

# Discovery Symbology API User Guide

V9

October 2023

User guide for discovery/symbology/v1 API



# Contents

<b>About the Discovery Symbology API</b> .....	<b>4</b>
<b>Access to Discovery Symbology API</b> .....	<b>4</b>
<b>How to use the Discovery Symbology API</b> .....	<b>5</b>
Background .....	5
Query Parameters .....	5
Query Fields.....	7
Types of navigation .....	8
Auto .....	8
Strict .....	10
Predefined .....	10
Format quick-help matrix, for auto and strict type queries.....	11
Auto type query: .....	11
Strict type query.....	12
Parameter Requirements for Predefined type queries .....	13
FindESGStatementParent .....	13
Equity Ticker + Country to RIC .....	13
Ticker + MIC to Organization .....	14
ISIN + Venue Identifier to Quote.....	15
FindPrimaryRIC .....	16
General information about query structure .....	16
objectTypes and Aliases .....	17
Reference parameter .....	17
Active Filter .....	18
objectTypes when navigating 'to' a PermID with an auto query .....	18
Pagination and dynamic pagination .....	19
Health check endpoint.....	20
Lookup-type endpoint.....	20
Endpoints .....	21
<b>Example Queries for /lookup endpoint</b> .....	<b>22</b>
Auto route query examples .....	22
Strict query examples.....	23
Predefined query examples.....	26
Point in Time and History navigation examples.....	28
<b>Example queries for /lookup-type endpoint</b> .....	<b>29</b>
<b>Customer Responsibilities</b> .....	<b>29</b>
<b>Error messages</b> .....	<b>30</b>
<b>List of objectTypes</b> .....	<b>35</b>
<b>List of relationships</b> .....	<b>36</b>

**Identifier types covered by asset class .....37**  
Change Notifications channels .....41  
Your Personal Information.....41

## About the Discovery Symbology API

For the internal and external data user who needs to understand and connect data, discovery/symbology/v1 is a linked data service that allows data mapping and navigation across the LSEG information model to enable easier access to data and building of new data workflows.

This API is a request/response service to assist users with concordance, navigation and discovery of content. The API enables customers to map from identifiers to a PermID and to navigate between identifiers.

- Map – map from an identifier to the corresponding PermID for that entity. For example, mapping an ISIN to an Instrument PermID, or mapping a Legal Entity Identifier (LEI) to an Organization PermID. PermIDs are an important key to interoperability across many products.
- Navigate – navigate between user selected start and end point identifiers (e.g. RIC to ISIN) using relationships to navigate from one entity to another. Enables content sets using different identifiers to be easily connected

The Discovery Symbology API supports a wide range of market and LSEG entity identifiers across multiple asset classes. A full list of the supported identifiers can be found in the Appendix A .

The service is available at no additional cost to all customers subscribing to content services through the data platform, it is not available on a stand-alone basis.

## Access to Discovery Symbology API

The API Playground is where you can experiment with the Discovery Symbology API, as well as access documentation.

Access to this site is available via a valid username and password, which you can obtain from your account manager.

This site is best viewed using Google Chrome Frame.

Access the Discovery Symbology API:

1. Open the [API Playground](#) in your Browser.
2. In the **Search (Filter)** field, start typing “symbology”.  
As you type, the **ALL APIs** field below displays the matching API results.
3. Select the **/discovery/symbology/v1** option.  
The available API endpoints details display in the main field.
4. Select any endpoint details box.  
The endpoint's details display in the main field.

# How to use the Discovery Symbology API

## Background

LSEG content for organizations, instruments, quotes is organized into objects and hierarchical relationships. Each object (e.g. an organization, instrument, fund, quote, trading venue) has an object Type associated with it and a PermID that uniquely identifies it.

Objects are related to each other via relationships, for example a company and an instrument can be linked by the IssuedBy relationship. Objects also have identifiers linked to them, for example an organisation can have Legal Entity Identifier.

The Discovery Symbology API allows the user to navigate between symbols for the same entity (e.g. from ISIN to CUSIP, both are identifiers of the same instrument) as well as using relationships to connect entities together and navigate from one to another (e.g. from ISIN to LEI, ISIN is instrument level and LEI is organization level). Thus, it provides intelligent navigation through the graph of relationships and identifiers that make up the core of the information model.

## Query Parameters

Below is a table of the parameters used

Parameter	Type	Fields Accepted	Mandatory/Optional	Function	Helpful tips
from	[array]	identifierTypes objectTypes values	Mandatory	'from' indicates the starting point of the request and contains 3 elements:  The array of values is mandatory, and the identifierTypes and objectTypes are optional.  If the 'from' is a PermID either identifierTypes or objectTypes must be specified.	Providing identifierTypes or objectTypes makes the query more efficient  Users can specify multiple values, IdentifierTypes or ObjectTypes in the query  Example queries in section 5 of this guide and on the API Playground
to	[array]	identifierTypes values objectTypes	Mandatory	'to' indicates the required result of the request.  The identifierTypes is mandatory,  objectTypes is optional unless the query type is auto and the To is a PermID	PermID can not be combined with other identifierTypes in one array.
path	[array]	relationshipTypes objectTypes identifierTypes values	Optional	'path' – is needed for specifying relationships in one-hop or multi-hop 'strict' type queries  Path contains information about the	The path parameter is not required if the type "auto" is used.

				relationship name and which objectTypes are being navigated	In a "strict" type query path is mandatory if a relationship traversal is required in the navigation
type	"string"	Accepted values: "auto" "strict" "predefined"	Mandatory	Indicates the type of navigation query: auto, strict or predefined	Predefined are accompanied by a route name which must be used in the query
effectiveAt	"string"		Optional	Is used to provide a 'point in time' response based on current value at the date/time specified by the user.  By default a current time will be applied.	Format should be ISO 8601 without offsets in UTC: yyyy-MM-ddTHH:mm:ss.msZ.  For example 2010-07-07T12:34:33.100Z
showHistory	operator	"showHistory" : true	Optional	Is used to provide the effectiveFrom and effectiveTo date relationship history between the From and To identifiers through time.  Effective From/To shows the date/time at which each of the outputs in the response are effective from/to. E.g. ISIN GB00B012T521 is linked to RIC IRP.L from Jun 2004 to Apr 2013.	Note the effective From and To dates in the response are the effective dates of the relationship between the entities linked to the identifiers, not the effective date of the identifiers themselves.  If reference parameter is included in the query, showHistory will also show the effective data history of the reference attributes where available.
reference	operator	"reference": ["name", "status", "classification", "PermID"]	Optional	Allows the user to add basic entity information to the query response to aid understanding and disambiguate results.	More info in Reference Parameter section of document.  The system makes a secondary internal call to get the reference information, reference parameter is best omitted for optimal performance once queries are operationalized.
filter	operator	"filter": {"status": "active"}	Optional	Allows the user to specify whether results are filtered based on whether the related entity is active	More info in the Active Filter section of the document

## Query Fields

Field	Field type	Mandatory/Optional	Function	Helpful tips
identifierTypes	[array]	Optional in 'from', Mandatory in 'to'	This indicates the type of identifier used in the Symbology API response service. Identifier types can be either public or Refinitiv identifier types of a Refinitiv PermID. E.g., ISIN, SEDOL, RIC etc.	Identifier types are string fields and should be enclosed within quotes  Any case (lower, upper or mixed) is accepted for the value.  User can specify a single or multiple identifierType names, or use the alias 'ANY'.
objectTypes	[array]	Optional unless identifierTypes is a PermID	This indicates a particular object type or type of entity associated with the input values in the Symbology API response service.  See Appendix A for list of available objectTypes.	In case the objectTypes is not known or user wants to include all possible objectTypes, user can provide the keyword 'ANY'  Since object type / identifier type in 'from' are optional parameters, if this field is empty - "objectTypes": [] or "objectTypes": [""] - the expected behavior is to resolve the request to all possible navigation options  Any cases (lower, upper or mixed) is accepted for the value, however field name should be an exact match  If 'to' identifierTypes is a PermID, then objectTypes must be specified to ensure the correct level of PermID is returned e.g. Organization, AnyInstrument, AnyQuote, AnyVehicle, MarketAttributableSource
values	[array]	Mandatory in From	The value provided by the user. It is mandatory in the 'From' part of the query. Multiple values can be provided in one query. Values are not case-sensitive.	Field name should be an exact match.  Leading and trailing blanks are ignored

## Types of navigation

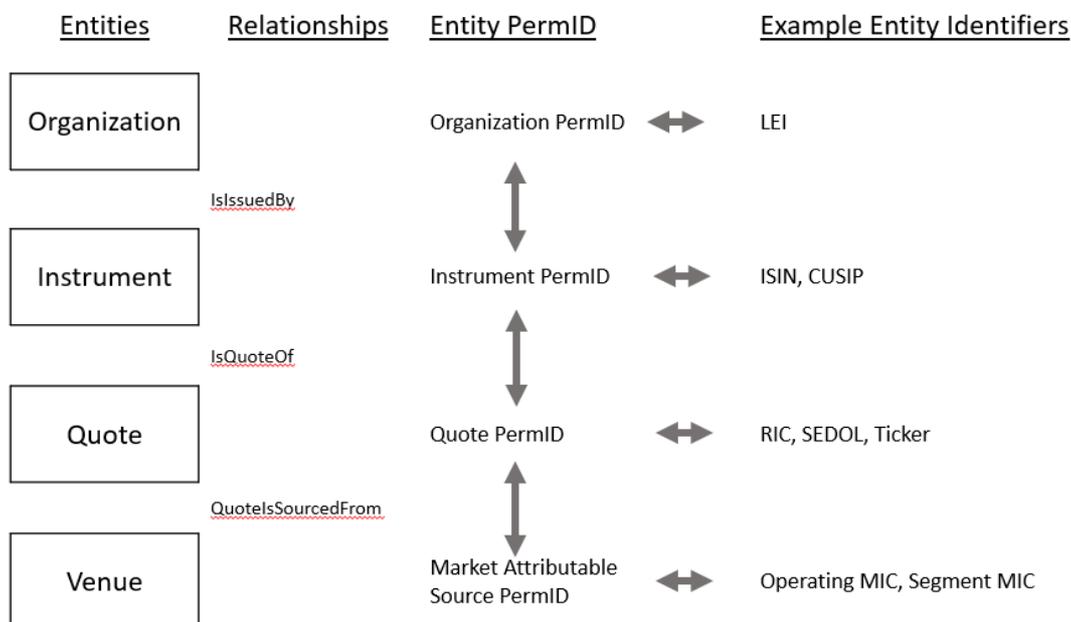
Three types of navigation are supported. This is set in the 'type' parameter of the query.

### Auto

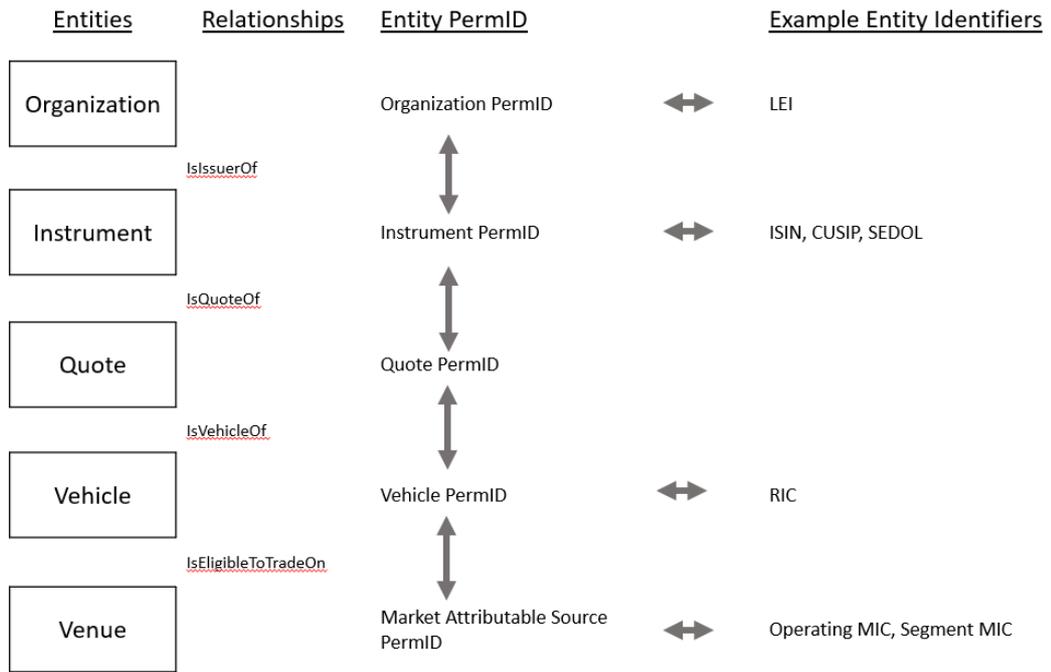
Auto query type allows users to navigate between the relationships linking an issuer to its issued instruments, the quotes of the instruments and the venues of the quotes, without the need to specify the relationships that link the entities together. This simplifies the query structure for many common navigation use cases.

When the query 'type' is set to 'auto' the API will utilize specific relationships to perform the navigation between the issuer – instrument – quote - venue object types.

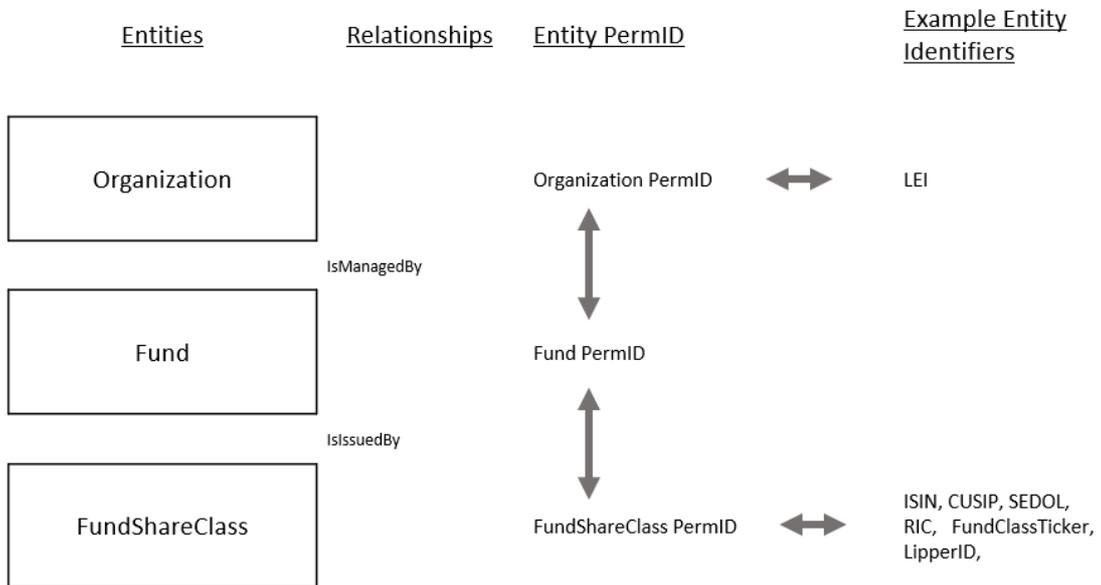
For equity issuers, instrument and quote these relationships are used:



For Fixed Income issuers, instruments and quotes the below relationships are used in auto queries



For Funds the below entities and relationships are used in auto route queries



## Strict

Strict navigation allows a user to specify the relationship to be navigated using a 'path' parameter. This will constrain the navigation between entities to only those relationships specified.

The most commonly used relationships are given in Appendix B and there are examples of strict queries in Section 5

Strict Queries will navigate 'at level' if no relationships are specified e.g. Instrument PermID to 'any' identifier in a strict query with no relationships specified will navigate only to the instrument level identifiers like ISIN, CUSIP, but not quote level identifiers like RIC.

The user can chain together multiple relationship 'hops' to traverse from one entity to the next in the same query. Examples of one-hop and two-hop queries are also in Section 5

Relationships are one-way so e.g. the 'IssuedBy' relationship connects an exchange traded instrument to an organization. To navigate the relationship in reverse, prefix the relationship with 'inverse' for example 'InverseIssuedBy'.

## Predefined

Predefined Routes enable specific navigations that can't be achieved through auto or strict query Types.

These are the currently supported Predefined routes:

Route Name	Purpose
FindESGStatementParent	Enables the user to navigate from a corporate bond instrument identifier to most relevant organisation that has Refinitiv ESG data available. (Only available to customers with Sustainable Finance and Investment related products)
EquityTickerCountryCodetoPrimaryRIC	Provide input pair of equity or EFT ticker and ISO2 country code and output the main RIC code at the intersection
ExchangeTickerMicToOrganization	Provide input pair of equity or EFT ticker and Segment MIC code and output the organization PermID or other organization identifier at the intersection
IsinVenueToQuote	Provide input pair of ISIN and venue identifier and output the quote. Venue identifier can be MIC, RDNExchangeCode, Price Source Code (PSC) to MarketAttributableSource PermID
FindPrimaryRIC	Navigate from an organization or instrument identifier, or SEDOL, to the main Quote/RIC for that Organization/Instrument

The allowed parameters for predefined queries can be found in the 'Parameter Requirements for Predefined type queries' section of the document. And practical examples of these predefined queries can be found in section 5 of the document.

## Format quick-help matrix, for auto and strict type queries

The tables below show more detail about what parameters are optional/mandatory depending on whether a strict or auto query type is used, and depending on whether the navigation is from or to a PermID.

### Auto type query:

		Navigating To ...		
		PermID	ObjectType	Identifier
Navigating From...	PermID	<p><u>FROM</u>                      value-mandatory                      objecttype - optional (objecttype alias can be used)                      identiertype(PermID) - can be specified (recommended)</p> <p><u>TO</u>                      objecttype – strongly recommended (multiple objects can be provided)                      identiertype - mandatory</p>	<p><u>FROM</u>                      value-mandatory                      objecttype - optional (objecttype alias can be used)                      identiertype(PermID) - can be specified (recommended)</p> <p><u>TO</u>                      objecttype - mandatory (multiple objects can be provided)                      identiertype(PermID) - mandatory</p>	<p><u>FROM Section</u>                      value-mandatory                      objecttype - optional (objecttype alias can be used including 'any')                      identiertype(PermID) - can be specified (recommended)</p> <p><u>TO Section</u>                      objecttype- cannot be provided with identifier type other than perm id                      identiertype - mandatory  <u>Note:</u> identifier type 'any' cannot be used in auto query</p>
	Identifier	<p><u>FROM</u>                      value-mandatory                      objecttype - optional (a broad alias such as objecttype 'any' cannot be used)                      identiertype - can be specified (recommended)</p> <p><u>TO</u>                      objecttype - mandatory (multiple objects or alias 'anyinstrument' can be provided)                      identiertype - mandatory</p>	<p><u>FROM</u>                      value-mandatory                      objecttype - optional                      identiertype - can be specified (recommended)</p> <p><u>TO</u>                      objecttype - mandatory (multiple objects or alias such as 'anyinstrument' can be provided)                      identiertype (PermID) - mandatory</p>	<p><u>FROM</u>                      value-mandatory                      objecttype - optional (a broad alias such as objecttype 'any' cannot be used)                      identiertype - can be specified (recommended)</p> <p><u>TO</u>                      objecttype- cannot be provided with identifier type other than perm id                      identiertype - mandatory</p>
	ObjectType	<p><u>FROM Section</u>                      value-mandatory                      objecttype - provided                      identiertype – optional</p> <p><u>TO Section</u>                      objecttype - mandatory (multiple objects can be provided)                      identiertype (PermID) - mandatory</p>	<p><u>FROM Section</u>                      value-mandatory                      objecttype - provided                      identiertype - optional  <u>Note:</u> if object type is provided identiertype will be a Perm ID</p> <p><u>TO Section</u>                      objecttype - mandatory (multiple objects or alias such as 'anyinstrument' can be provided)                      identiertype (PermID) - mandatory</p>	<p><u>FROM</u>                      value-mandatory                      objecttype - provided (objecttype alias can be used including 'any')                      identiertype - can be specified (recommended)</p> <p><u>TO</u>                      objecttype- cannot be provided with identifier type other than perm id                      identiertype - mandatory</p>

## Strict type query

		Navigating To ...		
		PermID	ObjectType	Identifier
Navigating From ...	PermID	<u>FROM Section</u> value-mandatory objecttype - optional identiertype(PermID) - can be specified (recommended)	<u>FROM Section</u> value-mandatory objecttype - optional identiertype(PermID) - can be specified (recommended)	<u>FROM Section</u> value-mandatory objecttype - optional identiertype(PermID) - can be specified (recommended)
	Identifier	<u>TO Section</u> Either one can be specified (one of them is mandatory) objecttype - optional identiertype (PermID)	<u>TO Section</u> Either one can be specified objecttype identiertype (PermID)- optional	<u>TO Section</u> objecttype - only provided in case of a 'path' identiertype - mandatory <u>Note:</u> Identiertype 'any' can be used <u>Note:</u> intermediate path 'objects' are mandatory except the last one
	ObjectType	<u>FROM Section</u> value-mandatory objecttype - provided identiertype (PermID) - optional identiertype (other than PermID) – mandatory	<u>FROM Section</u> value-mandatory objecttype - provided identiertype (PermID) - optional identiertype (other than PermID) – mandatory	<u>FROM Section</u> value-mandatory objecttype - provided identiertype (PermID) - optional identiertype (other than PermID) – mandatory
		<u>TO Section</u> Either one can be specified (one of them is mandatory) objecttype identiertype (PermID) <u>Note:</u> In this case identifier type will be a PermId even when not specified	<u>TO Section</u> Either one can be specified (one of them is mandatory) objecttype identiertype (PermID) <u>Note:</u> In this case identifier type will be a PermId even when not specified	<u>TO Section</u> objecttype identiertype - mandatory

Note, if a user provides a **mix** of PermIDs and Identifiers in the From input array, then the objectType parameter should not be included (even objectType=ANY) else the inputs that are identifiers will be ignored.

## Parameter Requirements for Predefined type queries

Predefined type queries perform specific navigations that can't be achieved through auto or strict query types. Predefined queries have limitations on how the parameters can be used and some values are pre-set. Example predefined type queries can be found in section 5.

### FindESGStatementParent

Map a corporate bond identifier (e.g. ISIN) to related 'ESGStatementParent' organisation. (Only available to customer subscribing to Sustainable Finance related products)

Query Parameter	Mandatory/Optional	Comment
From	mandatory	The from value will be an instrument identifier for a corporate bond e.g. ISIN, CUSIP, SEDOL, Instrument PermID
To	optional	The To parameter is optional and if omitted by default the output the Organization PermID of the ESG Statement Parent Organization. If included the To can only be an organization identifier e.g. LEI
objectTypes	not used	Not used in this this route
Type	mandatory	The Type parameter should be set to 'Predefined' indicating that this is a predefined route that follows a specific set of data relationships, navigation and logic
Route	mandatory	The Route name is FindESGStatementParent
Status	not used	Not used in this this route
effectiveAt	Not supported	Not supported for this predefined route
showHistory	Not supported	Not supported for this predefined route

### Equity Ticker + Country to RIC

Find out the RIC at the intersection of Ticker and country. Useful for disambiguating Tickers as they are not globally unique to a company.

Note – some ticker and country combinations can have a longer response time due to the complexity in the logic behind the predefined query.

Query Parameter	Mandatory/Optional	Comment
From	mandatory	Pairs of Ticker and Country code values are to be provided in the From parameter. In the format: "exchangeTicker": "LSEG", "countryCode": "GB" Multiple pairs of values can be provided in one query. Ticker is case sensitive. The country code must be provided in ISO 2 character format e.g. US Maximum 5 pairs of inputs in one query for best performance
To	optional	The user can set the To identifierTypes to RIC or PermID. If set to PermID the response will return the Quote PermID related to the RIC The To parameter is optional and if omitted by default the output will navigate to the RIC.

objectTypes	not required	If included must be set to "EdfQuote". If alias "anyquote" is used only EDFQuotes will be included in the navigation
Type	mandatory	The Type parameter should be set to 'Predefined' indicating that this is a predefined route that follows a specific set of data relationships, navigation and logic
Route	mandatory	The Route name is EquityTickerCountryCodeToPrimaryRIC
Status	optional	Filter out inactive quotes/RICs in the response
effectiveAt	optional	Restrict the navigation to a specific point in time
showHistory	Not supported	showHistory is not supported for this predefined route

## Ticker + MIC to Organization

Find out the organization at the intersection of Ticker and Market Segment MIC code.

Query Parameter	Mandatory/Optional	Comment
From	mandatory	Pairs of Ticker and MIC code values are to be provided in the From parameter. In the format: "exchangeTicker": "LSEG", "MIC": "XLON" Multiple pairs of values can be provided in one query, to a maximum of 5 pairs in one query Ticker and MIC values are case sensitive Use the most granular MIC available (segment MIC code)
To	optional	To is optional and if omitted the default will return the organization PermID at the intersection of Ticker+MIC User can specify an organization level identifier instead e.g. LEI or CIK
objectTypes	not required	Not required in From If included in To must be set to "Organization"
Type	mandatory	The Type parameter should be set to 'Predefined' indicating that this is a predefined route that follows a specific set of data relationships, navigation and logic
Route	mandatory	The Route name is ExchangeTickerMicToOrganization
Status	optional	Filter out inactive entities in the response
effectiveAt	optional	Restrict the navigation to a specific point in time
showHistory	Not supported	showHistory is not supported for this predefined route

## ISIN + Venue Identifier to Quote

Find out the RIC code at the intersection of ISIN and a venue identifier such as MIC, or price source

Query Parameter	Mandatory/Optional	Comment
From	mandatory	<p>Pairs of ISIN and venue identifiers are provided in the From parameter. In the format e.g.</p> <p>"MIC": "XNGS", "Isin": "US30303M1027"</p> <p>"Isin": "USG5690PAB79", "EjvPriceSourceCode": "CPL"</p> <p>"RDNExchangeCode": "NSM", "Isin": "US0231351067"</p> <p>"MarketAttributableSourcePermID": "21475145381", "Isin": "US0231351067"</p> <p>4 different venue identifiers are supported to identify the venue: MIC, RDNExchangeCode, PriceSourceCode, MarketAttributableSourcePermID If MIC is used it should be the most granular available MIC i.e. market segment MIC, if the venue has no segments then the operating MIC can be used. Multiple pairs of values can be provided in one query to a maximum of 5 pairs per query Values are not case sensitive</p>
To	optional	<p>The user can set the To identifierTypes to RIC or PermID. If set to PermID the response will return the Quote PermID related to the RIC The To parameter is optional and if omitted will default to Quote PermID</p>
objectTypes	not required	If included must use one of the quote level object Types e.g. 'EdfQuote'
Type	mandatory	The Type parameter should be set to 'Predefined' indicating that this is a predefined route that follows a specific set of data relationships and navigation and logic
Route	mandatory	<p>The route name is ISINVenueToQuote. This route was previously called IsinMicToQuote, this previous name remains backward compatible</p>
Status	optional	Filter out inactive entities in the response
effectiveAt	optional	Restrict the navigation to a specific point in time
showHistory	Not supported	showHistory is not supported for this predefined route

## FindPrimaryRIC

This predefined route provides a way to navigate from an organization or certain types of instrument to the main quote. The 'main' quote/RIC will typically be the quote listed on the venue with highest liquidity.

Query Parameter	Mandatory/Optional	Comment
From	mandatory	Can be any identifier type related to an Equity/Warrants/Government or Corporate bond Instrument or Organization. SEDOL is also supported.
To	optional	The user can set the To identifierTypes to RIC or PermID. If set to PermID the response will return the Quote PermID related to the Primary RIC The To parameter is optional and if omitted by default the output will navigate to the RIC.
objectTypes	depends	Optional if request starts with identifier Mandatory if request starts from an organization or instrument PermID
identifierTypes	optional	Optional but users are recommended to provide either objectType or identifierType for best performance
Type	mandatory	The Type parameter should be set to 'Predefined' indicating that this is a predefined route that follows a specific set of data relationships, from/to parameters and navigation logic
Route	mandatory	The Route name is FindPrimaryRIC.
Status	optional	Filter out inactive entities in the response
effectiveAt	optional	Restrict the navigation to a specific point in time
showHistory	optional	Show the history or primary RICs for the input value

## General information about query structure

Users are advised to provide as much information in the query as possible as this will give a better performance and more targeted results. Although specifying identifierTypes or objectTypes is optional, they should be added if known.

Certain combinations of to and from navigation are prohibited because the potential result set would be too large and negatively impact performance, for example navigation from trading venue identifier like MAS or MIC, to an instrument identifier is not permitted. Users will get an error message if trying not allowed navigations.

Search API is better suited to situations where criteria limitations beyond identifier/object type are required e.g. geography, currency, classification.

If a user provides a mix of PermIDs and Identifiers in the From input array, then the objectType parameter should not be included (even objectType=ANY) else the inputs that are identifiers will be ignored.

There is no predefined sort order to the entities returned in the API response and sort order should not be programmatically relied on to be in a certain order. If a more targeted response is required this should be done through a more specific query.

If a user experiences repeated timeouts on a specific query, the user is advised to reduce the complexity of the query through e.g. removing the reference parameters from the request, halving the number of inputs, adding objectType or IdentifierTypes details in the From part of the request.

## objectTypes and Aliases

Refinitiv entities have specific object types names depending on the type of entity. For example, an organization has the objectType 'organization'.

Instruments and quotes have more granular objectTypes that are asset class specific, for example: 'EDInstrument' (exchange data instrument), 'GovCorpInstrument' (government/corporate bond instrument).

For most general use cases, the customer does not need to know the granular object type and can instead use the aliases e.g. 'AnyInstrument' or 'AnyQuote' to include all possible types of that entity within the query navigation.

The objectType value 'Any' can be used as a super alias to scope in all objectTypes at the same time. 'Any' can also be used as an alias for all identifierTypes values in strict queries. The 'Any' alias is not allowed as a identifierType alias in auto type queries.

If the user specifies identifierTypes 'Any' in From, the identifier Types Edcoid, ExchangeISIN, UnderlyingInstrumentCodeESMA and UnderlyingInstrumentCodeDSB will be excluded from the navigation to improve performance. If the user wants these included in the navigation they must be explicitly stated.

Appendix A has a list of common objectTypes and aliases.

## Reference parameter

The reference parameter allows the user to add basic entity information to the query response to aid understanding and disambiguate results.

The parameter is optional and works with all query types in the /lookup end point. It is not available in the lookup-type end point.

The reference information provided is associated with entities, not identifiers. If the To is an identifier then the reference information will be that of the entity the identifier is linked to. For example, if the To is an ISIN, the reference information will be that of the instrument entity the ISIN is an identifier for.

The available reference fields are:

- Name – provides a common name for the entity. If the output is an identifier, the name will be the name of the entity the identifier is linked to.
- Status – indicates if the entity is active or inactive. If the navigation is to an identifier then this will indicate if the entity the identifier is linked to is active or inactive. Note, it is possible to have active (i.e. non end-dated) identifiers on inactive entities. Status is not available for some entity types e.g. funds
- Classification – a high level textual classification of the entity, or of the entity which the identifier is linked to

- PermID – provides the Permanent identifier of the entity an identifier is linked to. This will return the PermID of the ‘at-level’ entity to the input, e.g. if the ‘from’ is an ISIN this would return the instrument PermID. If the customer wants to map an identifier to a PermID it is better done through a query where the ‘to’ is a PermID and objectTypes is specified, rather than adding PermID as a reference field output.

The reference fields returned in symbology API should only be used to aid a human understanding of results. They must not be used for any other purpose such as a source of reference data for display or other applications, Refinitiv provides a range of specific reference data solutions for that purpose.

If the API is being called programmatically, we recommend not to return reference fields because response time performance may be reduced.

## Active Filter

The filter parameter can be used to show only the results related to active entities. This is useful for example when navigating to RICs, to return only the active RICs in the response.

The filter is applied to entities rather than identifiers. If the active filter is applied where the navigation is to a PermID, only PermIDs of active entities will be returned. If the active filter is applied where the navigation is to an identifier, then only the identifiers of active entities at the same level as the identifier would be returned. For example if active filter is used when the To is a RIC, only RICs attached to active quote entities would be returned.

The filter parameter only works on the response, not on the inputs. If a query involves multiple hops between relationships, the filter is only applied to the final output.

The filter parameter can be used in all types of queries in the lookup endpoint auto, strict and pre-defined.

Active filter use is included in the example queries.

## objectTypes when navigating ‘to’ a PermID with an auto query

Including the objectTypes is important and strongly recommended when navigating ‘to’ a PermID in an auto query to ensure the correct level of PermID is returned.

For example if the ‘from’ is a RIC and the ‘to’ is a PermID, the user should specify whether they want to navigate to the PermID of the quote, the instrument or the organization related to that RIC.

Examples in the table below:

Navigation	Query
Map identifier to Organization PermID, e.g. RIC	<pre>{   "from": [     {       "identifierTypes": ["RIC"],       "values": ["IBM.N", "TRI.TO"]     }   ],   "to": [     {       "objectTypes": ["organization"],       "identifierTypes": ["PermID"]     }   ],   "type": "auto" }</pre>
Map identifier to Instrument PermID	<pre>{   "from": [</pre>

	<pre> {   "identifierTypes": ["RIC"],   "values": ["IBM.N", "TRI.TO"] } ], "to": [   {     "objectTypes": ["anyinstrument"],     "identifierTypes": ["PermID"]   } ], "type": "auto" } </pre>
Map identifier to Quote PermID	<pre> {   "from": [     {       "identifierTypes": ["RIC"],       "values": ["IBM.N", "TRI.TO"]     }   ],   "to": [     {       "objectTypes": ["anyquote"],       "identifierTypes": ["PermID"]     }   ],   "type": "auto" } </pre>

If the To is a PermID and the objectTypes is not provided in the query, the output will be the PermID of the entity directly connected to the identifier.

## Pagination and dynamic pagination

By default, if an API response contains more than 10,000 rows the results will be paginated, this is to ensure large responses do not degrade experience by delaying delivery of the output. The query is run in full on Refinitiv systems and the full results are cached, and the user can then bring back those results from the cache page by page.

If results are paginated the customer will see the below at the end of the response:

```

"metadata": {
  "firstPage": "/lookup?id=5d85aab1-e984-4cf8-a891-7e6fcd3467a3&page=1",
  "nextPage": "/lookup?id=5d85aab1-e984-4cf8-a891-7e6fcd3467a3&page=2",
  "lastPage": "/lookup?id=5d85aab1-e984-4cf8-a891-7e6fcd3467a3&page=2",
  "totalPages": 2
}
"requestId": "...",
"effectiveAt": "...",
"messages": [
  "Output items for value X were carried to the next page"
]

```

In order to retrieve the second and/or subsequent pages the user would send a GET request to the API with the page details e.g. `discovery/symbology/v1/lookup?id=5d85aab1-e984-4cf8-a891-7e6fcd3467a3&page=2`. The link to receive subsequent pages has a 1 hour lifetime.

If an input value has results over multiple pages then the response is carried over to the next page. The relevant input value will be mentioned in the message and that input value is **repeated** on the subsequent page along with the additional responses to ensure inputs and outputs remain linked together.

The symbology API also supports dynamic pagination where the user can specify the page size in the request. To do this the user can add text below to a POST request to select the required page size, in this example required page size is 5

`?pageSize=5`

For example: `https://api.refinitiv.com/discovery/symbology/v1/lookup?pageSize=5`

## Health check endpoint

The Health check endpoint provides a health status of the application.

The user can send the following GET request: `api.refinitiv.com/discovery/symbology/v1/health`

If the application health is normal then the user will see the response:

```
{
  "status": "UP"
}
```

If the application is down the user will see an error message depending on your access method (Postman, API Docs etc)

## Lookup-type endpoint

The lookup-type endpoint allows a user to check the type information for a given input and any available effectiveFrom/effectiveTo dates for an identifier.

If the input is a PermID, the output will provide information on which identifierTypes and which objectTypes that PermID appears in.

If the input is an identifier, the output will provide the identifierType of the input.

The "showHistory": "true" parameter can be used in this endpoint which will return the effective from and effective to dates for identifiers.

## Endpoints

Discovery/symbology/v1 API has the following end points

<b>End Point</b>	<b>Description</b>
/LOOKUP	Provides mapping and navigating service to/from Identifiers and Refinitiv PermID based on the input. This is the main endpoint for requests on the symbology service.
/LOOKUP-TYPE	Check the Object or Identifier type information for a given input
/HEALTH	Healthcheck endpoint, check if the service is up

## Example Queries for /lookup endpoint

### Auto route query examples

Navigation	Query
Identifier to Identifier e.g. ISIN to LEI	<pre>{   "from": [     {       "identifierTypes": ["ISIN"],       "values": ["US30303M1027","US0231351067"]     }   ],   "to": [     {       "identifierTypes": ["LEI"]     }   ],   "type": "auto" }</pre>
Identifier to Identifier e.g. ISIN to CUSIP	<pre>{   "from": [     {       "identifierTypes": ["ISIN"],       "values": ["US30303M1027","US0231351067"]     }   ],   "to": [     {       "identifierTypes": ["CUSIP"]     }   ],   "type": "auto" }</pre>
Identifier to multiple Identifiers	<pre>{   "from": [     {       "identifierTypes": ["RIC"],       "values": ["DPWGn.DE"]     }   ],   "to": [     {       "identifierTypes": ["ISIN","LEI","ExchangeTicker"]     }   ],   "type": "auto" }</pre>

Query utilizing the reference parameter

```
{
  "from": [
    {
      "identifierTypes": [
        "LEI"
      ],
      "values": [
        "2138005T2FZP3AUV6X26"
      ]
    }
  ],
  "type": "auto",
  "to": [
    {
      "identifierTypes": [
        "PermID"
      ],
      "objectTypes": [
        "EDInstrument"
      ]
    }
  ],
  "reference": [
    "name",
    "status",
    "classification"
  ]
}
```

## Strict query examples

In strict queries the 'path' operator is used to define the relationships that join the entities between the 'from' and 'to'.

Navigation	Query
<p>Identifier to all the 'at-level' identifiers for the entity.</p> <p>'At level' means if the input is a quote level identifier, the output will be all related quote level identifiers; if input is organization level output will be organisation level identifiers etc</p>	<pre>{   "from": [     {       "identifierTypes": [         "ISIN"       ],       "values": [         "US4592001014"       ]     }   ],   "to": [     {       "identifierTypes": ["ANY"]     }   ],   "type": "strict" }</pre>
<p>Organization Identifier to Primary Equity RIC</p>	<pre>{   "from": [</pre>

'2-hop' query: organization  
to instrument, then  
instrument to quote

```
{
  "identifierTypes": ["LEI"],
  "values": ["549300561UZND4C7B569"]
},
"to": [
  {
    "identifierTypes": ["RIC"]
  }
],
"path": [
  {
    "relationshipTypes": ["InverselsPrimarySecurityOf"],
    "objectTypes": [
      {
        "from": "Organization",
        "to": "AnyInstrument"
      }
    ]
  }
],
{
  "relationshipTypes": ["InverselsValuationQuoteOf"],
  "objectTypes": [
    {
      "from": "AnyInstrument",
      "to": "AnyQuote"
    }
  ]
}
],
"type": "strict"
}
```

Equity Instrument to  
Valuation Quote RIC.

'One-hop' query instrument  
to quote

```
{
  "from": [
    {
      "identifierTypes": ["ISIN"],
      "values": ["CA8849037095"]
    }
  ],
  "to": [
    {
      "identifierTypes": ["RIC"]
    }
  ],
  "path": [
    {
      "relationshipTypes": [
        "InverselsValuationQuoteOf"
      ],
      "objectTypes": [
        {
          "from": "AnyInstrument",
          "to": "AnyQuote"
        }
      ]
    }
  ]
},
```

A query using multiple  
'from's

```

    "type": "strict"
  }
}
{
  "from": [
    {
      "identifierTypes": [
        "Isin"
      ],
      "values": [
        "XS0328866982"
      ]
    },
    {
      "identifierTypes": [
        "Cusip"
      ],
      "values": [
        "98975T101"
      ]
    },
    {
      "identifierTypes": [
        "Sedol"
      ],
      "values": [
        "BDDMNF9"
      ]
    }
  ],
  "type": "strict",
  "to": [
    {
      "objectTypes": [
        "Organization"
      ]
    }
  ],
  "path": [
    {
      "relationshipTypes": [
        "IssuedBy",
        "InverselyIssuedBy"
      ],
      "objectTypes": [
        {
          "from": "AnyInstrument",
          "to": "Organization"
        }
      ]
    }
  ]
}
]
}
}

```

## Predefined query examples

Predefined queries use specific built-in routes that are difficult to achieve or can't be written as auto or strict queries

Navigation	Query
FindESGStatementParent  Navigating from ISIN to PermID of the ESGStatementParent organisation	<pre>{   "from": [     {       "identifierTypes": [         "ISIN"       ],       "values": [         "US141784AR94"       ]     }   ],   "type": "predefined",   "route": "FindESGStatementParent" }</pre>
FindESGStatementParent  Navigating from ISIN to LEI of the ESGStatementParent organisation	<pre>{   "from": [     {       "identifierTypes": [         "ISIN"       ],       "values": [         "US141784AR94"       ]     }   ],   "to": [     {       "identifierTypes": [         "LEI"       ]     }   ],   "type": "predefined",   "route": "FindESGStatementParent" }</pre>
Ticker+MIC to Organization	<pre>{   "from": [     {       "values": [         {           "exchangeTicker": "LSEG",           "MIC": "XLON"         }       ]     }   ],   "to": [     {       "identifierTypes": [         "PermID"       ]     }   ], }</pre>

	<pre>       "objectTypes": [         "organization"       ]     },     "type": "predefined",     "route": "EquityTickerCountryCodeToPrimaryRIC"   } </pre>
<p>Ticker+Country to RIC route</p> <p>Multiple input pairs</p> <p>To return PermID instead of RIC, the 'to' parameter should be included with identifierTypes value PermID</p>	<pre> {   "from": [     {       "values": [         {           "exchangeTicker": "ADV",           "countryCode": "US"         },         {           "exchangeTicker": "VOD",           "countryCode": "US"         }       ]     }   ],   "to": [     {       "identifierTypes": [         "PermID"       ]     }   ],   "type": "predefined",   "route": "EquityTickerCountryCodeToPrimaryRIC" } </pre>
<p>ISIN + Venue Identifier to RIC using predefined route</p>	<pre> {   "from": [     {       "values": [         {           "MIC": "XLON",           "Isin": "GB00B0SWJX34"         }       ]     }   ],   "to": [     {       "identifierTypes": [         "RIC"       ]     }   ],   "type": "predefined",   "route": "IsinVenueToQuote" } </pre>
<p>FindPrimaryRIC</p>	<pre> {   "from": [     {       "identifierTypes": [ </pre>

From ISIN, if no 'to' is specified output defaults to RIC	<pre> "Isin" ], "values": [   "GB0030913577" ] } ], "type": "predefined", "route": "FindPrimaryRic" } </pre>
---	--

## Point in Time and History navigation examples

Navigation	Query
<p>Quote PermID to RIC navigation at a point in time.</p> <p>If effective date is moved to the following day the RIC for this Quote PermID changes from LSE.L to LSEG.L.</p>	<pre> {   "from": [     {       "objectTypes": [         "anyquote"       ],       "identifierTypes": [         "PermID"       ],       "values": [         "55850485015"       ]     }   ],   "to": [     {       "identifierTypes": [         "RIC"       ]     }   ],   "type": "auto",   "effectiveAt": "2021-01-28T12:34:33.100Z" } </pre>
<p>Navigation showing the history of primary RICs linked to ISIN GB00B012T521.</p> <p>RIC has multiple changes through time.</p>	<pre> {   "from": [     {       "identifierTypes": [         "Isin"       ],       "values": [         "GB00B012T521"       ]     }   ],   "reference": [     "name"   ],   "showHistory": true,   "type": "predefined",   "route": "FindPrimaryRic" } </pre>

## Example queries for /lookup-type endpoint

Look up	Query
Identifier to identifierTypes with effective from and to dates	<pre>{   "values": [     "RTR.L",     "FIA.MI",     "GDI.N"   ],   "showHistory": "true" }</pre>
PermID to identifierTypes and ObjectTypes	<pre>{   "values": [     "4298007752",     "8589953370"   ] }</pre>
Identifier to identifierTypes	<pre>{   "values": [     "LSEG.L",     "B0SWJX3",     "BQ4BKCS1HXDV9HN80Z93"   ] }</pre>

## Customer Responsibilities

Any use of symbology requires licensing by the relevant symbology owner. You may have the required usage rights under existing contracts with the symbology owner, and/or in some instances, symbology owners may allow usage on open licensing terms: please check your usage rights with relevant owners. Specifically (but without limitation) customers must ensure their licenses for SEDOL and CUSIP (covering CUSIP, CUSIP-based-ISIN and CIN) are kept current after access has been granted to view these fields.

Customers are reminded that any RIC usage requires a contractual RIC license granted by LSEG and is governed by its terms of use. Standard RIC usage rights granted as part of LSEG information services are set out in the Information Schedule of customer's contract with LSEG. Customers may also have specific RIC licenses granting additional RIC usage rights in specific use cases in addition to the standard ones. This [RIC Rights Overview](#) aims to provide a plain English description of RIC usage rights by providing answers to the most commonly asked questions by customers. If you have any queries relating to RICs, including the usage rights associated with them, please contact your account manager, or contact us at [www.refinitiv.com/en/contact-us](http://www.refinitiv.com/en/contact-us)

Customers should subscribe to the relevant alerts and Product Change Notifications in order to receive important update about service impacting issues or product changes and enhancements.

## Error messages

ID #	Error type	Status code	Message code	Error message
100	Request does not contain required parameter	400	MISSING_FIELDS_IN_QUERY_REQUEST_FORMAT	Missing fields in query request format
101			MISSING_REQUIRED_PARAMETER	Missing required parameter: <specific parameter>
102			RELATIONSHIP_TYPES_MISSED_INCOMPLETE_PATH	RelationshipTypes missed, incomplete path
103			PATH_STEP_ERROR	Path step #<segment number with issue> should have both `from` and `to` parts
104			SAME_FROM_TO_WITHOUT_PATH	The same object or identifier type was provided as input and output values. Please change either of these values or provide strict request with path between objects.
105			NO_REQUEST_BODY_PASSED	No request body is passed
106			MISSING_IDENTIFIER	Identifier type is missing in To part of query
107			AUTO_IDENTIFIER_TYPE_MISSING_IN_TO	Identifier type is missing in To part of auto query
107			AUTO_IDENTIFIER_TYPE_MISSING_IN_TO	Identifier type is missing in To part of index_auto query
108			TO_PART_MISSING_AUTOOBJECTS_DONT_MATCH_PATH	To part is missing from auto query
108			TO_PART_MISSING_AUTOOBJECTS_DONT_MATCH_PATH	To part is missing from index_auto query
109			FROM_PART_MISSING_AUTO	From part is missing from auto query
109			FROM_PART_MISSING_AUTO	From part is missing from index_auto query
110			VALUE_MISSING_AUTO	Value is missing in From part of auto query
110			VALUE_MISSING_AUTO	Value is missing in From part of index_auto query
111			NO_ID_PASSED	No id is passed
112			NO_PAGE_PASSED	No page is passed
113			MISSING_VALUES_IN_LOOKUP_TYPE_QUERY	Values is missing in look-up type query
114			USER_ID_NOT_PRESENT_IN_SYSTEM	User ID is not present in the system
TBA				
TBA				
200	Request has wrong format	400	INVALID_JSON_FORMAT	Invalid JSON format
201			PARSE_ERROR	Parse error. Expected array instead of singular value for the <field>
202			EFFECTIVE_AT_PARSE_ERROR	Unable to parse effectiveAt parameter (Invalid value). Correct format: yyyy-MM-ddTHH:mm:ss.msZ (without offsets)

ID #	Error type	Status code	Message code	Error message
203			EFFECTIVE_AT_NOT_IN_ROOT	Please put effectiveAt parameter in the root for multipath and filter requests
204			REFERENCE_IN_WRONG_PLA CE	Please put reference parameter in the root for multipath and filter requests
206			MULTIPLE_FROM_OR_TO_IN_ PREDEFINED_ROUTE	'From' and 'to' should be a single array, please provide multiple values in the same array
207			OR_AND_PREDEFINED_ERRO R	Predefined request can't be used in 'or' or 'and' request
208			FILTER_ERROR	Invalid filter format
209			PATH_WRONG_FORMAT_ERR OR	Incorrect format in field 'path'
TBA				
TBA				
300	Request has wrong data	400	UNRECOGNIZED_FIELD	Unrecognized field <non_existent_key>
301			NOT_SUPPORTED_TYPE	<type_value> type is not supported
302			OBJECTS_DONT_MATCH_PAT H	Object types in \"from\" or \"to\" don't match to path
303			IDENTIFIER_TYPES_NOT_FOU ND	Some identifier types are not found: UnknownType1, UnknownType2
304			OBJECT_TYPES_NOT_FOUND	Some object types are not found: UnknownType1, UnknownType2
305			RELATIONSHIP_TYPES_NOT_F OUND	Some relationship types are not found: UnknownType1, UnknownType2
306			ANY_OBJECT_IS_NOT_ALLOW ED_FOR_AUTO	ANY type should not be used for output object types of auto query
306			ANY_OBJECT_IS_NOT_ALLOW ED_FOR_AUTO	ANY type should not be used for output object types of index_auto query
307			ANY_IDENTIFIER_IS_NOT_ALL OWED_FOR_AUTO	ANY type should not be used for output identifier types of auto query
307			ANY_IDENTIFIER_IS_NOT_ALL OWED_FOR_AUTO	ANY type should not be used for output identifier types of index_auto query
308			ANY_OBJECT_IS_NOT_ALLOW ED_FOR_PREDEFINED	ANY type should not be used for object types in predefined query
309			ANY_IDENTIFIER_IS_NOT_ALL OWED_FOR_PREDEFINED	ANY type should not be used for identifier types in predefined query
310			NOT_SUPPORTED_IDENTIFIER _TYPE_FOR_AUTO_REQUEST_ ERROR	Input (or output) identifier type is not supported for auto request: <identifier type>
310			NOT_SUPPORTED_IDENTIFIER _TYPE_FOR_AUTO_REQUEST_ ERROR	Input (or output) identifier type is not supported for index_auto request: <identifier type>
311			NOT_SUPPORTED_OBJECT_T YPE_FOR_AUTO_REQUEST_ ERROR	Input (or output) object type is not supported for auto request: <object type>
311			NOT_SUPPORTED_OBJECT_T YPE_FOR_AUTO_REQUEST_ ERROR	Input (or output) object type is not supported for index_auto request: <object type>
312			PATH_NOT_APPLICABLE	Path is not applicable to <request type> requests

ID #	Error type	Status code	Message code	Error message
313			INCORRECT_ROUTE_NAME	Incorrect route name
313			INCORRECT_ROUTE_NAME	Values or types provided do not belong to this predefined route
318			FIELDS_IN_STRICT_REQUEST_ERROR	Field '<field name>' can be used only for some requests with predefined type. Please use fields 'identifierTypes' and 'values' instead
319			WRONG_REFERENCE	Reference parameter(s) <wrong parameter> is not supported
320			PAGINATION_ERROR	Incorrect request id or page
321			INPUT_VALUE_DOES_NOT_EXIST	Input value does not exist
322			INPUT_VALUE_FOR_IDENTIFIER_TYPE_DOES_NOT_EXIST	Input value for identifier type does not exist
323			INPUT_VALUE_FOR_OBJECT_TYPE_DOES_NOT_EXIST	Input value for object type does not exist
324			INPUT_VALUE_PROVIDED_IS_NOT_CURRENT	Input value does not exist in current time. You can use look-up type endpoint or 'showHistory' parameter to get more information
326			INPUT_VALUES_DO_NOT_EXIST	Input values do not exist
328			INPUT_IDENTIFIER_NOT_SUPPORTED_AUTO	Input identifier type is not supported for auto request
329			INPUT_OBJECT_NOT_SUPPORTED_AUTO	Input object type is not supported for auto request
330			LOOKUP_TYPE_PERM_ID_ERROR	PermId type should not be used for identifier types in lookup types query
331			PAGE_SIZE_NOT_NUMBER	Page size should be a digit number
332			INCORRECT_VALUE_IN_FIELD_SHOW_HISTORY	Incorrect format in field 'showHistory', should be true or false
334			VALUE_HAS_MULTIPLE_CIQM_TYPES_ERROR	Critical CIQM data error: Value <value> has <types> types.
335			HISTORY_NOT_SUPPORTED	History feature are not supported for such kind of request
336			FORBIDDEN_PARAMETER_WITH_SHOW_HISTORY	<forbidden parameter> parameter cannot be used along with showHistory
337			OBJECT_CANNOT_BE_CASTED	Object cannot be casted from <type 1> to <type 2>
339			MIX_PERMID_ANY_ERROR	Can't mix PermID with other identifier types in one array, please use request with 'or'
340			IDENTIFIER_TYPE_IS_NOT_VALID	<identifier type> is not valid
341			SIZE_NUMBER_IS_OUT_OF_SCOPE	Page size should be between 1 and 10000
342			INPUT_VALUE_PROVIDED_IS_NOT_CURRENT_LOOKUP_TYPE_ERROR	Input value does not exist in current time. You can use 'showHistory' parameter to get more information
TBA				
TBA				
400	Request is not allowed	400	INPUT_GREATER_THAN_ALLOWED	Number of input values is greater than maximum allowed 1500

ID #	Error type	Status code	Message code	Error message
402			NAVIGATION_NOT_ALLOWED	Navigation from <inputTypes> to <outputTypes> is not allowed because result size is too big
403			PREDEFINED_PERFORMANCE_EXCEPTION	A maximum of 5 input pairs are allowed for predefined routes
410		403	FORBIDDEN_FOR_DATA_EXPLORATION_ACCOUNT	Data Exploration account is not entitled to requested query
411			EXTERNAL_NOT_ENTITLED_FOR_RELATIONSHIPS	External account is not entitled to requested relationships
412			DUPLICATE_UUID_IS_NOT_ALLOWED_ERROR	Duplicate user id is not allowed in our system
413			DATA_EXPLORATION_ONLY_AUTO_QUERY_IS_ENTITLED	Only auto type is allowed for Data Exploration account
414			DATA_EXPLORATION_COMPLEX_QUERY_NOT_ENTITLED	Data Exploration account is not entitled to requested "and" or "or" query
415			DATA_EXPLORATION_NOT_ENTITLED_FOR_OBJECT_TYPE	Data Exploration account is not entitled to requested object types
416			DATA_EXPLORATION_IDENTIFIERS_NOT_ENTITLED_FOR_ALL	Data Exploration account is only entitled to navigate to a PermID
417			FORBIDDEN_ENDPOINT_FOR_DATA_EXPLORATION_ACCOUNT	This endpoint is forbidden for Data Exploration accounts
418			REFERENCE_NOT_ALLOWED_FOR_DATA_EXPLORATION_ACCOUNT	Reference fields are not allowed with Data Exploration account
419			FORBIDDEN_ENDPOINT_FOR_EXTERNAL	This endpoint is forbidden for external users
420		400	INPUT_GREATER_THAN_ALLOWED_DATA_EXPLORATION_ACCOUNT	Number of input values is greater than maximum <number> allowed with Data Exploration account
TBA				
TBA				
500	Unexpected exception	500	INTERNAL_SERVER_ERROR	Sorry, there was an unexpected problem with your request. Please contact our customer support for help.
501			DESERIALIZATION_ERROR	Deserialization error
502		400	PARSING_ERROR	Error, while parsing request. Please check syntax
503			NO_PROCESS_FOR_VALUE_ERROR	Unfortunately, we cannot process this query for value <value> because result size is too big
TBA				
TBA				
600	Debug exceptions	400	NO_NODE_PRESENT_IN_DEBUG	Should be at least one node present to show debug path - Neptune returned with empty result
601			DIFFERENT_NUMBER_OF_OBJECT_AND_PERMID	Incompatible object and permId size. Expected same, but got <number> objects - <number> permIds
602			MIXED_QUERY_ERROR	Error while processing mixed query
603			ERT_QUERY_NOT_SUPPORTED_YET	ERT query not supported yet
604			TWO_STEP_SEQUENTIAL_QUERY_NOT_SUPPORTED_YET	Two step sequential query not supported yet

ID #	Error type	Status code	Message code	Error message
605			QUERY_TYPE_NOT_SUPPORTED_YET	Query type not supported yet
606			QUERY_NOT_SUPPORTED_BY_CONVERTER	This query is currently not supported by query converter.
TBA				
TBA				
700			ALL_RESULTS_FILTERED	All of the results are filtered out
701			DATA_EXPLORATION_OBJECTS_NOT_ENTITLED_FOR_ALL	Data Exploration account is not entitled to all requested object types
702			EXTERNAL_PARTIALLY_NOT_ENTITLED_FOR_QUERY	External account is not entitled to part of the query
703			EXCLUDED_IDENTIFIERS_MESSAGE	If "\"identifierTypes\" is omitted in \"from\" then response only contains commonly used identifiers. If this does not match requirement please use exact identifier type
704			COUNT_OF_OUTPUTS_IS_TOO_HIGH	Reference fields could not be returned because count of outputs is too high
705			ERROR_VIA_CALL_OCS	There was an error returning some reference fields
706			ACCOUNT_NOT_ENTITLED_TO_IDENTIFIER_TYPES	This account is not entitled to all requested identifier types
707			STATUS_ERROR	Unable to retrieve status field for value(s) <values>
708			USER_NOT_ENTITLED_FOR_IDENTIFIER	User is not entitled to receive the <identifier> identifier back
			OUTPUT_ITEMS_WERE_CARRIED_TO_THE_NEXT_PAGE	Output items for <values> were carried to the next page

## List of objectTypes

Entity Type	Alias	ObjectType Name	Note
Organization		Organization	
Fund		Fund	This is the fund entity itself, not the organization that manages or administers the fund
Instrument	Anyinstrument	AbsCmoInstrument	Asset Backed or CMO fixed income Instrument
		CdsContract	
		EDInstrument	Exchange Traded Instrument
		FxirInstrument	FX or rates instrument
		GovCorpInstrument	Government/corporate bond instrument
		MbsPoolInstrument	
		MbsTbaInstrument	Mortgage Backed Security To Be Announced instrument
		MuniInstrument	
		FundShareClass	Lipper 'LP' RICs are at this level
		SPInstrument	Structured Product Instrument.
		CIQMInstrument	CIQMInstruments will only be displayed in results if the instrument is not available within the other instrument objectTypes
Quote	Anyquote	AbsCmoQuote	
		CdsQuote	
		EdfQuote	
		FxirQuote	
		GovCorpQuote	
		MbsPoolQuote	
		MbsTbaQuote	
		MuniQuote	
		SPQuote	
Vehicle	Anyvehicle	AbsCmoVehicle	
		CdsVehicle	
		FxirVehicle	
		GovCorpVehicle	
		MuniVehicle	
Venue		MarketAttributableSource	Source of pricing, e.g. trading venue
Fundamental		Fundamental	A set of financial statements for a company
Fundamental Series		FundamentalSeries	A series of Fundamentals combined to create a continuous timeseries through time that have common accounting standards (e.g. GAAP) and basis (e.g. consolidated)

Note - 'Any' can be used as a super-alias to include all objectTypes





ExchangeTicker		X								
FinanceMinistryVenezuelaCode			X							
FinlandCode			X							
HongKongCode		X	X		X					
HongKongMonetaryAuthorityIssueCode			X							
HungaryCode			X							
IndepthDataId			X		X					
Isin		X	X	X	X	X	X	X		
Ismald			X		X					
ISMNumber		X								
Israelfundid						X				
ItalianCode			X		X					
JapanItaCode										
JapanSicc		X								
JohannesburgCode		X								
JohannesburgStockExchangeCode		X								
KazakstanCode		X								
KenyaCode			X							
KoreaCode		X								
LEI	X									
LimaStockExchangeCode			X							
LocalCodeValue			X		X					
LuxembourgCode			X							
MalaysiaCode		X	X							
MexicoBmvCode			X							
MexicoCode		X								
Mic (note this is the segment MIC)		X	X							X
MontevideoStockExchangeCode			X							
NetherlandsCode		X								
NewZealandCode		X								
NorwayCode			X							
OperatingMIC		X	X							X
OportoDerivativesCode		X								
OsloStockExchangeTicker			X							
PanamaStockExchangeSymbol			X							
PhilippinesCode		X								
RioDeJaneiroCode		X								
RussianRegistrationCode			X							
SantiagoStockExchangeCode			X							
SaoPaoloCode		X								
Sedol		X	X	X	X					
ShanghaiCode		X	X							



ORC (aka 'Other RIC Code' used for expired Fixed Income RICs) RIC includes ORC as an alias			X		X		X	X	X	
BridgeSymbol			X		X		X			
GemAlphaNumericId	X									
NdaOrgId	X									
MXID	X									
TmtCompanyId	X									
Repno	X									
RDNExchangeCode										X
LipperId	X					X				
DsQuotationNumber		X								
TMTPermID		X								
IlxId		X								
VentureEconomicsId	X									
ThomsonTicker		X								
DatastreamId		X								
WorldscopePermId		X								
WorldscopId		X								
Edcoid	X									
IbesTicker		X								
FundClassTicker						X				
VEFirmID	X									
Zpage			X		X					
EJVPriceSourceCode			X							

# Support

For support using this product, go to our [Help & Support](#) page where you can raise a query for your problem.

You can keep informed of changes to products and data, as well as subscribing to real-time service alerts through our [Notifications and Alerts](#) page. A [Subscriptions](#) section is included on the page where you can subscribe to different support channels, or manage your current subscriptions.

You are encouraged to subscribe to the following support channels:

## Change Notifications channels

### Data Notifications

Content notifications of new, enhanced, or changed functionality. These may require your action, in products that you use.

### Content Changes

Upcoming changes to real-time and historical data across all asset classes that are relevant to you.

### RIC Change Events

Planned changes to Refinitiv's Instrument Codes.

## Service Alerts

### Service Alerts

Real-time service alerts about planned maintenance and unplanned service issues affecting your products and services. You can be notified via SMS or email.

# Your Personal Information

LSEG is committed to the responsible handling and protection of personal information.

We invite you to review our [Privacy Statement](#).

This describes how we collect, use, disclose, transfer, and store personal information when needed to provide our services, and for our operational and business purposes.

The Privacy Statement provides information about your rights. Also, how you can contact us if you have questions about how we handle your information.

