionalCurrency": "USD", "SettlementCurrency": "JMD" } }, "requestContext": { "user": "username@email.com", "requestID": "ID1", "additional information": "blob" } }</pre>	
Response	
<pre>{ "record": { "Header": { <truncated> }, "Attributes": { <truncated> }, "ISIN": { "ISIN": "EZ6LSHR0B9D5", "Status": "New", "StatusReason": "", "LastUpdateDateTime": "2017-08-30T11:44:13" <truncated> }, "TemplateVersion": 1, "Derived": { <truncated> } }, "responseCode": 200, "requestContext": { "user": "username@email.com", "requestID": "ID1", "additional information": "blob" } }</pre>	

5.5 Retrieve ISIN Record by Attribute

POST /records

POST /records?create=false

```
{
  "record": {
    "Header": {
      "AssetClass": "Foreign_Exchange",
      "InstrumentType": "Forward",
      "UseCase": "NDF",
      "Level": "InstRefDataReporting"
    },
    "Attributes": {
      "NotionalCurrency": "EUR",
      "ExpiryDate": "2055-06-13",
      "OtherNotionalCurrency": "USD",
      "SettlementCurrency": "JMD"
    }
  },
  "requestContext": {
    "user": "username@email.com",
    "requestID": "ID1",
    "additional information": "blob"
  }
}
```

Response

```
{
  "record": {
    "Header": {
      <truncated>
    },
    "Attributes": {
      <truncated>
    },
    "ISIN": {
      "ISIN": "",
      "Status": "New",
      "StatusReason": "",
      "LastUpdateDateTime": "2017-08-30T11:56:19"
      <truncated>
    },
    "TemplateVersion": 1,
    "Derived": {
      <truncated>
    }
  },
  "responseCode": 200,
  "requestContext": {
    "user": "username@email.com",
    "requestID": "ID1",
    "additional information": "blob"
  }
}
```

5.6 Retrieve ISIN Record by ISIN

GET	/records/<ISIN>
GET	/records/EZ6LSHR0B9D5
Response	
<pre>{ "record": { "Header": { "AssetClass": "Foreign_Exchange", "InstrumentType": "Forward", "UseCase": "NDF", "Level": "InstRefDataReporting" }, "Attributes": { "NotionalCurrency": "EUR", "ExpiryDate": "2033-06-13", "OtherNotionalCurrency": "USD", "SettlementCurrency": "JMD", "DeliveryType": "CASH", "PriceMultiplier": 1 }, "ISIN": { "ISIN": "EZ6LSHR0B9D5", "Status": "New", "StatusReason": "", "LastUpdateDateTime": "2017-08-30T11:44:13" }, "TemplateVersion": 1, "Derived": { "IssuerorOperatoroftheTradingVenueIdentifier": "NA", "CommodityDerivativeIndicator": "FALSE", "FXType": "FXMJ", "UnderlyingAssetType": "Spot", "ReturnorPayoutTrigger": "Forward price of underlying instrument", "FullName": "Foreign_Exchange Forward NDF EUR USD 20330613", "ShortName": "NA/Fwd NDF EUR USD 20330613", "ClassificationType": "JFTXFC" } }, "responseCode": 200, "message": "Success" }</pre>	

5.7 Search for ISIN Records by attributes

Search for Rates instruments that contains Libor

GET	/records
GET /search?query=LIBOR AND Rates&pageSize=5&pageNum=3&requestContext={"requestID":"ID1"}	
Response	
<pre>{ "query": "LIBOR AND Rates", "pageNum": 3, "pageSize": 5, "totalResults": 59, "records": [{ <truncated> }, { <truncated> }, { <truncated> }, { <truncated> }, { <truncated> },], "responseCode": 200, "requestContext": { "requestID": "ID1" } }</pre>	

5.8 Download Historical ISINs

GET /file-download/isin/<YYYYMMDD>/<ASSET-CLASS>/<AssetClass>-<YYYYMMDD>.records

GET /file-download/isin/20170829/Rates/Rates-20170829.records

Response

```
<truncated>
<truncated>
<truncated>
<truncated>
...
<truncated>
```

About Derivatives Service Bureau (DSB)

The Association of National Numbering Agencies (“ANNA”), is founding the Derivatives Service Bureau (DSB), for the issuance and maintenance of International Securities Identification Numbers (ISINs) for OTC Derivatives. The DSB will rely on an automated platform capable of allocating ISINs in near real-time.