

ESG DATA API DOCUMENTATION

MSCI ESG Data Application Programming Interface (API) Documentation. Additional documentation can be found at:

<https://developer.msci.com/apis/esg-data-api>

Version 6.0

March 2025

Overview and Conventions	3
ESG Data API	3
HTTP Method	3
API Versioning	4
Version wise change details.....	4
Metering and Throttling	5
Gateway Limits.....	5
Batching response strategy.....	6
Backward compatibility	7
Authentication	7
Step 1 – Generate Client Id	7
Step 2 – Generate Client Secret.....	8
Step 3 – Obtain token	8
ESG Data API	10
Description	10
Authentication Details	10
Endpoints List.....	11
Endpoints Detail.....	12
/funds/	12
/funds/History	22
/instruments/history	46
/issuers/	65
/issuers/history.....	76
/metadata/factors	103
/metadata/history/factors	106
/metadata/history/datasets	109
/metadata/history/historicalproducts	111
/metadata/history/thematictags	113
/metadata/history/mscimodeltags	115
/parameterValues/countries.....	117
/parameterValues/coverages.....	120
/parameterValues/esgIndustries.....	120

/parameterValues/factorCategoryPaths	121
/parameterValues/factorProductNames.....	122
/parameterValues/fundAssetClasses	122
/parameterValues/fundAssetUniverses	123
/parameterValues/fundDomiciles	123
/parameterValues/fundLipperGlobalClasses	123
/parameterValues/gicsSubIndustries	123
/parameterValues/Indexes.....	124
/INSTRUMENT-REFERENCE.....	124

OVERVIEW AND CONVENTIONS

MSCI ESG APIs are RESTful Web APIs that enable direct access to MSCI ESG reports and data over the internet, enabling the seamless integration of ESG Reports and data into client and third-party platforms.

Additional documentation is available on the MSCI API Portal located here: <https://developer.msci.com/apis/esg-data-api>

ESG DATA API

The ESG Data API offers access to MSCI ESG data including ratings and raw data. All client facing data is expected to be available via the Data API.

HTTP METHOD

Unless stated otherwise, all Web Services are using HTTP 'GET' method.

API VERSIONING

ESG APIs uses version numbers. If change is necessary to be made to an API that will break the current version then the version number will be changed.

API	V1 (Basic Auth)	V2 (OAuth 2.0)	V3(OAuth 2.0)
Ongoing Data	/esg/data/v1.0/issuers	/esg/data/v2.0/issuers	/esg/data/v3.0/issuers
	/esg/data/v1.0/funds	/esg/data/v2.0/funds	/esg/data/v3.0/funds
	NA	NA	Upcoming
Historical Data	/esg/data/v1.0/issuers/history (Beta)	/esg/data/v2.0/issuers/history (Beta)	/esg/data/v3.0/issuers/history
	/esg/data/v1.0/funds/history (Beta)	/esg/data/v2.0/funds/history (Beta)	/esg/data/v3.0/funds/history
	/esg/data/v1.0/instruments/history (Beta)	/esg/data/v2.0/instruments/history (Beta)	/esg/data/v3.0/instruments/history
Metadata	/esg/data/v1.0/metadata/factors	/esg/data/v2.0/metadata/factors	/esg/data/v3.0/metadata/factors
	NA	NA	/esg/data/v3.0/metadata/history/factors
Instrument Ref API	NA	/esg/data/v2.0/instrument-reference	/esg/data/v3.0/instrument-reference

VERSION WISE CHANGE DETAILS

Sr No	Changed Description	Document Version
1	New features support in History API Urls. ("historical_product" & "dataset_list") Refer to sections "/issuers/history" & "/funds/history" for these new query parameters.	Version 4
2	Historical Product List support in History Metadata URLs ("historical_product_list") Refer to sections "/metadata/history/factors" and "metadata/history/datasets" for this new query parameter.	Version 4
3	New endpoint to retrieve all available Historical products list. Refer to section "/metadata/history/historicalProducts" for more details.	Version 4
4	Gateway Limits (For v3 history batch requests)	Version 4

5	Batching Response Strategy	Version 4
6	Historical API input parameter “as_at_date” description changes. Please refer history API urls sections.	Version 5
7	Historical Instrument v2 (Beta) APIs migrated to v3 (GA) & New features support in Instrument History API (“historical_product” & “dataset_list”).	Version 6

METERING AND THROTTLING

To prevent misuse due to either malicious intent or programmatic errors and ensure continuous availability of API Services to all our Clients, MSCI implements Throttling (also referred to as Rate Limiting Features). Rate Limits are typically defined in terms of number of API requests made per minute and per day for an API Key. These limits may be subject to commercial Terms & Conditions.

HTTP Response 429 - Too Many Requests is sent back by the API Server (Gateway) whenever a client breaches a predefined rate limit.

This response is also including a Retry-After header, specifying the number of seconds after which the request can be retried.

GATEWAY LIMITS

GATEWAY LIMITS (ALL URLS)

Parameter	Description	Value
rate limit - user request rate	How many requests/second can an individual user make to your API?	100
rate limit - user connection limit	How many concurrent HTTP connections can an individual user have to your API?	75
rate limit - cluster/global request limit	Across all users, how many total requests/second can your API handle? This will depend on the characteristics of your API and how it scales.	1000
rate limit - cluster/global connection limit	Across all users, how many total concurrent HTTP connections can your API handle? This will depend on the characteristics of your API and how it scales.	500

GATEWAY LIMITS (FOR V3 HISTORY BATCH REQUESTS)

Parameter	Description	Value
rate limit - user request rate	How many requests/second can an individual user make to your API?	25
rate limit - cluster/global request limit	Across all users, how many total requests/second can your API handle? This will depend on the characteristics of your API and how it scales.	100

DATA RANGE LIMIT PER “DATA_LAYOUT” AND “DATA_SAMPLE_FREQUENCY”

FOR “HISTORICAL_PRODUCT” REQUEST - V3.0

data_sample_frequency	daily	NA	daily	calendar_month_end	calendar_month_end
data_layout	by_factor	change_ranges	matrix_by_factor	by_factor	matrix_by_factor
/v3.0/issuers/history	1 day	1 year	1 day	1 year	1 year
/v3.0/funds/history	1 day	1 year	1 day	1 year	1 year
/v3.0/instruments/history	1 day	1 year	1 day	1 year	1 year

BATCHING RESPONSE STRATEGY

While using History API, a user might request data for a large number of Identifiers and factors for a wide date range (say 1 year), which could result in a response that is too large to handle efficiently. To address this, we have implemented a batching strategy, where we send the response in smaller, more manageable chunks or 'batches'.

If the number of Identifiers or the number of Factors is more than 100, History APIs response will be in the form of Batches where each Batch can have data for maximum 100 identifiers and 100 factors.

Batch No.	Identifiers (Range)	Factors (Range)
1	1-100	1-100
2	1-100	101-200
3	1-100	201-250
4	101-150	1-100
5	101-150	101-200
6	101-150	201-250

In this scenario, we are dealing with 150 identifiers and 250 factors. Our batching strategy is designed to manage this data efficiently:

- The first batch contains data for the initial 100 identifiers and the first 100 factors.
- The second batch includes data for the same initial 100 identifiers but moves on to the next 100 factors.

- The third batch continues with the same 100 identifiers, but now includes the remaining 50 factors.

Through these first three batches, we have successfully transmitted data for all 250 factors pertaining to the first 100 identifiers.

- The fourth batch begins with data for the next set of 50 identifiers, covering the first 100 factors.
- The fifth batch continues with these 50 identifiers but includes data for the next 100 factors.

This pattern continues, ensuring that we first send data for the initial 100 identifiers across all factors (in multiple batches), before proceeding to the next set of identifiers. This strategy allows for the efficient transmission of large data sets by breaking them down into manageable, organized batches.

If the number of Identifiers and number of factors are less than 100, batching will not be required, and all the data will be sent in a single response.

Batch No.	Identifiers (Range)	Factors (Range)
1	1-50	1-50

Notes:

1. Refer to sections “/funds/history”, “/issuers/history” and “/instruments/history” for batching request and response.
2. Maximum number of identifiers and maximum number of factors is 100 for Issuers and Instrument API, 50 for Funds API.

BACKWARD COMPATIBILITY

When changes are needed to an API, MSCI will strive to make them backward compatible.

AUTHENTICATION

The following section describes machine to machine authentication.

To use MSCI Web Services, the user needs:

- Client Id
- Client Secret

Client Id and secret are obtained once and are reusable. To obtain a Client Id and secret, the client needs to call MSCI’s client service.

STEP 1 – GENERATE CLIENT ID

1. Log in to the MSCI Customer Center using your MSCI credentials.
2. Select the OAuth 2.0 Client IDs tab.
3. In the top-right corner, click Request Client ID.
4. From the product drop-down list, select ESG.

5. Provide the required additional information– describe the purpose and indicate that you are requesting a new ID.
6. Select the I agree to MSCI terms check box.
7. Click Submit. You will receive an email with your Client ID.

Note that you must have an active subscription of the product for which you are requesting the Client ID.

A Client ID is generated and sent to the backend team for provisioning with the requested API.

You will receive an email confirmation once the ID is provisioned. You can also track the status from the MSCI Customer Center.

STEP 2 – GENERATE CLIENT SECRET

1. Once you receive an email confirmation that your Client ID is provisioned, go to the MSCI Customer Center again. Select the OAuth 2.0 Client IDs tab.
2. Select the OAuth 2.0 Client IDs tab.
3. Locate the new Client ID that was assigned to you, as described in the previous section.
4. Click Generate Secret. For security reasons, this is a one-time activity that is visible in the portal only at the time of generation.
5. Save your IDs in a safe place. If you require assistance with this process, contact your MSCI representative. Note that MSCI does not store the API Secret.

STEP 3 – OBTAIN TOKEN

Once you have the Client ID and Client Secret, you need to make an API call to obtain the token.

Auth Server URL <https://accounts.msci.com/oauth/token>

Request Type POST

Body

client_id: <Client_ID>

client_secret: <Client_Secret>

grant_type: "client_credentials"

audience: "<https://esg/data>"

Response of a successful call contains the access token and expiry time.

Use the access token as a bearer token for further API communication.

To ensure security, all API requests must be made over HTTPS.

EXPIRY OF TOKEN: 24 HOUR

After Token expires, we will get "401 unauthorized" response.

DOs:

Please reuse the OAuth token until it expires.

Don'ts:

Please avoid requesting a new OAuth token for every single API request. Instead, reuse the existing token until its validity expires.

CODE EXAMPLE OF USAGE:

CURL:

(TOKEN GENERATION) –

```
curl -X POST \
https://accounts.msci.com/oauth/token/ \
-H 'Content-Type: application/json' \
-d '{
  "grant_type": "client_credentials",
  "client_id": "***Client-Id-value***",
  "client_secret": "***Client-Secret-value***",
  "audience": "https://esg/data"
}'
```

(API REQUEST)

```
curl -H "Authorization: Bearer
***bearer token value***"
-X GET "https://api.msci.com/esg/data/v3.0/metadata/factors?factor_type=fund"
```

JAVA :

(TOKEN GENERATION) –

```
OkHttpClient client = new OkHttpClient();
MediaType mediaType = MediaType.parse("application/json");
RequestBody body = RequestBody.create(mediaType, "{ \"grant_type\": \"client_credentials\", \"client_id\": \"***Client-Id-Value***\", \"client_secret\": \"***Client-Secret-Value***\", \"audience\": \"https://esg/data\" }");
Request request = new Request.Builder()
    .url("https://accounts.msci.com/oauth/token/")
    .post(body)
    .addHeader("Content-Type", "application/json")
    .build();
Response response = client.newCall(request).execute();
```

(API REQUEST) –

```
OkHttpClient client = new OkHttpClient();
Request request = new Request.Builder()
    .url("https://api.msci.com/esg/data/v3.0/metadata/factors?factor_type=fund")
    .get()
    .addHeader("Accept", "application/json")
    .addHeader("Content-Type", "application/json")
    .addHeader("Authorization", "Bearer **Bearer-Token-Value")
    .build();
```

```
Response response = client.newCall(request).execute();
```

PYTHON :

(TOKEN GENERATION) –

```
import http.client
import json
conn = http.client.HTTPSConnection("accounts.msci.com")
payload = json.dumps({"grant_type": "client_credentials",
"client_id": "**Client-Id-value**",
"client_secret": "**Client-Id-value**",
"audience": "https://esg/data"})
headers = {
'Content-Type': 'application/json'}
conn.request("POST", "/oauth/token/", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

(API REQUEST) –

```
import http.client
conn = http.client.HTTPSConnection("api.msci.com")
headers = {'Accept ': 'application/json', 'Content-Type ': 'application/json', 'Authorization': 'Bearer **Bearer Token Value**'}
payload = ""
conn.request("GET", "/esg/data/v3.0/metadata/factors?factor_type=fund ", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

ESG DATA API

The base URL of the ESG Data API is <https://api.msci.com/esg/data/v3.0>

DESCRIPTION

The ESG Data API provides a way for a client to retrieve current day ESG data for all issuers and data points that the user is entitled to receive. Data entitlements match what is available in MSCI ESG Manager.

AUTHENTICATION DETAILS

Use basic authentication header with: API Key/Secret, MSCI username prefixed with MSCI\, Email/password or Internal technical account (sys_*)

ENDPOINTS LIST

Endpoint	Description
/funds (GET)	Retrieve a list of funds and related fund Ratings data
/funds (POST)	Retrieve a list of funds and related fund Ratings data based on the posted input specification
/funds/history [GET]	Retrieve ESG data history for a given set of fund identifiers
/funds/history [POST]	Retrieve ESG data history for a given set of fund identifiers based on the posted input specification
/instruments/history [GET]	Retrieve ESG data history for a given set of instrument identifiers
/instruments/history [POST]	ESG data history for a given set of instrument identifiers based on the posted input specification
/issuers/history [GET]	Retrieve issuer ESG Data History for a set of issuers identifiers
/issuers/history [POST]	Retrieve issuer ESG Data History for a set of issuers identifiers based on the posted input specification
/metadata/factors [GET]	Retrieve a list of factors permitted to the account, optionally limited to either issuer ,instrument or fund factors
/metadata/history/factors [GET]	Retrieve a list of issuer, fund and instrument history factors irrespective of account permission.
/parameterValues/countries	Return a list of available countries
/parameterValues/coverages	Return a list of available issuer coverage universes
/parameterValues/esgIndustries	Return a list of available ESG Ratings Industries
/parameterValues/factorCategoryPaths	Retrieve a list of Category Paths used to organize factors

/parameterValues/factorProductNames	Retrieve a list of product names used to identify groupings of factors
/parameterValues/fundAssetClasses	Return a list of available fund asset class names
/parameterValues/fundAssetUniverses	Return a list of fund asset universe names
/parameterValues/fundDomiciles	Return a list of fund domicile names
/parameterValues/fundLipperGlobalClasses	Return a list of Lipper global class names
/parameterValues/gicsSubIndustries	Retrieve a list of available GICS Sub-Industries
/parameterValues/indexes	Retrieve a list of available indexes
/instrument-reference/[GET]	Retrieve a list of all instruments for an issuer
/instrument-reference/[POST]	Retrieve a list of all instruments for an issuer

ENDPOINTS DETAIL

/FUNDS/

HTTP Method:

GET

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a set of funds, containing factor data for each fund, based on the parameters given in the request. The results are governed by the data points and fund coverage permissioned to the account. This request allows the caller to specify which data they want to retrieve, and from which 'universe' of funds the results should come from. For example, a request could ask for all Fund Metrics:Summary category data for funds in the "Equity" fund asset class. The result would be a JSON object with a list of funds along with the requested data. The query would look like this:

https://api.msci.com/esg/data/v3.0/funds?category_path_list=Fund Metrics:Summary&fund_asset_class_list=Equity

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/funds>

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
format	string		json	json, csv, xml	<p>The result of the issuers endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. XML can also be specified, and the result will be a properly formatted XML document without a DTD or schema.</p>
offset	Integer (must be greater than zero)				<p>The Data API has the potential to generate large amounts of data. If a query will result in extra large amounts of data, the user will be required to specify paging parameters. The offset value indicates which record to start retrieving values. This option is usually paired with the limit parameter to create a window of results to return. When paging is using, an additional section of the JSON response will contain information about the total number of records that could be returned, along with links for the next,prev,first,last pages where appropriate. If you are on the last page, no next link will be generated, for example.</p>
limit	Integer (must be greater than one)				<p>The Data API has the potential to generate large amounts of data. If a query will result in extra large amounts of data, the user will be required to specify paging parameters. The limit value indicates the maximum number of records to return. This option is usually paired with the offset parameter to create a window of results to return. When paging is using, an additional section of the JSON response will contain information about the total number of records that could be returned, along with links for the next,prev,first,last pages where appropriate. If you are on the last page, no next link will be generated, for example.</p>
product_name_list	Array				<p>This parameter is a list of one or more product names which contain groups of data factors. Data factors are grouped by category as well as by product name. A product name is used to identify a product to which a data factor must belong in order to be returned. The list of products available to the caller can be found via the /parameterValues/factorProductNames endpoint.</p>
[array item]	String				<p>Data factors are grouped by category as well as by product name. A product name is used to identify a product to which a data factor must belong in order to be returned. The list of products available to the caller can be found via the /parameterValues/factorProductNames endpoint.</p>

Name	Type	Mandatory	Default value	Accepted values	Description
category_path_list	Array				<p>This parameter is a list of one or more category path strings.</p> <p>A category path is used to identify a collection of data factors. All category paths available to the caller can be obtained from the <code>/parameterValues/factorCategoryPaths</code> endpoint.</p>
[array item]	String				<p>A category path is used to identify a collection of data factors. All category paths available to the caller can be obtained from the <code>/parameterValues/factorCategoryPaths</code> endpoint.</p>
factor_name_list	Array				<p>This is a list of factor names. A factor name is used to identify a unique data point value associated with an issuer. A full list of available factors for the caller can be found by using the <code>/metadata/factors</code> endpoint.</p>
[array item]	string				<p>A factor name is used to identify a unique data point value associated with an issuer. A full list of available factors for the caller can be found by using the <code>/metadata/factors</code> endpoint.</p>
name_contains	String				<p>The name matches string is used to locate issuers whose primary issuer name value contains the given string anywhere in the name. If parent/child options are used, additional issuers may be returned that are related to the primary issuer but will not necessarily match the pattern given.</p>
starts_with	String				<p>Limit the primary issuer to a name that starts with the specified value. If parent/child options are specified, there may be issuers included that don't start with the specified name.</p>
fund_lipper_global_class_list	Array				<p>This parameter contains a list of one or more fund lipper global class names. The available names can be determined by first issuing a request to the <code>/parameterValues/fundLipperGlobalClasses</code> endpoint. Names from that list may be used in this query to limit results to funds located in the specified Lipper global classes.</p>
[array item]	string				
fund_domicile_list	Array				<p>This parameter contains a list of one or more fund domicile names. The available domicile names can be determined by first issuing a request to the <code>/parameterValues/fundDomicileNames</code> endpoint. Names from that list may be used in this query to limit results to funds located in the specified domiciles</p>
[array item]	string				<p>A Country Code is the 2-character code representing a country.</p>

Name	Type	Mandatory	Default value	Accepted values	Description
fund_asset_universe_list	Array				This parameter contains a list of one or more fund asset universe names. The available fund asset universe names can be determined by first issuing a request to the /parameterValues/fundAssetUniverses endpoint. Names from that list may be used in this query to limit results to funds located in the specified fund asset universes.
[array item]	string				A GICS SubIndustry code is a string value which is used to identify a particular GICS SubIndustry. A full list of GICS SubIndustries is available from the /parameterValues/gicsSubIndustries endpoint.
fund_asset_class_list	Array				This parameter contains a list of one or more fund asset class names. The available fund asset class names can be determined by first issuing a request to the /parameterValues/fundAssetClasses endpoint. Names from that list may be used in this query to limit results to funds located in the specified fund asset classes.
[array item]	string				The ESG Industry ID is a string value that identifies an ESG Industry. A full list of available ESG Industries can be retrieved from the /parameterValues/esgIndustries endpoint
fund_identifier_type	string				When requesting data, the client can specify a list of one or more issuer identifiers for which data will be returned. The default issuer identifier is an MSCI issuer id. However, by specifying a different lookup factor, the client can specify IDs such as ISINs. All identifiers must be of the same type, however. For example, mixing ISINs and MSCI Issuer Ids is not allowed.
fund_identifier_list	Array				This parameter is used to limit the results to a specific set of issuers. The caller can list one or more issuer identifiers which identified the issuers for which data should be returned.
[array item]	string				The issuer identifier is a value that is used to locate an issuer within the universe of issuers available to the client. By default, this identifier is an MSCI Issuer ID value, but this can also be a different identifier type when the issuer_identifier_type is specified. All identifiers must be of the same type.
[array item]	string				The issuer id is an identifier used to locate an issuer within the universe of issuers available to the client. By default, this id is an MSCI Issuer ID value, but this can also be a different identifier type when the lookup_factor is specified. All ids must be of the same type.

Name	Type	Mandatory	Default value	Accepted values	Description
fund_metrics_coverage_only	string				The 'fund_metrics_coverage_only' flag is supposed to limit results to funds in fund metrics coverage if set to true. If you want to have fund which are not part of coverage you need to add 'fund_metrics_coverage_only=false' in the API query. By default fund endpoint fund_metrics_coverage_only is set to true. By default fund endpoint fund_metrics_coverage_only is set to true. If you set it to false, it will find funds that aren't in the fund metrics universe but not include those that are in the universe.

RESPONSE

Possible Outputs

Status	Description
200	Success
206	Partial Content may have been returned
400	Malformed request
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json, text/csv

Output Description

The fund endpoint returns a collection of data for the fund and datapoints requested. By default, the result is returned in JSON format. The results can also be returned as a CSV file if the request uses the format parameter, or if the accept header specifies text/csv as an acceptable result.

Sample Output

Possible outputs for status: 200

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "9d19a15ff2e4832a",
  "timestamp": "2019-03-15T21:22:11Z",
  "messages": [],
}
```



```

"result": {
  "funds": [
    {
      "FUND_NAME": "Parnassus Fund;Investor",
      "FUND_SHARE_CLASS_ID": "40001494",
      "FUND_ID": "35529804",
      "FUND_ISIN": "US7017651099",
      "FUND_TICKER": "PARNX",
      "FUND_ESG_QUALITY_SCORE_PCTL_GLOBAL": 41.547134356404634,
      "FUND_IN_UNIVERSE": true
    },
    {
      "FUND_NAME": "Fidelity Advisor Capital Development Fund;O",
      "FUND_SHARE_CLASS_ID": "40001683",
      "FUND_ID": "35530080",
      "FUND_ISIN": "US3161272089",
      "FUND_TICKER": "FDETX",
      "FUND_ESG_QUALITY_SCORE_PCTL_GLOBAL": 37.2439711869715,
      "FUND_IN_UNIVERSE": true
    },
    {
      "FUND_NAME": "JPMorgan US Equity Fund;I",
      "FUND_SHARE_CLASS_ID": "40001538",
      "FUND_ID": "36450272",
      "FUND_ISIN": "US4812A11594",
      "FUND_TICKER": "JUESX",
      "FUND_ESG_QUALITY_SCORE_PCTL_GLOBAL": 48.62198559348575,
      "FUND_IN_UNIVERSE": true
    }
  ]
}

```

CODE SAMPLES

cURL

```

curl -i -H "Accept: application/json"
-X GET
https://api.msci.com/esg/data/v3.0/funds?category_path_list=ESG+Ratings:Company+Summary&format=json

```

Python

```

import requests url =
"https://api.msci.com/esg/data/v3.0/funds?category_path_list=ESG+Ratings:Company+Summary&format=json"; headers = { "Accept": "application/json" } response = requests.request("POST", url,
headers=headers) print(response.text)

```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```

HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/funds?category_path_list=ESG+Ratings:Company+Summary&format=json") .header("Accept", "application/json") .asString();

```

C#

```

var client = new
RestClient("https://api.msci.com/esg/data/v3.0/funds?category_path_list=ESG+Ratings:Company+Summary&format=json"); var request = new RestRequest(Method.GET); request.AddHeader("Accept",
"application/json"); IRestResponse response = client.Execute(request);

```

HTTP Method:

POST

ENDPOINT DESCRIPTION

In addition to requesting fund data via a GET request, users can also create a POST request where the POSTed data consists of the various query parameters that are available on the GET request. The POST form of this endpoint allows for potentially large amounts of data to be specified. For example, a client may have a list of issuers IDs that they wish to retrieve data for. Specifying the ids in a GET request may exceed the maximum length allowed for a GET request. A POST request has no such limit.

It is important to note that any POST request specify the Content-Type being used for submission. For example, if JSON is being used to send data, the Content-Type should be application/json, otherwise the request may be rejected with a 400 status code and an 'Invalid formatting' error message.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/funds>

Body payload

Name	Type	Mandatory	Default value	Accepted values	Description
fund_identifier_list	Array				
[array item]	String				
fund_identifier_type	String		ISIN	ASSET_OVERVIEW_ID, ISIN, SHARE_CLASS_ID, TICKER	The fund identifier type is a string defining the factor name used for to identify funds specified in the fund_identifier_list parameter. Available options vary by user, but include one or more of the following ASSET_OVERVIEW_ID, ISIN, SHARE_CLASS_ID, TICKER
fund_asset_class_list	Array				
[array item]	String				
fund_asset_universe_list	Array				
[array item]	String				
fund_domicile_list	Array				
[array item]	String				
fund_lipper_global_class_list	Array				
[array item]	String				
fund_metrics_universe_only	boolean		TRUE		The fund_metrics_universe_only flag restricts the funds that are retrieved to only those that are in the MSCI Fund Ratings Universe

Name	Type	Mandatory	Default value	Accepted values	Description
starts_with	String				Locate issuers whose name starts with the given value and return requested data related to those issuers. If parent/child is specified, the related issuers will not necessarily have names that start with the given string. The value is only used to identify the primary issuers that are the result of the query.
name_contains	String				If a name_contains value is specified, only issuers that have the specified string will be included in the results.
factor_name_list	Array				
[array item]	String				
category_path_list	Array				
[array item]	String				
product_name_list	Array				
[array item]	String				
Limit	integer			Must be greater than 1	The limit parameter is used to define the maximum number of results that will be returned in the request. This parameter is option; however some data requests will result in a large number of issuers being returned. When this is detected by the API, an error will be returned indicating that a limit must be specified.
Offset	integer			Must be greater than 0	The offset value is paired with a limit parameter to define a window of results. When a limit is specified without an offset, the offset is treated as 0. This means that the results will contain the first record, up to the limit of items requested. To get the next 'page', a new offset must be specified.
Format	String	json		json, csv, xml	When paging is in effect, and JSON is being returned, the results will include predefined links which can be used to get the next, previous, first, and last pages based on the limit and offset values given in the request. The result of the issuers endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one

Name	Type	Mandatory	Default value	Accepted values	Description
fund_metrics_coverage_only	String			True, false	<p>or more rows of data. XML can also be specified and the result will be a properly formatted XML document without a DTD or schema.</p> <p>The 'fund_metrics_coverage_only' flag is supposed to limit results to funds in fund metrics coverage if set to true. If you want to have fund which are not part of coverage you need to add 'fund_metrics_coverage_only=false' in the API query. By default fund endpoint fund_metrics_coverage_only is set to true. By default fund endpoint fund_metrics_coverage_only is set to true. If you set it to false, it will find funds that aren't in the fund metrics universe but not include those that are in the universe.</p>

RESPONSE

Possible Outputs

Status	Description
200	OK
206	Partial Content may have been returned
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not found
500	Internal Server Error

Output Encoding

application/xml, application/json, text/csv

Output Description

The fund endpoint returns a collection of data for the fund and datapoints requested. By default, the result is returned in JSON format. The results can also be returned as a CSV file if the request uses the format parameter, or if the accept header specifies text/csv as an acceptable result.

Sample Output

Possible outputs for status: 200

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "9d19a15ff2e4832a",
  "timestamp": "2019-03-15T21:22:11Z",
  "messages": [],
  "result": {
    "funds": [
      {
        "FUND_NAME": "Parnassus Fund;Investor",
        "FUND_SHARE_CLASS_ID": "40001494",
        "FUND_ID": "35529804",
        "FUND_ISIN": "US7017651099",
        "FUND_TICKER": "PARNX",
        "FUND_ESG_QUALITY_SCORE_PCTL_GLOBAL": 41.547134356404634,
        "FUND_IN_UNIVERSE": true
      },
      {
        "FUND_NAME": "Fidelity Advisor Capital Development Fund;O",
        "FUND_SHARE_CLASS_ID": "40001683",
        "FUND_ID": "35530080",
        "FUND_ISIN": "US3161272089",
        "FUND_TICKER": "FDETX",
        "FUND_ESG_QUALITY_SCORE_PCTL_GLOBAL": 37.2439711869715,
        "FUND_IN_UNIVERSE": true
      },
      {
        "FUND_NAME": "JPMorgan US Equity Fund;I",
        "FUND_SHARE_CLASS_ID": "40001538",
        "FUND_ID": "36450272",
        "FUND_ISIN": "US4812A11594",
        "FUND_TICKER": "JUESX",
        "FUND_ESG_QUALITY_SCORE_PCTL_GLOBAL": 48.62198559348575,
        "FUND_IN_UNIVERSE": true
      }
    ]
  }
}
```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/xml" -d "{}" -X POST https://api.msci.com/esg/data/v3.0/funds
```

Python

```
import requests import json url = "https://api.msci.com/esg/data/v3.0/funds"; payload =
json.dumps( {} ) headers = { "Accept": "application/xml" } response =
requests.request("POST", url, data=payload, headers=headers) print(response.text)
```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse response = Unirest.post("https://api.msci.com/esg/data/v3.0/funds")
.header("Accept", "application/xml") .body("{}") .asString();
```

C#

```
var client = new RestClient("https://api.msci.com/esg/data/v3.0/funds"); var request = new
RestRequest(Method.POST); request.AddHeader("Accept", "application/xml"); request.AddParameter("application/json", "{}",
ParameterType.RequestBody); IRestResponse response = client.Execute(request);
```

/FUNDS/HISTORY

The /funds/history endpoint provides a mechanism to retrieve ESG data history for a given set of fund identifiers. A user of the API can send in a list of fund identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

When making a POST request, a proper Content-Type must be specified. Since the API utilizes JSON for POST requests, the Content-Type should be application/json. If the Content-Type header is not properly specified, the request may be rejected with a 400 status code and an 'Invalid formatting' error message.

Depending on the amount of data being returned, multiple requests may be required. An initial request will return the first batch of data, along with information about whether more batches are available. If more batches are available, the user should choose one of the additional data batch request services documented below.

When requesting ESG history data, it is possible to create a request which returns a large amount of data. When a request is made for history data, the API will determine the size of the total result. Based on the calculated size, the API will determine the total number of batches necessary to return all of the requested data. Every response returned from the history API endpoint contains a response_metadata property. A typical response will look something like this:

```
"response_metadata": {
  "total_number_of_identifiers": 150,
  "total_number_of_batches": 3,
  "current_batch": 1,
  "data_request_id": "bb005730-262c-4d1c-88bb-acc916ee0ed7"
}
```

In this example, a request was made for history data. The response is indicating that there will be 150 funds in the response, broken into 3 batches. The current_batch property indicates that this is the first batch in the result.

To retrieve additional batches, additional calls are made to /funds/history, passing the following parameters instead of all the parameters sent in the first request:

Property Name	Description
data_request_id	This is an identifier that is assigned to a data request. It acts as a key to retrieve additional batches of information. The key remains valid for as long as data is being requested, but will expire at some point and cannot be used for caching purposes. While calling get endpoint use encoding as the request_id has special characters (code sample below), Note: The POST method should be the method used by most API applications.
batch_id	This is a number used to identify which batch of data to retrieve.

Note that this method of retrieving data does not involve “paging” through the data. Once an initial request is made, the client can make multiple parallel calls to retrieve different batches of data. Also, each request which utilizes the data_request_id will be idempotent.

The `data_request_id` is a unique identifier which can be re-used for some period of time to refer to the same request. After the first request, the client will know how many batches exist, and if desired can generate parallel requests to get subsequent batches of data. Each batch will contain a complete set of data for a requested identifier. There are no order dependencies in the response. After some set period of time, currently set to 24 hours after the first request, the `data_request_id` will expire and requests for batches using the given id will no longer return a response.

Below code sample for encoding in Java/Python

Java:

```
import java.net.URLEncoder;
```

url =

```
"https://api.msci.com/esg/data/v3.0/funds/history?batch_id=2&data_request_id="+URLEncoder.encode(data_request_id,
StandardCharsets.UTF_8.toString());
```

Python:

```
from urllib.parse import urlencode
urlencode(dict(data_request_id=data_request_id,batch_id=batch_id))
```

• RETRIEVE ESG DATA HISTORY FOR REQUESTED IDENTIFIERS

The `/funds/history` endpoint provides a mechanism to retrieve ESG data history for a given set of fund identifiers. A user of the API can send in a list of fund identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

The API accepts two types of input parameters to the request, one form of input initiates a new retrieval request, the other is used to return subsequent batches of data. The `/funds/history` endpoint uses a dynamic batching scheme to return data if the results are too large to return in one request.

HTTP Method:

GET

ENDPOINT DESCRIPTION

The `/funds/history` endpoint provides a mechanism to retrieve ESG data history for a given set of fund identifiers. A user of the API can send in a list of fund identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned. The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

The API accepts two types of input parameters to the request, one form of input initiates a new retrieval request, the other is used to return subsequent batches of data. The `/funds/history` endpoint uses a dynamic batching scheme to return data if the results are too large to return in one request (see the section on data batching for more details on how large results are returned)

The basic request for data will include the following information:

- A list of one or more identifiers. These identifiers will be used to locate companies (funds) who have ESG Data. The identifiers can represent Bonds, Securities, or the funds themselves.
- A list of data factors. The factors can be the names of the factors themselves. Factor information is available through an API endpoint “/metadata/history/factors”.
- Select a date range with a maximum duration of one year.
- Other parameters to fine tune the request.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/funds/history>

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description				
format	string		json	json, csv	The result of the fund history endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. The CSV format is supported with only data_layout parameter value "by_factor".				
start_date	string				The first date to be used for data retrieval. If only a start_date is given, only data for the specified date will be returned. Format - date (as full-date in RFC3339).				
fund_identifier_list	Array				A list of one or more fund identifiers which resolve to funds. Identifiers can be MDSUIDS, etc.				
inherit_missing_values	boolean				A Boolean value used to indicate whether the API should inherit missing values from a parent fund. This option will automatically be set to true if the caller is permitted to receive inherited data. This option is not available to clients who do not have the inheritance feature associated with their account.				
data_layout	String				<div>The API offers different data layouts to suit the different ways that a client may want to process the data that they receive. The Data Layout options provide a way for the client to tailor the output to their needs</div> <table><tr><th>Data Layout Name</th><th>Description</th></tr><tr><td>by_factor</td><td>The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order</td></tr></table>	Data Layout Name	Description	by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order
Data Layout Name	Description								
by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order								

Name	Type	Mandatory	Default value	Accepted values	Description
					<div>matrix_by_factor</div> <div>This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.</div>
					<div>change_ranges</div> <div>This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.</div>
<p>The API treats historical data as a continuous stream of values for the date range specified in the call. The user can choose how often they would like to sample the stream of data. Perhaps the client only cares about what the value is at the end of each month, for example. The following options are available for the Data Sample Frequency parameter:</p>					
data_sample_frequency	String				<div>Data Sample Name</div> <div>daily</div> <div>A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.</div>
					<div>business_month_end</div> <div>A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for</div>

Name	Type	Mandatory	Default value	Accepted values	Description
					<div> <div></div> <div>each business month end day in the range</div> </div>
				calendar_month_end	<div> <div></div> <div>A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range</div> </div>
as_at_date	String				<p>A date which defines which corrections should be returned. If the as_at_date is before the end_date, the as_at_date will override the end_date. No values will be returned beyond the specified as_at_date. If no as_at_date is specified, the data will be viewed from the perspective of the date that the request was made. The Historical Data API returns data as published by MSCI ESG Research, including normal data updates and data corrections. If and when MSCI ESG Research corrects a data error, the Historical Data API returns the corrected value starting with the date the correction was made in our database. As a general rule, MSCI ESG Research does not restate historical data. The response using as_at_date will be the same as that of the response using as_of_date. Please see the MSCI ESG Research Data Correction Policy for more information: https://support.msci.com/sustainability_climate/data-correction-policy. Format - date (as full-date in RFC3339).</p>
end_date	String				<p>The end date for the range of values to be returned. If not specified, the request will return a single value corresponding to the given start_date. The end_date cannot exceed the as_at_date. If the end_date is after the as_at_date, the request will be rejected with a 400 status code. Format - date (as full-date in RFC3339).</p>
factor_name_list	Array				<p>A list of ESG Factor names as defined in the ESG Factor Metadata. Factor names, product names, and category paths can be mixed together, but at least one factor name, category path, or product name must be specified for a request to be valid.</p>
historical_product	String				<p>The Funds historical product name for which historical data is to be retrieved. Only a single</p>

Name	Type	Mandatory	Default value	Accepted values	Description
					product name is accepted per request. All available products can be retrieved at /metadata/history/historicalProducts. Cannot be used along with factor_name_list or dataset_list. CSV Format is not supported with historical_product parameter.
					User is only allowed to fetch one day data when using 'daily - by_factor' and 'daily - matrix_by_factor' layout with 'historical_product' parameter.
dataset_list	Array				A list of Datasets as defined in /metadata/history/datasets. dataset_list Should be used along with fund_identifier_list to retrieve historical data for specified identifiers and factors belonging to the datasets. CSV Format is not supported with dataset_list parameter.

RESPONSE

Possible Outputs

Status	Description
200	Success
206	Partial Content may have been returned
400	Malformed request
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

RESPONSE BODY

Name	Description
total_number_of_identifiers	Total number of identifiers returned.

Name	Description
total_number_of_unresolved_identifiers	Total number of identifiers that were not able to be resolved , An identifier is only unresolved if it did not exist at all during the specified time range.
total_number_of_batches	Total number of result batches.
current_batch	The index of the current batch.
data_request_id	The id representing this request. It can be used in future requests to retrieve subsequent batches of data.
data_request_id_expiration_time	This field specifies the expiration time of data_request_id.
unresolved_identifiers	A list of Unresolved Identifier objects.The Unresolved Identifier object is used to hold information about identifiers that were passed as parameters but were unable to be resolved for some reason and therefore data will not be returned.An identifier is only unresolved if it did not exist at all during the specified time range.
unresolved_factors	The Unresolved Factor object is used to hold information about factors that were passed as parameters but were not valid for some reason.
resolved_factors	A list of resolved Factor objects.
resolved_series_factors	A list of resolved series Factor objects.
identifiers_data_not_available	A list of requested identifiers for which all the requested factors data are not available.
factors_data_not_available	A list of factors for which data is not available for any of the requested identifiers.
messages[]	factor permissions will come under messages only in messages[] tag.

Output Encoding

application/json, text/csv

Output Description

All responses from the API are returned in JSON format. All responses have a similar structure but at the core of any response is the data property.

Sample Output

Possible outputs for status: 200

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fee6825430a69436fd75c81a948a5",
  "timestamp": "2024-04-29T19:00:56Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 2,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "2-5945cb06-7d96-4721-a7f1-ea33b6b0e5c9-2024-04-30T19:00:56Z",
      "data_request_id_expiration_time": "2024-04-30 19:00:56.296783791",
      "resolved_factors": [
        "FUND_ACCOUNTING_FLAG_PCT",
        "FUND_CUSIP"
      ],
      "identifiers_data_not_available": [
        "UFD000000001449170"
      ],
      "factors_data_not_available": [
        "FUND_CUSIP"
      ]
    },
    "data": {
      "requested_ids": [
        "UFD000000001162834",
        "UFD000000001449170"
      ],
      "as_at_date": "2024-04-29T19:00:00Z",
      "as_of_dates": [
        "2023-10-01",
        "2023-10-02",
        "2023-10-03",
        "2023-10-04"
      ],
      "factors": [
        {
```

```

    "factor": "FUND_ACCOUNTING_FLAG_PCT",
    "values": [
      [
        [
          null,
          null,
          "1.816067025341814",
          "1.816067025341814"
        ]
      ],
      [
        [
          null,
          null,
          null,
          null
        ]
      ]
    ],
    {
      "factor": "FUND_CUSIP",
      "values": [
        [
          [
            null,
            null,
            null,
            null
          ]
        ],
        [
          [
            null,
            null,
            null,
            null
          ]
        ]
      ]
    }
  ]

```

CODE SAMPLES

```
curl --request GET \
  --url
'https://api.msici.com/esg/data/v3.0/funds/history?start_date=json&fund_identifier_list=json&format=json&inherit_missing_values=SOME_BOOLEAN_VALUE&data_layout=json&data_sample_frequency=json&as_at_date=json&end_date=json&factor_name_list=json'
```

```
import http.client
conn = http.client.HTTPSConnection("api.msci.com")
headers = { 'Authorization': "Replace with your Authorization Token" }

conn.request("GET",
"/esg/data/v3.0/funds/history?start_date=json&fund_identifier_list=json&format=json&inherit_m
issing_values=SOME_BOOLEAN_VALUE&data_layout=json&data_sample_frequency=json&as_at_date=json&
end_date=json&factor_name_list=json", headers=headers)

res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse<String> response =
Unirest.get("https://api.msci.com/esg/data/v3.0/funds/history?start_date=json&fund_identifier_list=json&format=json&inherit_missing_values=SOME_BOOLEAN_VALUE&data_layout=json&data_sample_frequency=json&as_at_date=json&end_date=json&factor_name_list=json")
    .header("Authorization", "Replace with your Authorization Token ")
    .asString();
```

```
var client = new
RestClient("https://api.msci.com/esg/data/v3.0/funds/history?start_date=json&fund_identifier_
list=json&format=json&inherit_missing_values=SOME_BOOLEAN_VALUE&data_layout=json&data_sample_
frequency=json&as_at_date=json&end_date=json&factor_name_list=json");

var request = new RestRequest(Method.GET);

request.AddHeader("Authorization", Replace with your Authorization Token");

IRestResponse response = client.Execute(request);
```

POST

ENDPOINT DESCRIPTION

The /funds/history endpoint provides a mechanism to retrieve ESG data history for a given set of fund identifiers. A user of the API can send in a list of fund identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

The API accepts two types of input parameters to the request, one form of input initiates a new retrieval request, the other is used to return subsequent batches of data. The /funds/history endpoint uses a dynamic batching scheme to return data if the results are too large to return in one request (see the section on data batching for more details on how large results are returned)

The basic request for data will include the following information:

- A list of one or more identifiers. These identifiers will be used to locate companies (funds) who have ESG Data. The identifiers can represent MDSUID funds themselves.
- A list of data factors. The factors can be the names of the factors themselves. Factor information is available through an API endpoint /metadata/history/factors.
- Select a date range with a maximum duration of one year.
- Other parameters to fine tune the request.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/funds/history>

Body payload

Name	Type	Mandatory	Default value	Accepted values	Description
format	string		json	json, csv	The result of the fund history endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. The CSV format is supported with only data_layout parameter value "by_factor".
start_date	string				The first date to be used for data retrieval. If only a start_date is given, only data for the specified date will be returned. Format - date (as full-date in RFC3339).
fund_identifier_list	Array				A list of one or more fund identifiers which resolve to funds. Identifiers can be MDSUIDS, etc.
inherit_missing_values	boolean				A Boolean value used to indicate whether the API should inherit missing values from a parent fund. This option will automatically be set to true if the caller is permitted to receive inherited data.

Name	Type	Mandatory	Default value	Accepted values	Description								
data_layout	String				<p>The API offers different data layouts to suit the different ways that a client may want to process the data that they receive. The Data Layout options provide a way for the client to tailor the output to their needs.</p> <table><tr><th>Data Layout Name</th><th>Description</th></tr><tr><td>by_factor</td><td>The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order</td></tr><tr><td>matrix_by_factor</td><td>This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.</td></tr><tr><td>change_ranges</td><td>This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.</td></tr></table>	Data Layout Name	Description	by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order	matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.	change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.
		Data Layout Name	Description										
		by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order										
		matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.										
change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.												
data_sample_frequency	String				<p>The API treats historical data as a continuous stream of values for the date range specified in the call. The user can choose how often they would like to sample the stream of data. Perhaps the client only cares about what the value is at the end of each month, for example. The following options are available for the Data Sample Frequency parameter:</p> <table><tr><th>Data Sample Name</th><th>Description</th></tr><tr><td>daily</td><td>A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.</td></tr></table>	Data Sample Name	Description	daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.				
Data Sample Name	Description												
daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.												

Name	Type	Mandator y	Defaul t value	Accepte d values	Description
					<div>business_month_end</div> <p>A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range</p>
					<div>calendar_month_end</div> <p>A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range</p>
as_at_date	String				<p>A date which defines which corrections should be returned. If the as_at_date is before the end_date, the as_at_date will override the end_date. No values will be returned beyond the specified as_at_date. If no as_at_date is specified, the data will be viewed from the perspective of the date that the request was made. The Historical Data API returns data as published by MSCI ESG Research, including normal data updates and data corrections. If and when MSCI ESG Research corrects a data error, the Historical Data API returns the corrected value starting with the date the correction was made in our database. As a general rule, MSCI ESG Research does not restate historical data. The response using as_at_date will be the same as that of the response using as_of_date. Please see the MSCI ESG Research Data Correction Policy for more information: https://support.msci.com/sustainability_climate/data-correction-policy. Format - date (as full-date in RFC3339).</p>
end_date	String				<p>The end date for the range of values to be returned. If not specified, the request will return a single value corresponding to the given start_date. The end_date cannot exceed the as_at_date. If the end_date is after the as_at_date, the request will be rejected with a 400 status code. Format - date (as full-date in RFC3339).</p>
factor_name_list	Array				<p>A list of ESG Factor names as defined in the ESG Factor Metadata. Factor names, product names, and category paths can be mixed together, but at least one factor name, category path, or product name must be specified for a request to be valid.</p>
historical_product	String				<p>The Funds historical product name for which historical data is to be retrieved. Only a single product name is accepted per request. All available products can be retrieved at /metadata/history/historicalProducts. Cannot be used along with factor_name_list or dataset_list. CSV Format is not supported with historical_product parameter.</p>

Name	Type	Mandator y	Defaul t value	Accepte d values	Description
					User is only allowed to fetch one day data when using 'daily - by_factor' and 'daily - matrix_by_factor' layout with 'historical_product' parameter.
dataset_list	Array				A list of Datasets as defined in /metadata/history/datasets. dataset_list Should be used along with fund_identifier_list to retrieve historical data for specified identifiers and factors belonging to the datasets. CSV Format is not supported with dataset_list parameter.

RESPONSE

Possible Outputs

Status	Description
200	OK
206	Partial Content may have been returned
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not found
500	Internal Server Error

RESPONSE BODY

Name	Description
total_number_of_identifiers	Total number of identifiers returned.
total_number_of_unresolved_identifiers	Total number of identifiers that were not able to be resolved , An identifier is only unresolved if it did not exist at all during the specified time range.
total_number_of_batches	Total number of result batches.
current_batch	The index of the current batch.
data_request_id	The id representing this request. It can be used in future requests to retrieve subsequent batches of data.

Name	Description
data_request_id_expiration_time	This field specifies the expiration time of data_request_id.
unresolved_identifiers	A list of Unresolved Identifier objects.The Unresolved Identifier object is used to hold information about identifiers that were passed as parameters but were unable to be resolved for some reason and therefore data will not be returned.An identifier is only unresolved if it did not exist at all during the specified time range.
unresolved_factors	The Unresolved Factor object is used to hold information about factors that were passed as parameters but were not valid for some reason.
resolved_factors	A list of resolved Factor objects.
resolved_series_factors	A list of resolved series Factor objects.
identifiers_data_not_available	A list of requested identifiers for which all the requested factors data are not available.
factors_data_not_available	A list of factors for which data is not available for any of the requested identifiers.
messages[]	factor permissions will come under messages only in messages[] tag.

Output Encoding

application/xml, application/json, text/csv

Output Description

All responses from the API are returned in JSON format. All responses have a similar structure but at the core of any response is the data property.

Sample Output

Possible outputs for status: 200

by_factor layout:

Input

```
{
  "fund_identifier_list": [
    "UFD000000001162834"
  ],
  "factor_name_list": [
```

```

        "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV",
        "FUND_AFFORD_HOUSING_INVOLV_PCT",
        "FUND_SEDOL"
    ],
    "data_layout" : "by_factor",
    "start_date": "2023-10-27",
    "end_date": "2023-12-07",
    "data_sample_frequency": "business_month_end"
}

```

Output

```

{
  "status": "OK",
  "code": 200,
  "trace_id": "662faf994ee80076fed195319e125c79",
  "timestamp": "2024-04-29T14:32:57Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 1,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "0-85ca6373-90fd-4b08-8b60-497b592a3843-2024-04-30T14:32:57Z",
      "data_request_id_expiration_time": "2024-04-30 14:32:57.867351987",
      "resolved_factors": [
        "FUND_AFFORD_HOUSING_INVOLV_PCT",
        "FUND_SEDOL",
        "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV"
      ]
    },
    "data": [
      {
        "requested_id": "UFD000000001162834",
        "factors": [
          {
            "name": "FUND_SEDOL",
            "data_values": [

```

```

        "value": "B0L12X8",
        "as_of_date": "2023-10-31",
        "as_at_date": "2024-04-29T14:32:00Z"
    },
    {
        "value": "B0L12X8",
        "as_of_date": "2023-11-30",
        "as_at_date": "2024-04-29T14:32:00Z"
    }
]
},
{
    "name": "FUND_AFFORD_HOUSING_INVOLV_PCT",
    "data_values": [
        {
            "value": 3.073290528863116,
            "as_of_date": "2023-10-31",
            "as_at_date": "2024-04-29T14:32:00Z"
        },
        {
            "value": 3.006817309241662,
            "as_of_date": "2023-11-30",
            "as_at_date": "2024-04-29T14:32:00Z"
        }
    ]
},
{
    "name": "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV",
    "data_values": [
        {
            "value": 0.46781023,
            "as_of_date": "2023-10-31",
            "as_at_date": "2024-04-29T14:32:00Z"
        },
        {
            "value": 0.44978977,
            "as_of_date": "2023-11-30",
            "as_at_date": "2024-04-29T14:32:00Z"
        }
    ]
}

```

change_ranges layout:

Input

Output

© 2019 MSCI Inc. All rights reserved.

```

    "data_request_id": "1-47955461-f3a0-435f-af0c-8e8cbb8daf6f-2024-04-30T14:34:48Z",
    "data_request_id_expiration_time": "2024-04-30 14:34:48.753549368",
    "resolved_factors": [
      "FUND_AFFORD_HOUSING_INVOLV_PCT",
      "FUND_SEDOL",
      "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV"
    ]
  },
  "data": [
    {
      "requested_id": "UFD000000001162834",
      "factors": [
        {
          "name": "FUND_SEDOL",
          "data_values": [
            {
              "value": "B0L12X8",
              "as_of_date": "2023-10-27",
              "valid_until_date": "2023-12-07",
              "as_at_date": "2024-04-29T14:34:00Z"
            }
          ]
        },
        {
          "name": "FUND_AFFORD_HOUSING_INVOLV_PCT",
          "data_values": [
            {
              "value": 3.073290528863117,
              "as_of_date": "2023-10-27",
              "valid_until_date": "2023-10-31",
              "as_at_date": "2024-04-29T14:34:00Z"
            },
            {
              "value": 3.073290528863116,
              "as_of_date": "2023-10-31",
              "valid_until_date": "2023-11-01",
              "as_at_date": "2024-04-29T14:34:00Z"
            },
            {
              "value": 3.073290528863117,

```



```

      "as_of_date": "2023-11-01",
      "valid_until_date": "2023-11-07",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 3.006817309241662,
      "as_of_date": "2023-11-07",
      "valid_until_date": "2023-11-15",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 3.006817309241661,
      "as_of_date": "2023-11-15",
      "valid_until_date": "2023-11-16",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 3.006817309241662,
      "as_of_date": "2023-11-16",
      "valid_until_date": "2023-11-17",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 3.006817309241661,
      "as_of_date": "2023-11-17",
      "valid_until_date": "2023-11-20",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 3.006817309241662,
      "as_of_date": "2023-11-20",
      "valid_until_date": "2023-11-22",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 3.006817309241661,
      "as_of_date": "2023-11-22",
      "valid_until_date": "2023-11-23",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {

```

```

{
  "value": 3.006817309241662,
  "as_of_date": "2023-11-23",
  "valid_until_date": "2023-12-07",
  "as_at_date": "2024-04-29T14:34:00Z"
}
],
{
  "name": "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV",
  "data_values": [
    {
      "value": 0.46781023,
      "as_of_date": "2023-10-27",
      "valid_until_date": "2023-11-07",
      "as_at_date": "2024-04-29T14:34:00Z"
    },
    {
      "value": 0.44978977,
      "as_of_date": "2023-11-07",
      "valid_until_date": "2023-12-07",
      "as_at_date": "2024-04-29T14:34:00Z"
    }
  ]
}
]
}
]
}
}

```

matrix by factor layout:

Input

```

{
  "fund_identifier_list": [
    "UFD000000001162834"
  ]
}

```

```

],
"factor_name_list": [
    "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV",
    "FUND_AFFORD_HOUSING_INVOLV_PCT",
    "FUND_SEDOL"
],
"data_layout" : "matrix_by_factor",
"start_date": "2023-11-27",
"end_date": "2023-12-07",
"data_sample_frequency": "daily"
}

```

Output

```

{
  "status": "OK",
  "code": 200,
  "trace_id": "662fb105423345c0dbfb4f7b8d6394be",
  "timestamp": "2024-04-29T14:39:01Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 1,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "2-3c8f336e-65ca-4e90-88fa-1ef1efe3cbba-2024-04-30T14:39:01Z",
      "data_request_id_expiration_time": "2024-04-30 14:39:01.145055775",
      "resolved_factors": [
        "FUND_AFFORD_HOUSING_INVOLV_PCT",
        "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV",
        "FUND_SEDOL"
      ]
    },
    "data": {
      "requested_ids": [
        "UFD000000001162834"
      ],
      "as_at_date": "2024-04-29T14:39:00Z",
      "as_of_dates": [

```

```

    "2023-11-27",
    "2023-11-28",
    "2023-11-29",
    "2023-11-30",
    "2023-12-01",
    "2023-12-02",
    "2023-12-03",
    "2023-12-04",
    "2023-12-05",
    "2023-12-06"
  ],
  "factors": [
    {
      "factor": "FUND_SEDOL",
      "values": [
        [
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8",
          "B0L12X8"
        ]
      ]
    }
  ],
  {
    "factor": "FUND_REV_EXP_EST_EU_TAXONOMY_MAX_REV",
    "values": [
      [
        "0.44978977",
        "0.44978977",
        "0.44978977",
        "0.44978977"
      ]
    ]
  }

```

```

        "0.44978977",
        "0.44978977",
        "0.44978977",
        "0.44978977",
        "0.44978977",
        "0.44978977"
    ]
}
]
},
{
    "factor": "FUND_AFFORD_HOUSING_INVOLV_PCT",
    "values": [
        [
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662",
            "3.006817309241662"
        ]
    ]
}
]
}
}
}

```

CODE SAMPLES

cURL

```

curl --request POST \
  --url https://api.msci.com/esg/data/v3.0/funds/history \
  --header 'Authorization:Replace with your Auth token' \
  --header 'content-type: application/json' \

```

```
--data
'{"fund_identifier_list":["string"],"factor_name_list":["string"],"start_date":"2019-08-24","end_date":"2019-08-24","as_at_date":"2019-08-24","data_sample_frequency":"string","data_layout":"string","inherit_missing_values":true,"format":"string"}
```

Python

```
import http.client
conn = http.client.HTTPSConnection("api.msci.com")
payload =
'{"fund_identifier_list":["string"],"factor_name_list":["string"],"start_date":"2019-08-24","end_date":"2019-08-24","as_at_date":"2019-08-24","data_sample_frequency":"string","data_layout":"string","inherit_missing_values":true,"format":"string"}'
headers = {
    'Authorization': " Replace with your Auth token",
    'content-type': "application/json"
}
conn.request("POST", "/esg/data/v3.0/funds/history", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse<String> response =
Unirest.post("https://api.msci.com/esg/data/v3.0/funds/history")
    .header("Authorization", " Replace with your Auth token ")
    .header("content-type", "application/json")
    .body("{\"fund_identifier_list\":[\"string\"],\"factor_name_list\":[\"string\"],\"start_date\":"2019-08-24\", \"end_date\":"2019-08-24\", \"as_at_date\":"2019-08-24\", \"data_sample_frequency\":"string\", \"data_layout\":"string\", \"inherit_missing_values\":true, \"format\":"string\"}")
    .asString();
```

C#

```
var client = new RestClient("https://api.msci.com/esg/data/v3.0/funds/history");
request.AddHeader("Authorization", " Replace with your Auth token ");
request.AddHeader("content-type", "application/json");
request.AddParameter("application/json",
"{"fund_identifier_list":["string"],"factor_name_list":["string"],"start_date":"2019-08-24","end_date":"2019-08-24","as_at_date":"2019-08-24","data_sample_frequency":"string","data_layout":"string","inherit_missing_values":true,"format":"string"}", ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
```

/INSTRUMENTS/HISTORY

The /instruments/history endpoint provides a mechanism to retrieve ESG data history for a given set of instruments identifiers. A user of the API can send in a list of instruments identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

When making a POST request, a proper Content-Type must be specified. Since the API utilizes JSON for POST requests, the Content-Type should be application/json. If the Content-Type header is not properly specified, the request may be rejected with a 400 status code and an 'Invalid formatting' error message.

Depending on the amount of data being returned, multiple requests may be required. An initial request will return the first batch of data, along with information about whether more batches are available. If more batches are available, the user should choose one of the additional data batch request services documented below.

When requesting ESG history data, it is possible to create a request which returns a large amount of data. When a request is made for history data, the API will determine the size of the total result. Based on the calculated size, the API will determine the total number of batches necessary to return all of the requested data. Every response returned from the history API endpoint contains a response_metadata property. A typical response will look something like this:

```
"response_metadata": {
  "total_number_of_identifiers": 300,
  "total_number_of_batches": 3,
  "current_batch": 1,
  "data_request_id": "bb005730-262c-4d1c-88bb-acc916ee0ed7"
}
```

In this example, a request was made for history data. The response is indicating that there will be 300 instruments in the response, broken into 3 batches. The current_batch property indicates that this is the first batch in the result.

To retrieve additional batches, additional calls are made to /instruments/history, passing the following parameters instead of all the parameters sent in the first request:

Property Name	Description
data_request_id	This is an identifier that is assigned to a data request. It acts as a key to retrieve additional batches of information. The key remains valid for as long as data is being requested, but will expire at some point and cannot be used for caching purposes. While calling get endpoint use encoding as the request_id has special characters (code sample below), Note: The POST method should be the method used by most API applications.
batch_id	This is a number used to identify which batch of data to retrieve.

Note that this method of retrieving data does not involve “paging” through the data. Once an initial request is made, the client can make multiple parallel calls to retrieve different batches of data. Also, each request which utilizes the data_request_id will be idempotent.

The data_request_id is a unique identifier which can be re-used for some period of time to refer to the same request. After the first request, the client will know how many batches exist, and if desired can generate parallel requests to get subsequent batches of data. Each batch will contain a complete set of data for a requested identifier. There are no order dependencies in the response. After some set period of time, currently set to 24 hours after the first request, the data_request_id will expire and requests for batches using the given id will no longer return a response.

Below code sample for encoding in Java/Python

Java:
import java.net.URLEncoder;

```
url = "https://api.msci.com/esg/data/v3.0/instruments
/history?batch_id=2&data_request_id="+URLEncoder.encode(data_request_id, StandardCharsets.UTF_8.toString());
```

Python:

```
from urllib.parse import urlencode
urlencode(dict(data_request_id=data_request_id,batch_id=batch_id))
```

RETRIEVE ESG DATA HISTORY FOR REQUESTED IDENTIFIERS

The /instruments/history endpoint provides a mechanism to retrieve ESG data history for a given set of instrument identifiers. A user of the API can send in a list of instrument identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

The API accepts two types of input parameters to the request, one form of input initiates a new retrieval request, the other is used to return subsequent batches of data. The /instruments/history endpoint uses a dynamic batching scheme to return data if the results are too large to return in one request (see the section on data batching for more details on how large results are returned).

HTTP Method:

GET

ENDPOINT DESCRIPTION

The /instruments/history endpoint provides a mechanism to retrieve ESG data history for a given set of instrument identifiers. A user of the API can send in a list of instrument identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The basic request for data will include the following information:

- A list of one or more identifiers. These identifiers will be used to locate bonds, equities etc who has ESG Data.
- A list of data factors. The factors can be the names of the factors themselves. Factor information is available through other API endpoints such as /metadata/history/factors.
- A date range. allowed date range must be within a 1 year. the API supports date ranges of 1 date to many dates. All dates are in the format yyyy-mm-dd, i.e. 2020-04-01 would be April 1, 2020.
- Other parameters to fine tune the request.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/instruments/history>

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
format	string		json	json, csv	The result of the instruments endpoint will be returned in JSON format by default. However, the result can also be

Name	Type	Mandatory	Default value	Accepted values	Description								
					returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. The CSV format is supported with only data_layout parameter value "by_factor".								
start_date	string				The first date to be used for data retrieval. If only a start_date is given, only data for the specified date will be returned. Format - date (as full-date in RFC3339).								
factor_name_list	Array				A list of ESG Factor names as defined in the ESG Factor Metadata. Factor names, but at least one factor name must be specified for a request to be valid.								
					The API offers different data layouts to suit the different ways that a client may want to process the data that they receive. The Data Layout options provide a way for the client to tailor the output to their needs.								
					<table><tr><th>Data Layout Name</th><th>Description</th></tr><tr><td>by_factor</td><td>The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order</td></tr><tr><td>matrix_by_factor</td><td>This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. This layout is compatible with both daily and business_month_end sample frequencies.</td></tr><tr><td>change_ranges</td><td>This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.</td></tr></table>	Data Layout Name	Description	by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order	matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. This layout is compatible with both daily and business_month_end sample frequencies.	change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.
Data Layout Name	Description												
by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order												
matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. This layout is compatible with both daily and business_month_end sample frequencies.												
change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.												
data_layout	String												
					The API treats historical data as a continuous stream of values for the date range specified in the call. The user can choose how often they would like to sample the stream of data. Perhaps the client only cares about what the value is at the end of each month, for example. The following options are available for the Data Sample Frequency parameter:								
data_sample_frequency	String												

Name	Type	Mandatory	Default value	Accepted values	Description								
					<table><tr><th>Data Sample Name</th><th>Description</th></tr><tr><td>daily</td><td>A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.</td></tr><tr><td>business_month_end</td><td>A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range</td></tr><tr><td>calendar_month_end</td><td>A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range</td></tr></table>	Data Sample Name	Description	daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.	business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range	calendar_month_end	A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range
Data Sample Name	Description												
daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.												
business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range												
calendar_month_end	A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range												
as_at_date	String				A date which defines which corrections should be returned. If the as_at_date is before the end_date, the as_at_date will override the end_date. No values will be returned beyond the specified as_at_date. If no as_at_date is specified, the data will be viewed from the perspective of the date that the request was made. The Historical Data API returns data as published by MSCI ESG Research, including normal data updates and data corrections. If and when MSCI ESG Research corrects a data error, the Historical Data API returns the corrected value starting with the date the correction was made in our database. As a general rule, MSCI ESG Research does not restate historical data. The response using as_at_date will be the same as that of the response using as_of_date. Please see the MSCI ESG Research Data Correction Policy for more information: https://support.msci.com/sustainability_climate/data-correction-policy . Format - date (as full-date in RFC3339).								
end_date	String				The end date for the range of values to be returned. If not specified, the request will return a single value corresponding to the given start_date. The end_date cannot exceed the as_at_date. If the end_date is after the as_at_date, the request will be rejected with a 400 status code. Format - date (as full-date in RFC3339).								
instrument_identifier_list	Array				A list of one or more instrument identifiers which resolve to issuers. Identifiers can be ISINs, CUSIPs, MDS UIDS, etc.								

Name	Type	Mandatory	Default value	Accepted values	Description
historical_product	String				<p>The Instrument historical product name for which historical data is to be retrieved. Only a single product name is accepted per request. All available products can be retrieved at /metadata/history/historicalProducts. Cannot be used along with factor_name_list or dataset_list. CSV Format is not supported with historical_product parameter.</p> <p>User is only allowed to fetch one day data when using 'daily - by_factor' and 'daily - matrix_by_factor' layout with 'historical_product' parameter.</p>
dataset_list	Array				<p>A list of Datasets as defined in /metadata/history/datasets. dataset_list Should be used along with instrument_identifier_list to retrieve historical data for specified identifiers and factors belonging to the datasets. CSV Format is not supported with dataset_list parameter.</p>

RESPONSE

Possible Outputs

Status	Description
200	Success
206	Partial Content may have been returned
400	Malformed request
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json, text/csv

Output Description

The instrument endpoint returns a collection of data for the instrument identifiers and datapoints requested. By default, the result is returned in JSON format. The results can also be returned as a CSV file if the request uses the format parameter, or if the accept header specifies text/csv as an acceptable result. And

Sample Output

Possible outputs for status: 200

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "67b42de6c83cf4acdf6b527e7657d846",
  "timestamp": "2025-02-18T06:51:18Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 1,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "0-d1428a46-0dbf-4024-aab2-5aced1d409ff-2025-02-19T06:51:18Z",
      "data_request_id_expiration_time": "2025-02-19T06:51:18+0000",
      "resolved_factors": {
        "BOND": [
          "GB_MSCI_VERIFIED",
          "GB_SELF_LABELLED"
        ]
      }
    },
    "data": [
      {
        "requested_id": "UIF000000027804394",
        "instrument_type": "BOND",
        "factors": [
          {
            "name": "GB_MSCI_VERIFIED",
            "data_values": [
              {
                "value": "Not Assessed",
                "as_of_date": "2023-04-26",
                "as_at_date": "2025-02-18T06:51:00Z"
              }
            ]
          }
        ]
      },
      {
        "name": "GB_SELF_LABELLED",
```

```

    "data_values": [
      {
        "value": "True",
        "as_of_date": "2023-04-26",
        "as_at_date": "2025-02-18T06:51:00Z"
      }
    ]
  }
}

```

CODE SAMPLES

cURL

```

curl --request GET \
  --url
'https://api.msci.com/esg/data/v3.0/instruments/history?start_date=json&factor_name_list=json&format=json&data_layout=json&data_sample_frequency=json&as_at_date=json&end_date=json&instrument_identifier_list=json' \
  --header 'Authorization: Replace with your Authorization Token'

```

Python

```

import http.client
conn = http.client.HTTPSConnection("api.msci.com")
headers = { 'Authorization': " Replace with your Authorization Token" }

conn.request("GET",
"/esg/data/v3.0/instruments/history?start_date=json&factor_name_list=json&format=json&data_layout=json&data_sample_frequency=json&as_at_date=json&end_date=json&instrument_identifier_list=json", headers=headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))

```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```

HttpResponse<String> response =
Unirest.get("https://api.msci.com/esg/data/v3.0/instruments/history?start_date=json&factor_name_list=json&format=json&data_layout=json&data_sample_frequency=json&as_at_date=json&end_date=json&instrument_identifier_list=json")
    .header("Authorization", " Replace with your Authorization Token ")
    .asString();

```

C#

```

var client = new
RestClient("https://api.msci.com/esg/data/v3.0/instruments/history?start_date=json&factor_name_list=json&format=json&data_layout=json&data_sample_frequency=json&as_at_date=json&end_date=json&instrument_identifier_list=json");

var request = new RestRequest(Method.GET);

```

```
request.AddHeader("Authorization", " Replace with your Authorization Token ");
IRestResponse response = client.Execute(request);
```

HTTP Method:

POST

ENDPOINT DESCRIPTION

The /instruments/history endpoint provides a mechanism to retrieve ESG data history for a given set of instrument identifiers. A user of the API can send in a list of instrument identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The basic request for data will include the following information:

- A list of one or more identifiers. These identifiers will be used to locate bonds/equities (instruments) who have ESG Data. The identifiers can represent Bonds, Securities.
- A list of data factors. The factors can be the names of the factors themselves. Factor information is available through other API endpoints such as /metadata/history/factors.
- A date range. allowed date range must be within a 1 year. the API supports date ranges of 1 date to many dates. All dates are in the format yyyy-mm-dd, i.e. 2020-04-01 would be April 1, 2020.
- Other parameters to fine tune the request.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/instruments/history>

Body payload

Name	Type	Mandatory	Default value	Accepted values	Description
format	string		json	json, csv	The result of the instruments endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. The CSV format is supported with only data_layout parameter value "by_factor".
start_date	string				The first date to be used for data retrieval. If only a start_date is given, only data for the specified date will be returned. Format - date (as full-date in RFC3339).
factor_name_list	Array				A list of ESG Factor names as defined in the ESG Factor Metadata. Factor names, but at least one factor name must be specified for a request to be valid.
data_layout	String				The API offers different data layouts to suit the different ways that a client may want to process the data that they receive. The Data Layout options provide a way for the client to tailor the output to their needs

Name	Type	Mandatory	Default value	Accepted values	Description								
					<table><tr><th>Data Layout Name</th><th>Description</th></tr><tr><td>by_factor</td><td>The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order</td></tr><tr><td>matrix_by_factor</td><td>This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.</td></tr><tr><td>change_ranges</td><td>This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.</td></tr></table>	Data Layout Name	Description	by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order	matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.	change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.
					Data Layout Name	Description							
					by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order							
					matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.							
change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.												
The API treats historical data as a continuous stream of values for the date range specified in the call. The user can choose how often they would like to sample the stream of data. Perhaps the client only cares about what the value is at the end of each month, for example. The following options are available for the Data Sample Frequency parameter:													
data_sample_frequency	String				<table><tr><th>Data Sample Name</th><th>Description</th></tr><tr><td>daily</td><td>A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.</td></tr><tr><td>business_month_end</td><td>A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values,</td></tr></table>	Data Sample Name	Description	daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.	business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values,		
					Data Sample Name	Description							
					daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.							
business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values,												

Name	Type	Mandatory	Default value	Accepted values	Description
					<div> <div></div> <div>one for each business month end day in the range</div> </div> <div> <div>calendar_month_end</div> <div>A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range</div> </div>
as_at_date	String				<p>A date which defines which corrections should be returned. If the as_at_date is before the end_date, the as_at_date will override the end_date. No values will be returned beyond the specified as_at_date. If no as_at_date is specified, the data will be viewed from the perspective of the date that the request was made. The Historical Data API returns data as published by MSCI ESG Research, including normal data updates and data corrections. If and when MSCI ESG Research corrects a data error, the Historical Data API returns the corrected value starting with the date the correction was made in our database. As a general rule, MSCI ESG Research does not restate historical data. The response using as_at_date will be the same as that of the response using as_of_date. Please see the MSCI ESG Research Data Correction Policy for more information: https://support.msci.com/sustainability_climate/data-correction-policy. Format - date (as full-date in RFC3339).</p>
end_date	String				<p>The end date for the range of values to be returned. If not specified, the request will return a single value corresponding to the given start_date. The end_date cannot exceed the as_at_date. If the end_date is after the as_at_date, the request will be rejected with a 400 status code. Format - date (as full-date in RFC3339).</p>
instrument_identifier_list	Array				<p>A list of one or more instrument identifiers which resolve to issuers. Identifiers can be ISINs, CUSIPs, MDS UIDS, etc.</p>
historical_product	String				<p>The Instrument historical product name for which historical data is to be retrieved. Only a single product name is accepted per request. All available products can be retrieved at /metadata/history/historicalProducts. Cannot be used along with factor_name_list or dataset_list. CSV Format is not supported with historical_product parameter.</p> <p>User is only allowed to fetch one day data when using 'daily - by_factor' and 'daily - matrix_by_factor' layout with 'historical_product' parameter.</p>

Name	Type	Mandatory	Default value	Accepted values	Description
dataset_list	Array				A list of Datasets as defined in /metadata/history/datasets. dataset_list Should be used along with instrument_identifier_list to retrieve historical data for specified identifiers and factors belonging to the datasets. CSV Format is not supported with dataset_list parameter.

RESPONSE

Possible Outputs

Status	Description
200	Success
206	Partial Content may have been returned
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not found
500	Internal Server Error
503	Service Unavailable

RESPONSE BODY

Name	Description
total_number_of_identifiers	Total number of identifiers returned.
total_number_of_unresolved_identifiers	Total number of identifiers that were not able to be resolved , An identifier is only unresolved if it did not exist at all during the specified time range.
total_number_of_batches	Total number of result batches.
current_batch	The index of the current batch.
data_request_id	The id representing this request. It can be used in future requests to retrieve subsequent batches of data.
data_request_id_expiration_time	This field specifies the expiration time of data_request_id.

Name	Description
unresolved_identifiers	A list of Unresolved Identifier objects.The Unresolved Identifier object is used to hold information about identifiers that were passed as parameters but were unable to be resolved for some reason and therefore data will not be returned.An identifier is only unresolved if it did not exist at all during the specified time range.
unresolved_factors	The Unresolved Factor object is used to hold information about factors that were passed as parameters but were not valid for some reason.
resolved_factors	A list of resolved Factor objects.
resolved_series_factors	A list of resolved series Factor objects.
identifiers_data_not_available	A list of requested identifiers for which all the requested factors data are not available.
factors_data_not_available	A list of factors for which data is not available for any of the requested identifiers.
messages[]	factor permissions will come under messages only in messages[] tag.

Output Encoding

application/xml, application/json, text/csv

Output Description

All responses from the API are returned in JSON format. All responses have a similar structure but at the core of any response is the data property.

Sample Output

Possible outputs for status: 200

by_factor layout:

Input

```
{
  "start_date": "2023-04-26",
  "end_date": "2023-04-27",
  "data_layout": "by_factor",
  "data_sample_frequency": "daily",
  "instrument_identifier_list": [
```

```

    "UIF000000027804394"

  ],
  "factor_name_list": [
    "GB_MSCI_VERIFIED",
    "GB_SELF_LABELLED"
  ]
}

```

Output

```

{
  "status": "OK",
  "code": 200,
  "trace_id": "67b42de6c83cf4acdf6b527e7657d846",
  "timestamp": "2025-02-18T06:51:18Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 1,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "0-d1428a46-0dbf-4024-aab2-5aced1d409ff-2025-02-19T06:51:18Z",
      "data_request_id_expiration_time": "2025-02-19T06:51:18+0000",
      "resolved_factors": {
        "BOND": [
          "GB_MSCI_VERIFIED",
          "GB_SELF_LABELLED"
        ]
      }
    },
    "data": [
      {
        "requested_id": "UIF000000027804394",
        "instrument_type": "BOND",
        "factors": [
          {
            "name": "GB_MSCI_VERIFIED",
            "data_values": [
              {
                "value": "Not Assessed",

```

```

        "as_of_date": "2023-04-26",
        "as_at_date": "2025-02-18T06:51:00Z"
      }
    ]
  },
  {
    "name": "GB_SELF_LABELLED",
    "data_values": [
      {
        "value": "True",
        "as_of_date": "2023-04-26",
        "as_at_date": "2025-02-18T06:51:00Z"
      }
    ]
  }
]
}
}

```

change_ranges layout:

Input

```

{
  "start_date": "2023-04-26",
  "end_date": "2023-04-27",
  "data_layout": "change_ranges",
  "instrument_identifier_list": [
    "UIF000000027804394"
  ],
  "factor_name_list": [
    "GB_MSCI_VERIFIED",
    "GB_SELF_LABELLED"
  ]
}

```

Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "67b430595ba9a636fb6267075ed937c9",
  "timestamp": "2025-02-18T07:01:45Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 1,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "1-40690457-7557-4a8c-9c0d-88a9a5894750-2025-02-19T07:01:45Z",
      "data_request_id_expiration_time": "2025-02-19T07:01:45+0000",
      "resolved_factors": {
        "BOND": [
          "GB_MSCI_VERIFIED",
          "GB_SELF_LABELLED"
        ]
      }
    },
    "data": [
      {
        "requested_id": "UIF000000027804394",
        "instrument_type": "BOND",
        "factors": [
          {
            "name": "GB_MSCI_VERIFIED",
            "data_values": [
              {
                "value": "Not Assessed",
                "as_of_date": "2023-04-26",
                "valid_until_date": "2023-04-27",
                "as_at_date": "2025-02-18T07:01:00Z"
              }
            ]
          },
          {
            "name": "GB_SELF_LABELLED",
            "data_values": [
```

```
{
  {
    "value": "True",
    "as_of_date": "2023-04-26",
    "valid_until_date": "2023-04-27",
    "as_at_date": "2025-02-18T07:01:00Z"
  }
}
```

matrix by factor layout:

Input

```
{
  "start_date": "2023-04-26",
  "end_date": "2023-04-27",
  "data_sample_frequency": "daily",
  "data_layout": "matrix_by_factor",
  "instrument_identifier_list": [
    "UIF000000027804394"
  ],
  "factor_name_list": [
    "GB_MSCI_VERIFIED",
    "GB_SELF_LABELLED"
  ]
}
```

Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "67b4309b5a2f65c6370e0da435ebb76a",
}
```

```
"timestamp": "2025-02-18T07:02:51Z",
"messages": [],
"result": {
  "response_metadata": {
    "total_number_of_identifiers": 1,
    "total_number_of_unresolved_identifiers": 0,
    "total_number_of_batches": 1,
    "current_batch": 1,
    "data_request_id": "2-83337583-f474-4726-8dea-4c9b1e6f322e-2025-02-19T07:02:51Z",
    "data_request_id_expiration_time": "2025-02-19T07:02:51+0000",
    "resolved_factors": {
      "BOND": [
        "GB_MSCI_VERIFIED",
        "GB_SELF_LABELLED"
      ]
    }
  },
  "data": {
    "requested_ids": [
      "UIF0000000027804394"
    ],
    "as_at_date": "2025-02-18T07:02:00Z",
    "as_of_dates": [
      "2023-04-26"
    ],
    "factors": [
      {
        "factor": "GB_MSCI_VERIFIED",
        "values": [
          [
            "Not Assessed"
          ]
        ]
      },
      {
        "factor": "GB_SELF_LABELLED",
        "values": [

```

```
[
  "True"
]
]
}
]
}
}
}
```

CODE SAMPLES

cURL

```
curl --request POST \
  --url https://api.msci.com/esg/data/v3.0/instruments/history \
  --header 'Authorization:Replace with your Auth token' \
  --header 'content-type: application/json' \
  --data '{"fund_identifier_list":["string"],"factor_name_list":["string"],"start_date":"2019-08-24","end_date":"2019-08-24","as_at_date":"2019-08-24","data_sample_frequency":"string","data_layout":"string","inherit_missing_values":true,"format":"string"}'
```

Python

```
import http.client
conn = http.client.HTTPSConnection("api.msci.com")
payload =
"{\"fund_identifier_list\": [\"string\"], \"factor_name_list\": [\"string\"], \"start_date\": \"2019-08-24\", \"end_date\": \"2019-08-24\", \"as_at_date\": \"2019-08-24\", \"data_sample_frequency\": \"string\", \"data_layout\": \"string\", \"inherit_missing_values\": true, \"format\": \"string\"}"
headers = {
    'Authorization': " Replace with your Auth token",
    'content-type': "application/json"
}
conn.request("POST", "/esg/data/v3.0/instruments/history", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse<String> response =
Unirest.post("https://api.msci.com/esg/data/v3.0/instruments/history")
    .header("Authorization", " Replace with your Auth token ")
    .header("content-type", "application/json")
    .body("{\"fund_identifier_list\": [\"string\"], \"factor_name_list\": [\"string\"], \"start_date\": \"2019-08-24\", \"end_date\": \"2019-08-24\", \"as_at_date\": \"2019-08-24\", \"data_sample_frequency\": \"string\", \"data_layout\": \"string\", \"inherit_missing_values\": true, \"format\": \"string\"}")
    .asString();
```

C#

```
var client = new RestClient("https://api.msci.com/esg/data/v3.0/instruments/history");
var request = new RestRequest(Method.POST);
request.AddHeader("Authorization", " Replace with your Auth token ");
```



```
request.AddHeader("content-type", "application/json");
request.AddParameter("application/json",
    "{ \"fund_identifier_list\": [\"string\"], \"factor_name_list\": [\"string\"], \"start_date\": \"2019-08-24\", \"end_date\": \"2019-08-24\", \"as_at_date\": \"2019-08-24\", \"data_sample_frequency\": \"string\", \"data_layout\": \"string\", \"inherit_missing_values\": true, \"format\": \"string\" }", ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
```

/ISSUERS/

HTTP Method:

GET

ENDPOINT DESCRIPTION

This endpoint is used to retrieve factor data based on the parameters given in the request. The results are governed by the data points and issuer/ fund coverage permissioned to the account. This request allows the caller to specify which data they want to retrieve, and from which 'universe' of companies. For example, a request could ask for all ESG Ratings category data for issuers in the "Banks" ESG Rating Industry (15698). The result would be a JSON object with a list of issuers along with the requested data. The query would look like this:

https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG Ratings:Company Summary&esg_industry_id_list=15698&coverage=esg_ratings

The inclusion of the coverage ensures that only issuers who have ESG related information are returned in the result.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/issuers>

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
format	String		json	json, csv, xml	The result of the issuers endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. XML can also be specified and the result will be a properly formatted XML document without a DTD or schema.

Name	Type	Mandatory	Default value	Accepted values	Description
offset	integer			Must be greater than zero	The Data API has the potential to generate large amounts of data. If a query will result in extra large amounts of data, the user will be required to specify paging parameters. The offset value indicates which record to start retrieving values. This option is usually paired with the limit parameter to create a window of results to return. When paging is using, an additional section of the JSON response will contain information about the total number of records that could be returned, along with links for the next,prev,first,last pages where appropriate. If you are on the last page, no next link will be generated, for example.
limit	integer			Must be greater than 1	The Data API has the potential to generate large amounts of data. If a query will result in extra large amounts of data, the user will be required to specify paging parameters. The limit value indicates the maximum number of records to return. This option is usually paired with the offset parameter to create a window of results to return. When paging is using, an additional section of the JSON response will contain information about the total number of records that could be returned, along with links for the next,prev,first,last pages where appropriate. If you are on the last page, no next link will be generated, for example.
reference_column_list	Array			issuerid, issuername, level	If a parent/child option is specified, other than do_not_apply, the client can also specify one or more 'reference' columns to appear along with any inherited values. These values help to identify the source of the inheritance. The issuerid column will contain the MSCI issuer id of the issuer from which the value is derived. Similarly, the issuername column will contain the name of the issuer. Finally, the level is a numeric value indicating how many levels away from the issuer the value was inherited from.
[array item]	string			issuerid, issuer_id, issuername, issuer_name, level	
parent_child	string		do_not_apply	do_not_apply, inherit_missing_values, include_subsidiaries, full_parent_child	This option is used to control how data is inherited from parent entities, and or whether to include subsidiaries of an issuer. A client must have purchased the Subsidiary Mapping

Name	Type	Mandatory	Default value	Accepted values	Description
					feature for this option to have any effect.
					The 'inherit_missing_values' will include data values from parent issuers if the issuer does not have a value of it's own.
					The 'include_subsidiaries' will include subsidiaries of an issuer.
					The 'full_parent_child' option will both inherit missing data values from the parent as well as include subsidiaries of an issuer.
					The default is 'do_not_apply'.
product_name_list	array				This parameter is a list of one or more product names which contain groups of data factors. Data factors are grouped by category as well as by product name. A product name is used to identify a product to which a data factor must belong in order to be returned. The list of products available to the caller can be found via the /parameterValues/factorProductNames endpoint
[array item]	string				Data factors are grouped by category as well as by product name. A product name is used to identify a product to which a data factor must belong in order to be returned. The list of products available to the caller can be found via the /parameterValues/factorProductNames endpoint.
category_path_list	array				This parameter is a list of one or more category path strings. A category path is used to identify a collection of data factors. All category paths available to the caller can be obtained from the /parameterValues/factorCategoryPaths endpoint.
[array item]	string				A category path is used to identify a collection of data factors. All category paths available to the caller can be obtained from the /parameterValues/factorCategoryPaths endpoint.

Name	Type	Mandatory	Default value	Accepted values	Description
factor_name_list	array				This is a list of factor names. A factor name is used to identify a unique data point value associated with an issuer. A full list of available factors for the caller can be found by using the /metadata/factors endpoint.
[array item]	string				A factor name is used to identify a unique data point value associated with an issuer. A full list of available factors for the caller can be found by using the /metadata/factors endpoint.
coverage	string				A coverage is an identifier used to limit the list of issuers in the result to those that belong to the specified coverage universe. For example, the esg_ratings coverage would only include issuers that have an ESG Rating.
name_contains	string				The name_matches string is used to locate issuers whose primary issuer name value contains the given string anywhere in the name. If parent/child options are used, additional issuers may be returned that are related to the primary issuer, but will not necessarily match the pattern given.
starts_with	string				Limit the primary issuer to a name that starts with the specified value. If parent/child options are specified, there may be issuers included that don't start with the specified name.
gics_subindustry_id_list	array				This parameter is a list of one or more gics subindustry codes. The issuers returned will be limited to those belonging to the specified GICS Sub Industries.
[array item]	string				A GICS SubIndustry code is a string value which is used to identify a particular GICS SubIndustry. A full list of GICS SubIndustries is available from the /parameterValues/gicsSubIndustries endpoint.
esg_industry_id_list	array				A GICS SubIndustry code is a string value which is used to identify a particular GICS SubIndustry. A full list of GICS SubIndustries is available from the /parameterValues/gicsSubIndustries endpoint.
					This parameter is a list of one or more ESG Industry codes. The issuers returned will be limited to those

Name	Type	Mandatory	Default value	Accepted values	Description
					<p>belonging to the specified ESG Industries.</p> <p>The ESG Industry ID is a string value that identifies an ESG Industry. A full list of available ESG Industries can be retrieved from the <code>/parameterValues/esgIndustries</code> endpoint</p> <p>The ESG Industry ID is a string value that identifies an ESG Industry. A full list of available ESG Industries can be retrieved from the <code>/parameterValues/esgIndustries</code> endpoint</p>
[array item]	string				
country_code_list	array				<p>This parameter contains a list of one or more country codes which is associated with the issuer. A Country Code is the 2 character code representing a country.</p>
[array item]	string				<p>A Country Code is the 2 character code representing a country.</p>
index_identifier_list	array				<p>A collection of index identifiers compatible with MSCI ESG Manager. An index identifier is a string value which identifies an index to use for a query. Indexes are used to limit the results of a request to a specific set of issuers that belong to the specified index. A full list of indexes that are available to the caller can be retrieved at <code>/parameterValues/indexes</code></p>
[array item]	string				<p>The index id is a string value which identifies an index to use for a query. Indexes are used to limit the results of a request to a specific set of issuers that belong to the specified index. A full list of indexes that are available to the caller can be retrieved at <code>/parameterValues/indexes</code></p>
index_identifier_type	string				<p>When requesting data, the client can specify a list of one or more issuer identifiers for which data will be returned. The default issuer identifier is an MSCI issuer id. However, by specifying a different lookup factor, the client can specify IDs such as ISINs. All identifiers must be of the same type, however. For example, mixing ISINs and MSCI Issuer Ids is not allowed.</p>
issuer_identifier_list	array				<p>This parameter is used to limit the results to a specific set of issuers. The caller can list one or more issuer</p>

Name	Type	Mandatory	Default value	Accepted values	Description
[array item]	string				<p>identifiers which identified the issuers for which data should be returned.</p> <p>The issuer identifier is a value that is used to locate an issuer within the universe of issuers available to the client. By default, this identifier is an MSCI Issuer ID value, but this can also be a different identifier type when the issuer_identifier_type is specified. All identifiers must be of the same type.</p> <p>The issuer id is an identifier used to locate an issuer within the universe of issuers available to the client. By default, this id is an MSCI Issuer ID value, but this can also be a different identifier type when the lookup_factor is specified. All ids must be of the same type.</p>

RESPONSE

Possible Outputs

Status	Description
200	Success
206	Partial Content may have been returned
400	Malformed request
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/xml, application/json, text/csv

Output Description

The issuers endpoint returns a collection of data for the issuers and datapoints requested. By default, the result is returned in JSON format. The results can also be returned as a CSV file if the request uses the format parameter, or if the accept header specifies text/csv as an acceptable result.

Response Body Returned

Possible outputs for status: 200

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "9fc435f08e70b687",
  "timestamp": "2018-11-26T13:48:26Z",
  "messages": [],
  "result": {
    "issuers": [
      {
        "ISSUER_NAME": "CSX Corporation",
        "ISSUERID": "IID000000002123685",
        "ISSUER_TICKER": "CSX",
        "ISSUER_CUSIP": 126408103,
        "ISSUER_SEDOL": 2160753,
        "ISSUER_ISIN": "US1264081035",
        "ISSUER_CNTRY_DOMICILE": "US",
        "IVA_COMPANY_RATING": "A",
        "IVA_RATING_ANALYSIS": "CSX has been upgraded to 'A' from 'BBB'. Recent enhancements in our methodology have seen an improvement in CSX's corporate governance assessment, particularly in its board and pay themes\n\nCSX has a split CEO/chairman role and management oversight is further strengthened by a majority independent board and fully independent audit committee. The company has several programs to mitigate safety risks, but witnessed two employee fatalities in 2017 (2016: 0) and increased frequency of employee injuries. Despite investing USD 2 billion by end-2017, CSX looks unlikely to meet a 2018 deadline for the US-mandated Positive Train Control system. Large-scale layoffs continue with 6,200 jobs expected to be cut from 2018-2020 (26% of the workforce). While CSX employees have competitive benefits and 83% are represented by unions, negative morale and productivity impacts might be unavoidable.",
        "IVA_RATING_DATE": 20180405,
        "IVA_INDUSTY": "Road & Rail Transport",
        "IVA_PREVIOUS_RATING": "BBB",
        "IVA_RATING_TREND": 1,
        "ENVIRONMENTAL_PILLAR_QUARTILE": 3,
        "ENVIRONMENTAL_PILLAR_SCORE": 4.9,
        "ENVIRONMENTAL_PILLAR_WEIGHT": 22,
        "INDUSTRY_ADJUSTED_SCORE": 6.4,
        "GICS_SUB_IND": "Railroads",
        "GOVERNANCE_PILLAR_SCORE": 7.8,
        "GOVERNANCE_PILLAR_SCORE_2014": 10,
        "GOVERNANCE_PILLAR_QUARTILE": 1,
        "GOVERNANCE_PILLAR_WEIGHT": 22,
        "ESG_OVERALL_QUARTILE": 2,
        "SOCIAL_PILLAR_SCORE": 3.9,
        "SOCIAL_PILLAR_QUARTILE": 3,
        "SOCIAL_PILLAR_WEIGHT": 56,
        "TOTAL_NUMBER_CONTROVERSIES": 33,
        "WEIGHTED_AVERAGE_SCORE": 5
      }
    ]
  }
}
```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/json"
```

-X GET

```
https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701
```

Python

```
import requests url =
"https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701"; headers = {
"Accept": "application/json" } response = requests.request("POST", url, headers=headers)
print(response.text)
```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701")
.header("Accept", "application/json") .asString();
```

C#

```
var client = new
RestClient("https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701"); var
request = new RestRequest(Method.GET); request.AddHeader("Accept", "application/json");
IRestResponse response = client.Execute(request);
```

HTTP Method:

POST

ENDPOINT DESCRIPTION

In addition to requesting issuer data via a GET request, users can also create a POST request where the POSTed data consists of the various query parameters that are available on the GET request. The POST form of this endpoint allows for potentially large amounts of data to be specified. For example, a client may have a list of issuers IDs that they wish to retrieve data for. Specifying the ids in a GET request may exceed the maximum length allowed for a GET request. A POST request has no such limit.

It is important to note that any POST request specify the Content-Type being used for submission. For example, if JSON is being used to send data, the Content-Type should be application/json, otherwise the request may be rejected with a 400 status code and an 'Invalid formatting' error message.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/issuers>

Body payload

Name	Type	Mandatory	Default value	Accepted values	Description
issuer_identifier_list	array				
[array item]	string				
issuer_identifier_type	string				The issuer identifier type is a string defining the factor name used for issuers in the issuer_identifer_list parameter. Available options

Name	Type	Mandatory	Default value	Accepted values	Description
					vary by user, but include ISSUER_ID, ISIN, CIK, CUSIP, LEI, SEDOL
index_identifier_list	array				
[array item]	string				
country_code_list	array				
[array item]	string				
esg_industry_id_list	array				
[array item]	string				
gics_subindustry_id_list	array				
[array item]	string				
starts_with	string				Locate issuers whose name starts with the given value and return requested data related to those issuers. If parent/child is specified, the related issuers will not necessarily have names that start with the given string. The value is only used to identify the primary issuers that are the result of the query.
name_contains	string				If a name_contains value is specified, only issuers that have the specified string will be included in the results.
coverage	string				A coverage is used to restrict the results to issuers which have certain properties relating to data coverage. For example, the AGR coverage would restrict issuers to those which have an AGR rating. The list of coverages that are available for a user is found at the /parameterValues/coverages endpoint.
factor_name_list	array				
[array item]	string				
category_path_list	array				
[array item]	string				
product_name_list	array				
[array item]	string				
parent_child	string			do_not_apply, inherit_missing_values,	The parent_child option is used to control whether other issuers or values are to be

Name	Type	Mandatory	Default value	Accepted values	Description
reference_column_list	array			include_subsidiaries, full_parent_child	inherited or displayed based on the subsidiary structure of an issuer. The availability of this feature is based on client permissions. The default action is do_not_apply.
[array item]	string			issuerid, issuer_id, issuername, issuer_name, level	The specific reference column to include for each inherited data factor
limit	integer			Must be greater than 1	The limit parameter is used to define the maximum number of results that will be returned in the request. This parameter is option, however some data requests will result in a large number of issuers being returned. When this is detected by the API, an error will be returned indicating that a limit must be specified.
offset	integer			Must be greater than zero	The offset value is paired with a limit parameter to define a window of results. When a limit is specified without an offset, the offset is treated as 0. This means that the results will contain the first record, up to the limit of items requested. To get the next 'page', a new offset must be specified.
format	string		JSON	JSON, CSV, XML	When paging is in effect, and JSON is being returned, the results will include predefined links which can be used to get the next, previous, first, and last pages based on the limit and offset values given in the request. An optional format parameter can be specified to control the format of the output. Allowable values are: JSON , CSV, XML

RESPONSE

Possible Outputs

Status	Description
200	OK
206	Partial content may have been returned
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not found
500	Internal Server Error

Output Encoding

application/xml, application/json, text/csv

Output Description

The issuers endpoint returns a collection of data for the issuers and datapoints requested. By default, the result is returned in JSON format. The results can also be returned as a CSV file if the request uses the format parameter, or if the accept header specifies text/csv as an acceptable result.

Response Body Returned

Possible outputs for status: 20

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "9fc435f08e70b687",
  "timestamp": "2018-11-26T13:48:26Z",
  "messages": [],
  "result": {
    "issuers": [
      {
        "ISSUER_NAME": "CSX Corporation",
        "ISSUERID": "IID000000002123685",
        "ISSUER_TICKER": "CSX",
        "ISSUER_CUSIP": 126408103,
        "ISSUER_SEDOL": 2160753,
        "ISSUER_ISIN": "US1264081035",
        "ISSUER_CNTRY_DOMICILE": "US",
        "IVA_COMPANY_RATING": "A",
        "IVA_RATING_ANALYSIS": "CSX has been upgraded to 'A' from 'BBB'. Recent enhancements in our methodology have seen an improvement in CSX's corporate governance assessment, particularly in its board and pay themes\n\nCSX has a split CEO/chairman role and management oversight is further strengthened by a majority independent board and fully independent audit committee. The company has several programs to mitigate safety risks, but witnessed two employee fatalities in 2017 (2016: 0) and increased frequency of employee injuries. Despite investing USD 2 billion by end-2017, CSX looks unlikely to meet a 2018 deadline for the US-mandated Positive Train Control system. Large-scale layoffs continue with 6,200 jobs expected to be cut from 2018-2020 (26% of the workforce). While CSX employees have competitive benefits and 83% are represented by unions, negative morale and productivity impacts might be unavoidable.",
        "IVA_RATING_DATE": 20180405,
        "IVA_INDUSTRY": "Road & Rail Transport",
        "IVA_PREVIOUS_RATING": "BBB",
        "IVA_RATING_TREND": 1,
        "ENVIRONMENTAL_PILLAR_QUARTILE": 3,
        "ENVIRONMENTAL_PILLAR_SCORE": 4.9,
        "ENVIRONMENTAL_PILLAR_WEIGHT": 22,
        "INDUSTRY_ADJUSTED_SCORE": 6.4,
        "GICS_SUB_IND": "Railroads",
        "GOVERNANCE_PILLAR_SCORE": 7.8,
        "GOVERNANCE_PILLAR_SCORE_2014": 10,
        "GOVERNANCE_PILLAR_QUARTILE": 1,
        "GOVERNANCE_PILLAR_WEIGHT": 22,
        "ESG_OVERALL_QUARTILE": 2,
        "SOCIAL_PILLAR_SCORE": 3.9,
        "SOCIAL_PILLAR_QUARTILE": 3,
        "SOCIAL_PILLAR_WEIGHT": 56,
        "TOTAL_NUMBER_CONTROVERSIES": 33,

```

```

    "WEIGHTED_AVERAGE_SCORE": 5
  }
}
}

```

CODE SAMPLES

cURL

```

curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701

```

Python

```

import requests url =
"https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701"; headers = {
"Accept": "application/json" } response = requests.request("POST", url, headers=headers)
print(response.text)

```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```

HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701")
.header("Accept", "application/json") .asString();

```

C#

```

var client = new
RestClient("https://api.msci.com/esg/data/v3.0/issuers?category_path_list=ESG+Ratings:Company+Summary&coverage=esg_ratings&format=json&index_identifier_list=UNX000000012913701"); var
request = new RestRequest(Method.GET); request.AddHeader("Accept", "application/json");
IRestResponse response = client.Execute(request);

```

/ISSUERS/HISTORY

The /issuers/history endpoint provides a mechanism to retrieve ESG data history for a given set of issuer identifiers. A user of the API can send in a list of issuer identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

When making a POST request, a proper Content-Type must be specified. Since the API utilizes JSON for POST requests, the Content-Type should be application/json. If the Content-Type header is not properly specified, the request may be rejected with a 400 status code and an 'Invalid formatting' error message.

Depending on the amount of data being returned, multiple requests may be required. An initial request will return the first batch of data, along with information about whether more batches are available. If more batches are available, the user should choose one of the additional data batch request services documented below.

When requesting ESG history data, it is possible to create a request which returns a large amount of data. When a request is made for history data, the API will determine the size of the total result. Based on the calculated size, the

API will determine the total number of batches necessary to return all of the requested data. Every response returned from the history API endpoint contains a `response_metadata` property. A typical response will look something like this:

```
"response_metadata": {
  "total_number_of_identifiers": 10,
  "total_number_of_batches": 3,
  "current_batch": 1,
  "data_request_id": "bb005730-262c-4d1c-88bb-acc916ee0ed7"
}
```

In this example, a request was made for history data. The response is indicating that there will be 10 instruments in the response, broken into 3 batches. The `current_batch` property indicates that this is the first batch in the result.

To retrieve additional batches, additional calls are made to `/issuers/history`, passing the following parameters instead of all the parameters sent in the first request:

Property Name	Description
<code>data_request_id</code>	This is an identifier that is assigned to a data request. It acts as a key to retrieve additional batches of information. The key remains valid for as long as data is being requested, but will expire at some point and cannot be used for caching purposes. While calling get endpoint use encoding as the <code>request_id</code> has special characters (code sample below), Note: The POST method should be the method used by most API applications
<code>batch_id</code>	This is a number used to identify which batch of data to retrieve.

Note that this method of retrieving data does not involve “paging” through the data. Once an initial request is made, the client can make multiple parallel calls to retrieve different batches of data. Also, each request which utilizes the `data_request_id` will be idempotent.

The `data_request_id` is a unique identifier which can be re-used for some period of time to refer to the same request. After the first request, the client will know how many batches exist, and if desired can generate parallel requests to get subsequent batches of data. Each batch will contain a complete set of data for a requested identifier. There are no order dependencies in the response. After some set period of time, currently set to 24 hours after the first request, the `data_request_id` will expire and requests for batches using the given id will no longer return a response.

Below code sample for encoding in Java/Python

Java:

```
import java.net.URLEncoder;
url =
"https://api.msci.com/esg/data/v3.0/issuers/history?batch_id=2&data_request_id="+URLEncoder.encode(data_request_id, StandardCharsets.UTF_8.toString());
```

Python:

```
from urllib.parse import urlencode
urlencode(dict(data_request_id=data_request_id, batch_id=batch_id))
```

• RETRIEVE ESG DATA HISTORY FOR REQUESTED IDENTIFIERS

The /issuers/history endpoint provides a mechanism to retrieve ESG data history for a given set of issuer identifiers. A user of the API can send in a list of issuer identifiers, a range of dates, and a collection of data factors to retrieve, and the API will return the values for each of the requested factors for the given time range. There are several options available to control how much data is returned.

The endpoint will support requests using both GET and POST. The POST method should be the method used by most API applications since it can support a larger request size (identifiers, factor names, etc.). The GET method would typically be used for quick experimental queries.

The API accepts two types of input parameters to the request, one form of input initiates a new retrieval request, the other is used to return subsequent batches of data. The /issuers/history endpoint uses a dynamic batching scheme to return data if the results are too large to return in one request (see the section on data batching for more details on how large results are returned)

The basic request for data will include the following information:

- A list of one or more identifiers. These identifiers will be used to locate companies (issuers) who have ESG Data. The identifiers can represent Bonds, Securities, or the issuers themselves.
- A list of data factors. The factors can be the names of the factors themselves. Factor information is available through an API endpoint such as /metadata/history/factors.
- A date range. Allowed date range must be within a 1 year. The API supports date ranges of 1 date to many dates. All dates are in the format yyyy-mm-dd, i.e. 2020-04-01 would be April 1, 2020.
- Other parameters to fine tune the request

HTTP Method:

GET

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/issuers/history>

Query parameters

Name	Type	Description
issuer_identifier_list	array	A list of one or more issuer/instrument identifiers which resolve to issuers. Identifiers can be MSCI Issuer IDs,

Name	Type	Description
		ISINs, CUSIPs, MDS UUIDs, etc
[array item]	string	
factor_name_list	array	A list of ESG Factor names as defined in the ESG Factor Metadata. Factor names, product names, and category paths can be mixed together, but at least one factor name, category path, or product name must be specified for a request to be valid.
[array item]	string	
start_date	date	The first date to be used for data retrieval. If only a start_date is given, only data for the specified date will be returned. The format of the date is yyyy-mm-dd. For example, 2020-04-01 would be April 1, 2020.
end_date	date	The end date for the range of values to be returned. If not specified, the request will return a single value corresponding to the given start_date. The end_date cannot exceed the as_at_date. If the end_date is after the as_at_date, the request will be rejected with a 400 status code. The format of the date is yyyy-mm-dd. For example, 2020-04-01 would be April 1, 2020.
as_at_date	date	A date which defines which corrections should be returned. If the as_at_date is before the end_date, the as_at_date will override the end_date. No values will be returned beyond the specified as_at_date. If no as_at_date is specified, the data will be viewed from the perspective of the date that the request was made. The Historical Data API returns data as published by MSCI ESG Research, including normal data updates and data corrections. If and when MSCI ESG Research corrects a data error, the Historical Data API returns the corrected value starting with the date the correction was made in our database. As a general rule, MSCI ESG Research does not restate historical data. The response using as_at_date will be the same as that of the response using as_of_date. Please see the MSCI ESG Research Data Correction Policy for more information: https://support.msci.com/sustainability_climate/data-correction-policy . Format - date (as full-date in RFC3339). The format of the date is yyyy-mm-dd. For example, 2020-04-01 would be April 1, 2020.

Name	Type	Description								
data_sample_frequency	string	<p>The API treats historical data as a continuous stream of values for the date range specified in the call. The user can choose how often they would like to sample the stream of data. Perhaps the client only cares about what the value is at the end of each month, for example. The following options are available for the Data Sample Frequency parameter:</p>								
		<table><tr><th>Data Sample Name</th><th>Description</th></tr><tr><td>Daily</td><td>A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.</td></tr><tr><td>business_month_end</td><td>A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range</td></tr><tr><td>calendar_month_end</td><td>A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range</td></tr></table>	Data Sample Name	Description	Daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.	business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range	calendar_month_end	A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range
		Data Sample Name	Description							
		Daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the matrix_by_factor layout.							
business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range									
calendar_month_end	A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range									
data_layout	string	<p>The API offers different data layouts to suit the different ways that a client may want to process the data that they receive. The Data Layout options provide a way for the client to tailor the output to their needs</p>								
		<table><tr><th>Data Layout Name</th><th>Description</th></tr><tr><td>by_factor</td><td>The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order</td></tr><tr><td>matrix_by_factor</td><td>This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.</td></tr><tr><td>change_ranges</td><td>This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.</td></tr></table>	Data Layout Name	Description	by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order	matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.	change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.
		Data Layout Name	Description							
		by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order							
matrix_by_factor	This layout is more compressed than the standard by_factor layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and business_month_end sample frequencies.									
change_ranges	This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.									

If this value is not specified, the **by_factor** layout will be used.

Name	Type	Description
inherit_missing_values	boolean	A Boolean value used to indicate whether the API should inherit missing values from a parent issuer. This option will automatically be set to true if the caller is permitted to receive inherited data
format	string	The result of the issuers endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. The CSV format is supported with only data_layout parameter value "by_factor".
historical_product	String	<p>The historical product name for which historical data is to be retrieved. Only a single product name is accepted per request. All available products can be retrieved at /metadata/history/historicalProducts. Cannot be used along with factor_name_list or dataset_list. CSV Format is not supported with historical_product parameter.</p> <p>User is only allowed to fetch one day data when using 'daily - by_factor' and 'daily - matrix_by_factor' layout with 'historical_product' parameter.</p>
dataset_list	Array	A list of Datasets as defined in /metadata/history/datasets. dataset_list Should be used along with issuer_identifier_list to retrieve historical data for specified identifiers and factors belonging to the datasets. CSV Format is not supported with dataset_list parameter.

RESPONSE

Possible Outputs

Status	Description
200	Success
206	Partial Content may have been returned
400	Malformed request
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

RESPONSE BODY

Name	Description
total_number_of_identifiers	Total number of identifiers returned.
total_number_of_unresolved_identifiers	Total number of identifiers that were not able to be resolved , An identifier is only unresolved if it did not exist at all during the specified time range.
total_number_of_batches	Total number of result batches.
current_batch	The index of the current batch.
data_request_id	The id representing this request. It can be used in future requests to retrieve subsequent batches of data.
data_request_id_expiration_time	This field specifies the expiration time of data_request_id.
unresolved_identifiers	A list of Unresolved Identifier objects.The Unresolved Identifier object is used to hold information about identifiers that were passed as parameters but were unable to be resolved for some reason and therefore data will not be returned.An identifier is only unresolved if it did not exist at all during the specified time range.
unresolved_factors	The Unresolved Factor object is used to hold information about factors that were passed as parameters but were not valid for some reason.
resolved_factors	A list of resolved Factor objects.
resolved_series_factors	A list of resolved series Factor objects.
identifiers_data_not_available	A list of requested identifiers for which all the requested factors data are not available.

Name	Description
factors_data_not_available	A list of factors for which data is not available for any of the requested identifiers.
messages[]	factor permissions will come under messages only in messages[] tag.

Output Encoding

application/xml, application/json, text/csv

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fc2f67ca74d9a28d98ffd778a803b",
  "timestamp": "2024-04-29T15:55:34Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 2,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "0-9b9c4d83-2af6-487c-8c17-fff9b7c00213-2024-04-30T15:55:34+0000",
      "data_request_id_expiration_time": "2024-04-30T15:55:34+0000",
      "resolved_factors": [
        "CONTR_COLLECTIVE_BARG_N_SEVERE",
        "CONTR_HEALTH_SAFETY_N_SEVERE"
      ],
      "resolved_series_factors": [
        "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
      ],
      "identifiers_data_not_available": [
        "IID000000002286384"
      ],
      "factors_data_not_available": [
        "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
      ]
    },
    "data": [
      {
        "requested_id": "XS2709528959",
        "issuer_metadata": {
          "ISSUERID": "IID000000002236325",
          "ISSUER_NAME": "COMMERZBANK
          AKTIENGESELLSCHAFT",
          "CIK_NUM": "0000839463",
          "ISSUER_ISIN": "XS2709528959",
```

```

        "as_of_date": "2023-10-25",
        "valid_until_date": "2023-10-30"
    },
    ],
    "factors": [
        {
            "name":
"CONTR_COLLECTIVE_BARG_N_SEVERE",
            "data_values": [
                {
                    "value": 0,
                    "as_of_date": "2023-10-25",
                    "as_at_date": "2024-04-
29T15:55:00Z",
                    "reference_issuer_id":
"IID000000002135074",
                    "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
                },
                {
                    "value": 0,
                    "as_of_date": "2023-10-26",
                    "as_at_date": "2024-04-
29T15:55:00Z",
                    "reference_issuer_id":
"IID000000002135074",
                    "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
                },
                {
                    "value": 0,
                    "as_of_date": "2023-10-27",
                    "as_at_date": "2024-04-
29T15:55:00Z",
                    "reference_issuer_id":
"IID000000002135074",
                    "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
                },
                {
                    "value": 0,
                    "as_of_date": "2023-10-28",
                    "as_at_date": "2024-04-
29T15:55:00Z",
                    "reference_issuer_id":
"IID000000002135074",
                    "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
                },
                {
                    "value": 0,
                    "as_of_date": "2023-10-29",
                    "as_at_date": "2024-04-
29T15:55:00Z",
                    "reference_issuer_id":
"IID000000002135074",
                    "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
                }
            ]
        }
    ]

```

```

    },
    {
      "name":
"CONTR_HEALTH_SAFETY_N_SEVERE",
      "data_values": [
        {
          "value": 0,
          "as_of_date": "2023-10-25",
          "as_at_date": "2024-04-
29T15:55:00Z",
          "reference_issuer_id":
"IID000000002135074",
          "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-10-26",
          "as_at_date": "2024-04-
29T15:55:00Z",
          "reference_issuer_id":
"IID000000002135074",
          "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-10-27",
          "as_at_date": "2024-04-
29T15:55:00Z",
          "reference_issuer_id":
"IID000000002135074",
          "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-10-28",
          "as_at_date": "2024-04-
29T15:55:00Z",
          "reference_issuer_id":
"IID000000002135074",
          "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-10-29",
          "as_at_date": "2024-04-
29T15:55:00Z",
          "reference_issuer_id":
"IID000000002135074",
          "reference_issuer_name":
"COMMERZBANK Aktiengesellschaft."
        }
      ]
    }
  ]
}
]

```

```
}
}
```

HTTP Method:

POST

REQUEST

Service URLs:

https://api.msci.com/esg/data/v3.0/issuers/history

Body payload

Name	Type	Description
IssuerHistoryRequest		
issuer_identifier_list	array	A list of one or more issuer/instrument identifiers which resolve to issuers. Identifiers can be MSCI Issuer IDs, ISINs, CUSIPs, MDS UIDS, etc
[array item]	string	
factor_name_list	array	A list of ESG Factor names as defined in the ESG Factor Metadata. Factor names, product names, and category paths can be mixed together, but at least one factor name, category path, or product name must be specified for a request to be valid.
[array item]	string	
start_date	date	The first date to be used for data retrieval. If only a start_date is given, only data for the specified date will be returned. The format of the date is yyyy-mm-dd. For example, 2020-04-01 would be April 1, 2020.
end_date	date	The end date for the range of values to be returned. If not specified, the request will return a single value corresponding to the given start_date. The end_date cannot exceed the as_at_date. If the end_date is after the as_at_date, the request will be rejected with a 400 status code. The format of the date is yyyy-mm-dd. For example, 2020-04-01 would be April 1, 2020.
as_at_date	date	A date which defines which corrections should be returned. If the as_at_date is before the end_date, the as_at_date will override the end_date. No values will be returned beyond the specified as_at_date. If no as_at_date is specified, the data will be viewed from the perspective

Name	Type	Description														
		<p>of the date that the request was made. The Historical Data API returns data as published by MSCI ESG Research, including normal data updates and data corrections. If and when MSCI ESG Research corrects a data error, the Historical Data API returns the corrected value starting with the date the correction was made in our database. As a general rule, MSCI ESG Research does not restate historical data. The response using <code>as_at_date</code> will be the same as that of the response using <code>as_of_date</code>. Please see the MSCI ESG Research Data Correction Policy for more information: https://support.msci.com/sustainability_climate/data-correction-policy. Format - date (as full-date in RFC3339). The format of the date is yyyy-mm-dd. For example, 2020-04-01 would be April 1, 2020.</p> <p>The API treats historical data as a continuous stream of values for the date range specified in the call. The user can choose how often they would like to sample the stream of data. Perhaps the client only cares about what the value is at the end of each month, for example. The following options are available for the Data Sample Frequency parameter:</p> <table><thead><tr><th>Data Sample Name</th><th>Description</th></tr></thead><tbody><tr><td>daily</td><td>A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the <code>matrix_by_factor</code> layout.</td></tr><tr><td>business_month_end</td><td>A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range</td></tr><tr><td>calendar_month_end</td><td>A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range</td></tr></tbody></table> <p>If a data sample frequency is not specified, the value defaults to <code>business_month_end</code>. When specifying a date range, keep in mind that the range must include at least one business month end date if this option is in effect.</p> <p>The API offers different data layouts to suit the different ways that a client may want to process the data that they receive. The Data Layout options provide a way for the client to tailor the output to their needs</p> <table><thead><tr><th>Data Layout Name</th><th>Description</th></tr></thead><tbody><tr><td>by_factor</td><td>The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order</td></tr><tr><td>matrix_by_factor</td><td>This layout is more compressed than the standard <code>by_factor</code> layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and <code>business_month_end</code> sample frequencies.</td></tr></tbody></table>	Data Sample Name	Description	daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the <code>matrix_by_factor</code> layout.	business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range	calendar_month_end	A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range	Data Layout Name	Description	by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order	matrix_by_factor	This layout is more compressed than the standard <code>by_factor</code> layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and <code>business_month_end</code> sample frequencies.
Data Sample Name	Description															
daily	A data value sample will be taken from each date in the specified range. If there are 200 days in the given range, there will be 200 samples taken from the stream of values, one for each day in the range. The daily range is currently only available when using the <code>matrix_by_factor</code> layout.															
business_month_end	A data value sample will be taken from each business month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each business month end day in the range															
calendar_month_end	A data value sample will be taken from each calendar month end date in the specified range. If there are 12 months in the given range, there will be 12 samples taken from the stream of values, one for each calendar month end day in the range															
Data Layout Name	Description															
by_factor	The data results will be grouped by factor, and within each factor the values will be display in reverse chronological order															
matrix_by_factor	This layout is more compressed than the standard <code>by_factor</code> layout and is designed to support usage by quants who will want data rows with no gaps. The exact layout of the data content is still being finalized, but it will represent the data as a 2-d matrix. This layout is compatible with both daily and <code>business_month_end</code> sample frequencies.															
data_sample_frequency	string															
data_layout	string															

Name	Type	Description
		<div> <div>change_ranges</div> <div>This layout will only show change ranges for each value returned. Ranges will start at the first date in the requested range and end on the last date. If a value for a factor was value for 3 months in the range, and then changed and was valid for the remaining months, there would be two value ranges in the result for that particular value.</div> </div> <p>If this value is not specified, the by_factor layout will be used.</p>
inherit_missing_values	boolean	<p>A Boolean value used to indicate whether the API should inherit missing values from a parent issuer. This option will automatically be set to true if the caller is permitted to receive inherited data.</p> <p>This option will automatically be set to true if the caller is permitted to receive inherited data.</p>
format	String	<p>The result of the issuers endpoint will be returned in JSON format by default. However, the result can also be returned as a CSV file, which can then be imported directly into application such as Excel. The CSV format consists of a header row identifying each column, followed by one or more rows of data. The CSV format is supported with only data_layout parameter value "by_factor".</p>
historical_product	String	<p>The historical product name for which historical data is to be retrieved. Only a single product name is accepted per request. All available products can be retrieved at /metadata/history/historicalProducts. Cannot be used along with factor_name_list or dataset_list. CSV Format is not supported with historical_product parameter.</p> <p>User is only allowed to fetch one day data when using 'daily - by_factor' and 'daily - matrix_by_factor' layout with 'historical_product' parameter.</p>
dataset_list	Array	<p>A list of Datasets as defined in /metadata/history/datasets. dataset_list Should be used along with issuer_identifier_list to retrieve historical data for specified identifiers and factors belonging to the datasets. CSV Format is not supported with dataset_list parameter.</p>

RESPONSE

Possible Outputs

Status	Description
200	OK
206	Partial content may have been returned
400	Bad Request
401	Unauthorized

Status	Description
403	Forbidden
404	Not found
500	Internal Server Error

RESPONSE BODY

Name	Description
total_number_of_identifiers	Total number of identifiers returned.
total_number_of_unresolved_identifiers	Total number of identifiers that were not able to be resolved , An identifier is only unresolved if it did not exist at all during the specified time range.
total_number_of_batches	Total number of result batches.
current_batch	The index of the current batch.
data_request_id	The id representing this request. It can be used in future requests to retrieve subsequent batches of data.
data_request_id_expiration_time	This field specifies the expiration time of data_request_id.
unresolved_identifiers	A list of Unresolved Identifier objects.The Unresolved Identifier object is used to hold information about identifiers that were passed as parameters but were unable to be resolved for some reason and therefore data will not be returned.An identifier is only unresolved if it did not exist at all during the specified time range.
unresolved_factors	The Unresolved Factor object is used to hold information about factors that were passed as parameters but were not valid for some reason.
resolved_factors	A list of resolved Factor objects.
resolved_series_factors	A list of resolved series Factor objects.
identifiers_data_not_available	A list of requested identifiers for which all the requested factors data are not available.
factors_data_not_available	A list of factors for which data is not available for any of the requested identifiers.
messages[]	factor permissions will come under messages only in messages[] tag.

Sample Output

by_factor layout:

Input

```
{
  "issuer_identifier_list": [
    "IID000000002236325",
    "IID000000002286384"
  ],
  "factor_name_list": [
    "CONTR_COLLECTIVE_BARG_N_SEVERE",
    "CONTR_HEALTH_SAFETY_N_SEVERE",
    "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
  ],
  "start_date": "2023-10-25",
  "end_date": "2023-12-30",
  "data_layout": "by_factor",
  "data_sample_frequency": "business_month_end",
  "inherit_missing_values": true
}
```

Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fc41d2df026f167766b90adbb6d1e",
  "timestamp": "2024-04-29T16:00:29Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 2,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "0-d762a17e-4e11-47d9-acbb-093cd080b4e0-2024-04-30T16:00:29+0000",
      "data_request_id_expiration_time": "2024-04-30T16:00:29+0000",
      "resolved_factors": [
        "CONTR_COLLECTIVE_BARG_N_SEVERE",
        "CONTR_HEALTH_SAFETY_N_SEVERE"
      ],
      "resolved_series_factors": [
        "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
      ],
      "identifiers_data_not_available": [
        "IID000000002286384"
      ],
      "factors_data_not_available": [

```

```

    "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
  ],
  },
  "data": [
    {
      "requested_id": "IID000000002236325",
      "issuer_metadata": [
        {
          "ISSUERID": "IID000000002236325",
          "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
          "CIK_NUM": "0000839463",
          "ISSUER_ISIN": "XS2709528959",
          "as_of_date": "2023-10-25",
          "valid_until_date": "2023-12-30"
        }
      ],
      "factors": [
        {
          "name": "CONTR_COLLECTIVE_BARG_N_SEVERE",
          "data_values": [
            {
              "value": 0,
              "as_of_date": "2023-10-31",
              "as_at_date": "2024-04-29T16:00:00Z",
              "reference_issuer_id": "IID000000002135074",
              "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
            },
            {
              "value": 0,
              "as_of_date": "2023-11-30",
              "as_at_date": "2024-04-29T16:00:00Z",
              "reference_issuer_id": "IID000000002135074",
              "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
            },
            {
              "value": 0,
              "as_of_date": "2023-12-29",
              "as_at_date": "2024-04-29T16:00:00Z",
              "reference_issuer_id": "IID000000002135074",
              "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
            }
          ]
        },
        {
          "name": "CONTR_HEALTH_SAFETY_N_SEVERE",
          "data_values": [
            {
              "value": 0,
              "as_of_date": "2023-10-31",
              "as_at_date": "2024-04-29T16:00:00Z",
              "reference_issuer_id": "IID000000002135074",
              "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
            },
            {
              "value": 0,
              "as_of_date": "2023-11-30",

```

```

    "as_at_date": "2024-04-29T16:00:00Z",
    "reference_issuer_id": "IID000000002135074",
    "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
    },
    {
      "value": 0,
      "as_of_date": "2023-12-29",
      "as_at_date": "2024-04-29T16:00:00Z",
      "reference_issuer_id": "IID000000002135074",
      "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
    }
  ]
}

```

change_ranges layout:

Input

```

{
  "issuer_identifier_list": [
    "IID0000000002236325",
    "IID0000000002286384"
  ],
  "factor_name_list": [
    "CONTR_COLLECTIVE_BARG_N_SEVERE",
    "CONTR_HEALTH_SAFETY_N_SEVERE",
    "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
  ],
  "start_date": "2022-10-25",
  "end_date": "2023-09-30",
  "data_layout": "change_ranges",
  "inherit_missing_values": true
}

```

Output

```

{
  "status": "OK",
  "code": 200,
  "trace_id": "662fc5ad152f20d68a2c83959414a294",
  "timestamp": "2024-04-29T16:07:09Z",
  "messages": [],
  "result": {

```

```

"response_metadata": {
  "total_number_of_identifiers": 2,
  "total_number_of_unresolved_identifiers": 0,
  "total_number_of_batches": 1,
  "current_batch": 1,
  "data_request_id": "1-21fcf84a-1f01-42a3-bb86-9156e6bb82fc-2024-04-30T16:07:09+0000",
  "data_request_id_expiration_time": "2024-04-30T16:07:09+0000",
  "resolved_factors": [
    "CONTR_COLLECTIVE_BARG_N_SEVERE",
    "CONTR_HEALTH_SAFETY_N_SEVERE"
  ],
  "resolved_series_factors": [
    "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
  ],
  "identifiers_data_not_available": [
    "IID000000002286384"
  ],
  "factors_data_not_available": [
    "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
  ]
},
"data": [
  {
    "requested_id": "IID000000002236325",
    "issuer_metadata": [
      {
        "ISSUERID": "IID000000002236325",
        "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
        "CIK_NUM": "0000839463",
        "as_of_date": "2022-10-25",
        "valid_until_date": "2023-02-23"
      },
      {
        "ISSUERID": "IID000000002236325",
        "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
        "CIK_NUM": "0000839463",
        "ISSUER_ISIN": "XS2591115360",
        "as_of_date": "2023-02-23",
        "valid_until_date": "2023-05-23"
      },
      {
        "ISSUERID": "IID000000002236325",
        "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
        "CIK_NUM": "0000839463",
        "ISSUER_ISIN": "XS2628970878",
        "as_of_date": "2023-05-23",
        "valid_until_date": "2023-08-24"
      },
      {
        "ISSUERID": "IID000000002236325",
        "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
        "CIK_NUM": "0000839463",
        "as_of_date": "2023-08-24",
        "valid_until_date": "2023-09-04"
      },
      {
        "ISSUERID": "IID000000002236325",
        "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
        "CIK_NUM": "0000839463",

```

```

    "ISSUER_ISIN": "XS2677091063",
    "as_of_date": "2023-09-04",
    "valid_until_date": "2023-09-30"
  },
  ],
  "factors": [
    {
      "name": "CONTR_COLLECTIVE_BARG_N_SEVERE",
      "data_values": [
        {
          "value": 0,
          "as_of_date": "2023-01-05",
          "valid_until_date": "2023-01-06",
          "as_at_date": "2024-04-29T16:07:00Z",
          "reference_issuer_id": "IID000000002135074",
          "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-02-20",
          "valid_until_date": "2023-08-24",
          "as_at_date": "2024-04-29T16:07:00Z",
          "reference_issuer_id": "IID000000002135074",
          "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-08-31",
          "valid_until_date": "2023-09-30",
          "as_at_date": "2024-04-29T16:07:00Z",
          "reference_issuer_id": "IID000000002135074",
          "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
        }
      ]
    },
    {
      "name": "CONTR_HEALTH_SAFETY_N_SEVERE",
      "data_values": [
        {
          "value": 0,
          "as_of_date": "2023-01-05",
          "valid_until_date": "2023-01-06",
          "as_at_date": "2024-04-29T16:07:00Z",
          "reference_issuer_id": "IID000000002135074",
          "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
        },
        {
          "value": 0,
          "as_of_date": "2023-02-20",
          "valid_until_date": "2023-08-24",
          "as_at_date": "2024-04-29T16:07:00Z",
          "reference_issuer_id": "IID000000002135074",
          "reference_issuer_name": "COMMERZBANK
Aktiengesellschaft."
        }
      ]
    }
  ]
}

```

Aktiengesellschaft."

```

    "current_batch": 1,
    "data_request_id": "2-7035a24b-2629-47b8-bf1d-2438d91368cc-2024-04-30T16:08:50+0000",
    "data_request_id_expiration_time": "2024-04-30T16:08:50+0000",
    "resolved_factors": [
      "CONTR_COLLECTIVE_BARG_N_SEVERE",
      "CONTR_HEALTH_SAFETY_N_SEVERE"
    ],
    "resolved_series_factors": [
      "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
    ],
    "identifiers_data_not_available": [
      "IID000000002286384"
    ],
    "factors_data_not_available": [
      "CARBON_GOVERNMENT_IMPORTED_EMISSIONS_QUALITY_SCORE_TS"
    ]
  },
  "data": {
    "requested_ids": [
      "IID000000002286384",
      "IID000000002236325"
    ],
    "as_at_date": "2024-04-29T16:08:00Z",
    "as_of_dates": [
      "2023-10-31",
      "2023-11-30",
      "2023-12-29"
    ],
    "factors": [
      {
        "factor": "CONTR_COLLECTIVE_BARG_N_SEVERE",
        "values": [
          null,
          null,
          null
        ],
        [
          0,
          0,
          0
        ]
      },
      "reference_issuer_ids": [
        null,
        null,
        null
      ],
      [
        "IID000000002135074",
        "IID000000002135074",
        "IID000000002135074"
      ]
    ],
    "reference_issuer_names": [
      null,
      null,

```



```
{
  "issuer_identifier_list": [
    "IID000000002236325",
    "IID000000002286384"
  ],
  "factor_name_list": [
    "EVIC_GBP_TS"
  ],
  "start_date": "2024-02-03",
  "end_date": "2024-04-16",
  "data_layout": "by_factor",
  "data_sample_frequency": "business_month_end",
  "inherit_missing_values": true
}
```

Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fc6875dc3c28775fe1a1d448c9901",
  "timestamp": "2024-04-29T16:10:47Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_identifiers": 2,
      "total_number_of_unresolved_identifiers": 0,
      "total_number_of_batches": 1,
      "current_batch": 1,
      "data_request_id": "0-b193ede1-c1d3-4e3b-baef-06bbc57bea87-2024-04-30T16:10:47+0000",
      "data_request_id_expiration_time": "2024-04-30T16:10:47+0000",
      "resolved_series_factors": [
        "EVIC_GBP_TS"
      ],
      "identifiers_data_not_available": [
        "IID000000002286384"
      ]
    },
    "data": [
      {
        "requested_id": "IID000000002236325",
        "issuer_metadata": [
          {
            "ISSUERID": "IID000000002236325",
            "ISSUER_NAME": "COMMERZBANK AKTIENGESELLSCHAFT",
            "CIK_NUM": "0000839463",
            "ISSUER_ISIN": "XS2756340183",
            "as_of_date": "2024-02-03",
            "valid_until_date": "2024-04-16"
          }
        ]
      }
    ]
  }
}
```

```

    ],
    "series": [
      {
        "series_type": "SERIES_esgEsrCarbon",
        "series_type_rows": [
          {
            "series_id": "2",
            "factors": [
              {
                "name": "EVIC_GBP_TS",
                "data_values": [
                  {
                    "value": 162416.69,
                    "as_of_date": "2024-02-29",
                    "as_at_date": "2024-04-
29T16:10:00Z",
                    "reference_issuer_id":
                    "IID000000002135074",
                    "reference_issuer_name":
                    "COMMERZBANK Aktiengesellschaft."
                  },
                  {
                    "value": 162416.69,
                    "as_of_date": "2024-03-29",
                    "as_at_date": "2024-04-
29T16:10:00Z",
                    "reference_issuer_id":
                    "IID000000002135074",
                    "reference_issuer_name":
                    "COMMERZBANK Aktiengesellschaft."
                  }
                ]
              }
            ]
          },
          {
            "series_id": "9",
            "factors": [
              {
                "name": "EVIC_GBP_TS",
                "data_values": [
                  {
                    "value": 113522.78,
                    "as_of_date": "2024-02-29",
                    "as_at_date": "2024-04-
29T16:10:00Z",
                    "reference_issuer_id":
                    "IID000000002135074",
                    "reference_issuer_name":
                    "COMMERZBANK Aktiengesellschaft."
                  },
                  {
                    "value": 113522.78,
                    "as_of_date": "2024-03-29",
                    "as_at_date": "2024-04-
29T16:10:00Z",
                    "reference_issuer_id":
                    "IID000000002135074",
                    "reference_issuer_name":
                    "COMMERZBANK Aktiengesellschaft."
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  },
  {
    "series_id": "9",
    "factors": [
      {
        "name": "EVIC_GBP_TS",
        "data_values": [
          {
            "value": 113522.78,
            "as_of_date": "2024-02-29",
            "as_at_date": "2024-04-
29T16:10:00Z",
            "reference_issuer_id":
            "IID000000002135074",
            "reference_issuer_name":
            "COMMERZBANK Aktiengesellschaft."
          },
          {
            "value": 113522.78,
            "as_of_date": "2024-03-29",
            "as_at_date": "2024-04-
29T16:10:00Z",
            "reference_issuer_id":
            "IID000000002135074",
            "reference_issuer_name":
            "COMMERZBANK Aktiengesellschaft."
          }
        ]
      }
    ]
  }
]

```

```

    ]
  }
}
],
{
  "series_id": "15",
  "factors": [
    {
      "name": "EVIC_GBP_TS",
      "data_values": [
        {
          "value": 145323.83,
          "as_of_date": "2024-02-29",
          "as_at_date": "2024-04-
29T16:10:00Z",
          "reference_issuer_id":
          "IID000000002135074",
          "reference_issuer_name":
          "COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 145323.83,
          "as_of_date": "2024-03-29",
          "as_at_date": "2024-04-
29T16:10:00Z",
          "reference_issuer_id":
          "IID000000002135074",
          "reference_issuer_name":
          "COMMERZBANK Aktiengesellschaft."
        }
      ]
    }
  ],
{
  "series_id": "10",
  "factors": [
    {
      "name": "EVIC_GBP_TS",
      "data_values": [
        {
          "value": 108520.28,
          "as_of_date": "2024-02-29",
          "as_at_date": "2024-04-
29T16:10:00Z",
          "reference_issuer_id":
          "IID000000002135074",
          "reference_issuer_name":
          "COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 108520.28,
          "as_of_date": "2024-03-29",
          "as_at_date": "2024-04-
29T16:10:00Z",
          "reference_issuer_id":
          "IID000000002135074",
          "reference_issuer_name":
          "COMMERZBANK Aktiengesellschaft."
        }
      ]
    }
  ]
}

```

© 2019 MSCI Inc. All rights reserved.

```

    ]
  }
]
},
{
  "series_id": "12",
  "factors": [
    {
      "name": "EVIC_GBP_TS",
      "data_values": [
        {
          "value": 152021.52,
          "as_of_date": "2024-02-29",
          "as_at_date": "2024-04-
29T16:10:00Z",
          "reference_issuer_id":
          "IID000000002135074",
          "reference_issuer_name":
          "COMMERZBANK Aktiengesellschaft."
        },
        {
          "value": 152021.52,
          "as_of_date": "2024-03-29",
          "as_at_date": "2024-04-
29T16:10:00Z",
          "reference_issuer_id":
          "IID000000002135074",
          "reference_issuer_name":
          "COMMERZBANK Aktiengesellschaft."
        }
      ]
    }
  ],
  {
    "series_id": "11",
    "factors": [
      {
        "name": "EVIC_GBP_TS",
        "data_values": [
          {
            "value": 152200.22,
            "as_of_date": "2024-02-29",
            "as_at_date": "2024-04-
29T16:10:00Z",
            "reference_issuer_id":
            "IID000000002135074",
            "reference_issuer_name":
            "COMMERZBANK Aktiengesellschaft."
          },
          {
            "value": 152200.22,
            "as_of_date": "2024-03-29",
            "as_at_date": "2024-04-
29T16:10:00Z",
            "reference_issuer_id":
            "IID000000002135074",
            "reference_issuer_name":
            "COMMERZBANK Aktiengesellschaft."
          }
        ]
      }
    ]
  }
]

```

/METADATA/FACTORS

Retrieve data factor metadata information

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about the factor data that is available to the client. A request can filter results based on category paths, product names, and factor type (issuer, instruments or fund factors). This information can be used to formulate calls to the issuers end point which accepts category paths, product names, and factor IDs as query parameters.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/metadata/factors>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
category_path_list	array				ESG Data Factors are organized into a hierarchy of categories. Category Paths are used to locate a group of factors that belong to the specified path(s).
[array item]	string				A category path represents a series of category names which are used to reach a set of ESG Data Factors located in the specified portion of the category tree. A category path consists of one or more category names, separated by a colon.
product_name_list	array				ESG Data Factors are grouped into various product classifications. A user can specify a list of product names to limit the factors that are returned for this query.
[array item]	string				A product name is a string identifying a particular ESG Data Factor product. A full list of product names is available via the

Name	Type	Mandatory	Default value	Accepted values	Description
					/parameterValues/factorProductNames endpoint.
factor_type	string	all		fund,instrument, issuer,all	Factor Type is an optional value which will restrict the return factors to the specified type. If this option is not specified, all factor types will be returned. Available options are issuer, fund,instrument, and all. Note the options that are available are based on user permissions. Some users may only be allowed to use fund or all, other users may only be able to use issuers or all and other user may be allowed to use instrument or all.
factor_name_list	array				A caller can specify one or more factor names for which they want to retrieve factor metadata. Data will be returned for all valid factor names, along with a list of any factor names which weren't recognized. If a name isn't recognized, it could be the factor doesn't exist, or the caller doesn't have access to that factor.
[array item]	string				A factor name is a simple string value. Access to factors is permissioned and metadata is only available for factors for which the user has access.

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The factors endpoint is used to return metadata related to the data factors that can be returned in a data request made via the issuers endpoint. The metadata includes the name of the factor, the data type, a minimum/maximum value if appropriate, as well as other basic factor information.

Response Body Returned

Possible outputs for status: 20

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "da8691d185a3f4f9",
  "timestamp": "2021-05-17T12:17:18Z",
  "messages": [],
  "result": {
    "factors": [
      {
        "factor_name": "15PCT_RENEW_CAP",
        "description": "Companies with at least 15% of their installed capacity attributed to renewable sources.",
        "data_type": "Boolean",
        "factor_type": "issuer",
        "display_name": "15% Renewable Installed Capacity",
        "inheritance_allowed": true,
        "category_paths": [
          "Other:ESG Internal:Company Data"
        ],
        "product_names": [
          "Other"
        ]
      },
      {
        "factor_name": "1DEG_RED_REQ",
        "description": "The greenhouse gas direct emissions (scope 1) reduction requirements of the company, expressed in megatonnes per year in 2033, assuming a global 1.5°C target. Please refer to the Climate VaR methodology document for further details on scenario options.",
        "data_type": "Double",
        "factor_type": "issuer",
        "display_name": "1.5°C Scope 1 GHG Emissions Reduction Requirements [Mt/y]",
        "inheritance_allowed": true,
        "category_paths": [
          "Carbon and Fossil Fuel:Scenario Analysis:Climate Value at Risk:Transition risks and opportunities:1.5 degree Celsius scenario:Company summary"
        ],
        "product_names": [
          "Climate VaR",
          "Internal Only - Screening"
        ]
      }
    ]
  }
}
```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/metadata/factors?category_path_list=ESG+Ratings:Company+Summary&product_name_list=ESG+Ratings
```

Python

```
import requests url =
"https://api.msci.com/esg/data/v3.0/metadata/factors?category_path_list=ESG+Ratings:Company+Summary&product_name_list=ESG+Ratings"; headers = { "Accept": "application/json" } response =
requests.request("POST", url, headers=headers) print(response.text)
```

Java

```

HttpServletResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/metadata/factors?category_path_list=ESG+Rating
s:Company+Summary&product_name_list=ESG+Ratings").header("Accept", "application/json")
.asString();

```

C#

```

var client = new
RestClient("https://api.msci.com/esg/data/v3.0/metadata/factors?category_path_list=ESG+Rating
s:Company+Summary&product_name_list=ESG+Ratings"); var request = new RestRequest(Method.GET);
request.AddHeader("Accept", "application/json"); IRestResponse response =
client.Execute(request);

```

/METADATA/HISTORY/FACTORS

Retrieve history factor metadata information

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about the factor history metadata that is available to the client. A request can filter results based on factor names, datasets, and factor type (issuer, instruments or fund factors). This information can be used to formulate calls to the issuers end point which accepts factor IDs as query parameters.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/metadata/history/factors>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
factor_type	string		all	fund,instrument, issuer,all	Factor Type is an optional value which will restrict the return factors to the specified type. If this option is not specified, all factor types will be returned. Available options are issuer, fund, instrument, and all. Note the options that are available are based on available datasets and product permissions. Some users may only be allowed to use fund or all, other users may only be able to use issuers or all and other user may be allowed to use instrument or all.
factor_name_list	array				A caller can specify one or more factor names for which they want to retrieve factor metadata. Data will be returned for all valid factor names, along with a list of any factor names which weren't recognized. If a name isn't recognized, it could be the factor doesn't exist, or the caller doesn't have access to that factor.

Name	Type	Mandatory	Default value	Accepted values	Description
[array item]	string				A factor name is a simple string value. Factor metadata is available for all factors for which the dataset is available in repository.
dataset_list	array				A caller can specify one or more dataset names for which they want to retrieve factor metadata. Data will be returned for all valid datasets which are mapped to factors. If a name isn't recognized, it could be the dataset doesn't exist.
[array item]	string				A dataset name is a simple string value. Factor metadata is available for all datasets present in repository.
historical_product_list	array				Lists all the available historical products via historical data API.
[array item]	string				A historical name is a simple string value. Factor metadata is available for all historical product names present in repository.

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The factors endpoint is used to return metadata related to the data factors that can be returned in a data request made via the /issuers/history or /funds/history endpoint. The metadata includes the name of the factor, the data type, a minimum/maximum value if appropriate, as well as other basic factor information.

Response Body Returned

Possible outputs for status: 20

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fd4069625441f8f0fb3868bee41b2",
  "timestamp": "2024-04-29T17:08:22Z",
  "messages": [],
  "result": {
    "factors": [
      {
        "factor_header": "GOVERNMENT_RAW_POLIT_RIGHTS_FLAG",
        "description": "Indicates whether a country demonstrates weak, moderate or strong political rights. Assessment based on the Political rights data point.",
        "data_type": "String",
        "factor_type": "issuer",
        "display_name": "Political rights Flag",
        "dataset": [
          "Government Ratings Risk"
        ],
        "thematictag": [
          "PoliticalGovernance"
        ],
        "mscimodeltags": [
          "ESG Government Ratings - Governance Risk - Political Governance"
        ]
      },
      {
        "factor_header": "VAR_EXW_2DEG_ORDERLY_AGG_COLD_DEBT",
        "description": "A company's debt \"worst-case\" (95th percentile) downside or upside potential, expressed as a percentage of the company's debt market value, assuming trends in extreme cold continue along the 2°C REMIND Orderly scenario.",
        "data_type": "Double",
        "factor_type": "issuer",
        "display_name": "2°C Extreme Cold Debt Climate VaR (REMIND Orderly Aggressive outcome) [%]",
        "dataset": [
          "Physical REMIND NGFS 1.5 and 2 Degree - Issuer"
        ],
        "thematictag": [
          "ClimateChange",
          "PhysicalRisk",
          "ScenarioAnalysis"
        ],
        "mscimodeltags": [
          "Physical Risk Climate VaR"
        ]
      }
    ]
  }
}
```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/metadata/history/factors?dataset_list=Director Data,AGM Data
```

Python

```
import requests url =
"https://api.msci.com/esg/data/v3.0/metadata/history/factors?dataset_list=Director Data,AGM Data"; headers = { "Accept": "application/json" } response = requests.request("POST", url, headers=headers) print(response.text)
```

Java

```

HttpServletResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/metadata/history/factors?dataset_list=Director Data,AGM Data").header("Accept", "application/json").asString();
C#
var client = new RestClient("https://api.msci.com/esg/data/v3.0/metadata/history/factors?dataset_list=Director Data,AGM Data"); var request = new RestRequest(Method.GET);
request.AddHeader("Accept", "application/json"); IRestResponse response =
client.Execute(request);

```

/METADATA/HISTORY/DATASETS

Retrieve factor datasets information

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about the history factor datasets that is available to the client. A request can filter results based on factor type (issuer, instruments or fund factors).

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/metadata/history/datasets>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
factor_type	string		all	fund,instrument, issuer,all	Factor Type is an optional value which will restrict the return datasets to the specified type. If this option is not specified, datasets for all factor types will be returned. Available options are issuer, fund, instrument, and all. Note the options that are available are based on available datasets and product permissions. Some users may only be allowed to use fund or all, other users may only be able to use issuers or all and other user may be allowed to use instrument or all.
historical_product_list	array				List of all historical products that are available via the historical data API
[array item]	string				A historical name is a simple string value. Factor metadata is available for all historical product names present in repository.

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The datasets endpoint is used to return datasets related to the data factors available in system.

Response Body Returned

Possible outputs for status: 20

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fd4247d8a5aef97b7e93559993ce5",
  "timestamp": "2024-04-29T17:08:52Z",
  "messages": [],
  "result": {
    "datasets": [
      {
        "dataset": "Business Segments"
      },
      {
        "dataset": "Undiscounted Physical Costs 5 Degree TS -
Issuer"
      },
      {
        "dataset": "CVI Time Series"
      },
      {
        "dataset": "Emissions Intensity TS"
      },
      {
        "dataset": "USCCB_EXCLUSIONS"
      },
      {
        "dataset": "ISRAEL"
      },
      {
        "dataset": "Workforce and Diversity Data"
      },
      {
        "dataset": "Reported Non Financials Eligibility Time
Series"
      }
    ]
  }
}
```

```

        "dataset": "EU Taxonomy"
      },
      {
        "dataset": "Undiscounted Physical Costs 4 Degree TS -
Issuer"
      },
      {
        "dataset": "SFDR"
      }
    ]
  }
}

```

CODE SAMPLES

cURL

```

curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/metadata/history/datasets?factor_type=issuer

```

Python

```

import requests url =
"https://api.msci.com/esg/data/v3.0/metadata/history/datasets?factor_type=issuer"; headers =
{ "Accept": "application/json" } response = requests.request("POST", url, headers=headers)
print(response.text)

```

Java

```

HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/metadata/history/datasets?factor_type=issuer")
).header("Accept", "application/json").asString();

```

C#

```

var client = new
RestClient("https://api.msci.com/esg/data/v3.0/metadata/history/datasets?factor_type=issuer")
; var request = new RestRequest(Method.GET); request.AddHeader("Accept", "application/json");
IRestResponse response = client.Execute(request);

```

/METADATA/HISTORY/HISTORICALPRODUCTS

Retrieve factor historical products information.

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about the historical products that is available to the client. A request can filter results based on factor type (issuer, instruments or fund factors).

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/metadata/history/historicalProducts>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
factor_type	string		all	fund,instrument, issuer,all	Factor Type is an optional value which will restrict the return historicalProducts to the specified type. If this option is not specified, historicalProducts for all factor types will be returned. Available options are issuer, fund, instrument, and all. Note the options that are available are based on available historicalProducts and product permissions. Some users may only be allowed to use fund or all, other users may only be able to use issuers or all and other user may be allowed to use instrument or all.

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The historicalProducts endpoint is used to return historicalProducts related to the data factors available in system.

Response Body Returned

Possible outputs for status: 200

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "669ff71500311803ca6b7f319e1c79c1",
}
```



```

{
  "timestamp": "2024-07-23T18:31:49Z",
  "messages": [],
  "result": {
    "historical_products": [
      {
        "historical_product": "MSCI ESG Fund Ratings Historical
Data",
        "datasets": [
          "Fund Values Alignment",
          "Fund Impact",
          "Fund Climate Change",
          "Fund Risk",
          "Fund Biodiversity",
          "Fund Ratings And Scores",
          "Fund EU Susfi"
        ]
      }
    ]
  }
}

```

CODE SAMPLES

cURL

```

curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/metadata/history/historicalProducts?factor_type=issuer

```

Python

```

import requests url = "https://api.msci.com/esg/data/v3.0/metadata/history/
historicalProducts?factor_type=issuer"; headers = { "Accept": "application/json" } response =
requests.request("POST", url, headers=headers) print(response.text)

```

Java

```

HttpResponse response = Unirest.get("https://api.msci.com/esg/data/v3.0/metadata/history/
historicalProducts?factor_type=issuer").header("Accept", "application/json").asString();

```

C#

```

var client = new RestClient("https://api.msci.com/esg/data/v3.0/metadata/history/
historicalProducts?factor_type=issuer"); var request = new RestRequest(Method.GET);
request.AddHeader("Accept", "application/json"); IRestResponse response =
client.Execute(request);

```

/METADATA/HISTORY/THEMATICTAGS

Retrieve factor thematictags information

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about the history factor thematictags that is available to the client. A request can filter results based on factor type (issuer, instruments or fund factors).

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/metadata/history/thematictags>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
factor_type	string		all	fund,instrument, issuer,all	Factor Type is an optional value which will restrict the return thematictags to the specified type. If this option is not specified, thematictags for all factor types will be returned. Available options are issuer, fund, instrument, and all. Note the options that are available are based on available thematictags and product permissions. Some users may only be allowed to use fund or all, other users may only be able to use issuers or all and other user may be allowed to use instrument or all.

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The thematictags endpoint is used to return thematictags related to the data factors available in system.

Response Body Returned

Possible outputs for status: 20

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fd431430bf4997041816df1c5001d",
  "timestamp": "2024-04-29T17:09:05Z",
  "messages": [],
  "result": {
    "thematictags": [
      {
        "thematicTagName": "SustainableFinance"
      }
    ]
  }
}
```

```
{
  "thematicTagName": "ClimateChange"
}
```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/metadata/history/thematictags?factor_type=issuer
```

Python

```
import requests url =
"https://api.msci.com/esg/data/v3.0/metadata/history/thematictags?factor_type=issuer";
headers = { "Accept": "application/json" } response = requests.request("POST", url,
headers=headers) print(response.text)
```

Java

```
HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/metadata/history/thematictags?factor_type=iss
uer").header("Accept", "application/json").asString();
```

C#

```
var client = new
RestClient("https://api.msci.com/esg/data/v3.0/metadata/history/thematictags?factor_type=issu
er"); var request = new RestRequest(Method.GET); request.AddHeader("Accept",
"application/json"); IRestResponse response = client.Execute(request);
```

/METADATA/HISTORY/MSCIMODELTAGS

Retrieve factor mscimodeltags information

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about the history factor mscimodeltags that is available to the client. A request can filter results based on factor type (issuer, instruments or fund factors).

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/metadata/history/mscimodeltags>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
factor_type	string	all		fund,instrument, issuer,all	Factor Type is an optional value which will restrict the return mscimodeltags to the specified type. If this option is not specified, mscimodeltags for all factor types will be

Name	Type	Mandatory	Default value	Accepted values	Description
					returned. Available options are issuer, fund, instrument, and all. Note the options that are available are based on available mscimodeltags and product permissions. Some users may only be allowed to use fund or all, other users may only be able to use issuers or all and other user may be allowed to use instrument or all.

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The mscimodeltags endpoint is used to return mscimodeltags related to the data factors available in system.

Response Body Returned

Possible outputs for status: 20

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "662fd43fa950123abce92bc0374ac3c7",
  "timestamp": "2024-04-29T17:09:19Z",
  "messages": [],
  "result": {
    "mscimodeltags": [
      {
        "msciModelTagName": "ESG Ratings - Social - Social Opportunities"
      },
      {
        "msciModelTagName": "ESG Ratings - Social - Product Liability - Responsible Investment - Management"
      }
    ]
  }
}
```

```

    "msciModelTagName": "ESG Ratings - Social - Human
Capital - Supply Chain Labor Standards - Management"
  },
  {
    "msciModelTagName": "ESG Ratings - Environmental -
Environmental Opportunities - Opportunities in Green Building"
  },
  {
    "msciModelTagName": "ESG Ratings - Environment -
Pollution & Waste - Toxic Emissions & Waste - Management"
  },
  {
    "msciModelTagName": "ESG Ratings - Environment -
Natural Capital - Biodiversity & Land Use - Management"
  },
  {
    "msciModelTagName": "ESG Ratings - Social - Product
Liability - Consumer Financial Protection"
  },
  {
    "msciModelTagName": "ESG Ratings - Environment -
Climate Change - Product Carbon Footprint"
  }
]
}

```

CODE SAMPLES

cURL

```

curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/metadata/history/mscimodeltags?factor_type=issuer

```

Python

```

import requests url =
"https://api.msci.com/esg/data/v3.0/metadata/history/mscimodeltags?factor_type=issuer";
headers = { "Accept": "application/json" } response = requests.request("POST", url,
headers=headers) print(response.text)

```

Java

```

HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/metadata/history/mscimodeltags?factor_type=iss
uer").header("Accept", "application/json").asString();

```

C#

```

var client = new
RestClient("https://api.msci.com/esg/data/v3.0/metadata/history/mscimodeltags?factor_type=iss
uer"); var request = new RestRequest(Method.GET); request.AddHeader("Accept",
"application/json"); IRestResponse response = client.Execute(request);

```

/PARAMETERVALUES/COUNTRIES

Return a list of available countries

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a list of country code and corresponding country names that are available to the caller based on their permissions. The country codes can be used in queries for issuer data.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/countries>

HTTP Method:

GET

RESPONSE

Possible Outputs

Status	Description
200	Success
401	Unauthorized
403	No subscription available for this data
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Sample Output

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "194e01373b44a069",
  "timestamp": "2018-11-26T14:07:18Z",
  "messages": [],
  "result": {
    "countries": [
      {
        "country_name": "Poland",
        "country_code": "PL"
      },
      {
        "country_name": "British Solomon Islands",
        "country_code": "SB"
      },
      {
        "country_name": "Serbia and Montenegro",
        "country_code": "CS"
      },
      {
        "country_name": "Chad",
        "country_code": "TD"
      },
      {
        "country_name": "Angola",
        "country_code": "AO"
      },
      {
        "country_name": "Luxembourg",
        "country_code": "LU"
      },
      {
        "country_name": "Malawi",
        "country_code": "MW"
      }
    ]
  }
}
```

```

    },
    {
      "country_name": "HEARD AND MC DONALD ISLANDS",
      "country_code": "HM"
    },
    {
      "country_name": "British Indian Ocean Territory",
      "country_code": "IO"
    },
    {
      "country_name": "Finland",
      "country_code": "FI"
    },
    {
      "country_name": "Germany",
      "country_code": "DE"
    },
    {
      "country_name": "Egypt",
      "country_code": "EG"
    },
    {
      "country_name": "Bosnia and Herzegovina",
      "country_code": "BA"
    },
    {
      "country_name": "Cocos (Keeling) Islands",
      "country_code": "CC"
    },
    {
      "country_name": "SS",
      "country_code": "SS"
    },
    {
      "country_name": "Nicaragua",
      "country_code": "NI"
    },
    {
      "country_name": "Madagascar",
      "country_code": "MG"
    },
    {
      "country_name": "Virgin Islands, U.S.",
      "country_code": "VI"
    },
    {
      "country_name": "Uganda",
      "country_code": "UG"
    },
    {
      "country_name": "Brazil",
      "country_code": "BR"
    },
    {
      "country_name": "Seychelles",
      "country_code": "SC"
    },
    {
      "country_name": "Mexico",
      "country_code": "MX"
    },
    {
      "country_name": "New Zealand",
      "country_code": "NZ"
    }
  ]
}

```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/json" -X GET
https://api.msci.com/esg/data/v3.0/parameterValues/countries
```

Python

```
import requests url = "https://api.msci.com/esg/data/v3.0/parameterValues/countries"; headers
= { "Accept": "application/json" } response = requests.request("POST", url, headers=headers)
print(response.text)
```

Java

```
HttpResponse response =
Unirest.get("https://api.msci.com/esg/data/v3.0/parameterValues/countries") .header("Accept",
"application/json") .asString();
```

C#

```
var client = new RestClient("https://api.msci.com/esg/data/v3.0/parameterValues/countries");
var request = new RestRequest(Method.GET); request.AddHeader("Accept", "application/json");
IRestResponse response = client.Execute(request);
```

/PARAMETERVALUES/COVERAGES

Return a list of available issuer coverage universes

ENDPOINT DESCRIPTION

ESG Data is collected for a large number of issuers. In most cases, clients will want to limit the number of issuers they get in a response to only those who have data for a particular coverage universe. For example, a client may wish to only get issues that have ESG Ratings data. The coverage universe is a way to limit the response to issuers that meet particular data characteristics. This endpoint will return a list of data coverage names that are available to the caller.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/coverages>

HTTP Method:

GET

Available coverages:

```
"agr",
"bissr",
"esg_ratings",
"esg_controversies",
"governance_metrics",
"government_ratings",
"iva",
"climate_var",
"climate_change_metrics"
```

/PARAMETERVALUES/ESGINDUSTRIES

Return a list of available ESG Industries

ENDPOINT DESCRIPTION

This endpoint provides a list of ESG Industry codes and names. The industry codes can be used in other endpoints that take an ESG industry id as a parameter.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/esgIndustries>

HTTP Method:

GET

/PARAMETERVALUES/FACTORCATEGORYPATHS

Retrieve a list of category paths used to organize factors

ENDPOINT DESCRIPTION

This endpoint is used to return a list of Category Paths which identify the functional areas for all factors available to the caller. Factors are organized by categories in a tree-like structure. Category Paths represent one way which a client can indicate which factors they would like to retrieve when getting data for a set of issuers. This call will return a JSON object containing a list of all Category Paths to which a client is permitted. The request can contain optional parameters to limit the paths to those which start with a set of characters, or which contain a set of characters (or both).

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/factorCategoryPaths>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
contains	string				A string which will be used to match a string of characters anywhere inside of a Category Path. For example, a client might wish to see all categories that are related to 'Carbon'. If 'Carbon' is specified, all category paths containing the string 'Carbon' will be returned. Matches will be case-insensitive.
starts_with	string				A string which will be used to match the starting characters of a Category Path. For example, a client might wish to see all categories that are related to 'ESG Ratings'. If 'ESG Ratings' is specified, all category paths starting with 'ESG Ratings' will be returned. Matches will be case-insensitive.

/PARAMETERVALUES/FACTORPRODUCTNAMES

Retrieve a list of product names used to identify groupings of factors

ENDPOINT DESCRIPTION

ESG Data Factors are grouped into product categories. This request is used to provide a list of products to which the client has been granted access. The product names can be used in other queries to retrieve a list of factors relating to the specified product(s).

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/factorProductNames>

HTTP Method:

GET

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
contains	string				The contains parameter is used to limit the response to those product names that contain the given string. This parameter can be combined with the starts_with parameter.
starts_with	string				The starts_with parameter is used to limit the response to those product names that start with the given string. This parameter can be combined with the contains parameter.

/PARAMETERVALUES/FUNDASSETCLASSES

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a list of fund asset class names. The names can be used in the fund_asset_class_list parameter of the funds endpoint to restrict results to funds that belong to the specified fund asset class names.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/fundAssetClasses>

HTTP Method:

GET

/PARAMETERVALUES/FUNDASSETUNIVERSES

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a list of fund asset universe names. The names can be used in the fund_asset_universe_list parameter of the funds endpoint to restrict results to funds that belong to the specified fund asset universe names.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/fundAssetUniverses>

HTTP Method:

GET

/PARAMETERVALUES/FUNDDOMICILES

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a list of fund domicile names. The names can be used in the fund_domicile_list parameter of the funds endpoint to restrict results to funds that belong to the specified fund domicile names.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/fundDomiciles>

HTTP Method:

GET

/PARAMETERVALUES/FUNDLIPPERGLOBALCLASSES

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a list of Lipper global class names. The names can be used in the fund_lipper_global_class_list parameter of the funds endpoint to restrict results to funds that belong to the specified Lipper global class names.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/fundLipperGlobalClasses>

HTTP Method:

GET

/PARAMETERVALUES/GICSSUBINDUSTRIES

Retrieve a list of available GICS Sub-Industries

ENDPOINT DESCRIPTION

This endpoint is used to retrieve a list of GICS SubIndustry names and codes. The codes can be used in other endpoints that allow for the specification of GICS SubIndustry ids.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/gicsSubIndustries>

HTTP Method:

GET

/PARAMETERVALUES/INDEXES

Retrieve a list of available indexes

ENDPOINT DESCRIPTION

This endpoint returns a list of indexes to which the accessing account has been permitted. The returned value includes both the index name and the index code. The index code can be used in other requests that can use index IDs as input to the request.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/parameterValues/indexes>

HTTP Method:

GET

/INSTRUMENT-REFERENCE

HTTP Method:

GET

ENDPOINT DESCRIPTION

This endpoint is used to retrieve information about all the security identifiers which are linked to an issuer. A request can filter results based on identifiers list. Response will contain attributes as 'data_request_id' and 'batch_id' that can be used for getting response for specific batch. For batch request, request body will have only two parameters as 'batch_id' and 'data_request_id'. Maximum of 50 identifiers can be sent in a single request. The response will contain parameters as 'total_number_of_instrument_identifiers', 'data_request_id' and 'total_number_of_batches'. 'total_number_of_instrument_identifiers' implies the total number of identifiers that are available for all the issuers. CUSIP and SEDOL are optional, will be based on license. Possible values in factor_name_list ["LOOKUP_ID", "ISSUERID", "MARKET_STATUS", "ASSET_TYPE", "CINS", "TICKER", "SEDOL", "ISIN", "CUSIP", "EXCHANGE", "FIGI", "SHARECLASSFIGI", "COMPOSITEFIGI", "ISSUER_NAME", "SECURITY_COUNTRY"]

A request without any request parameters will, by default, fetch data for 10 issuer instrument identifiers in a single batch.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/instrument-reference>

Query parameters

Name	Type	Mandatory	Default value	Accepted values	Description
identifier_list	Array	No			A caller can specify one or more identifiers for which they want to retrieve identifiers data. Data will be returned for the specified identifiers. In case if none of the identifiers are passed, response will return all the available security identifiers for all the issuers in paginated format. And contains values up to 50
factor_name_list	Array	No			A caller can specify one or more factors name for which they want to retrieve identifiers data.
num_of_issuer_identifiers_in_batch	Integer	No			A caller can specify the number of issuer identifiers to be returned in a single batch. num_of_issuer_identifiers_in_batch must be of a multiple of 10 and fall within the range of 10 – 50. This parameter can be used independently or only along with factor_name_list. In all other cases, it will be ignored.
data_request_id	String	If batch_id is passed			The data_request_id is an identifier that is assigned to a data request. It acts as a key to retrieve additional batches of information. The key remains valid for as long as data is being requested but will expire at some point like if data_request_id is generated before the daily data load process. The data_request_id is always displayed in the response_metadata section of the response.
batch_id	Integer	If data_request_id is passed			The batch_id parameter is used to identify which batch of data to retrieve. Multiple requests with the same data_request_id and same batch_id will return the same results. This will be true until the data_request_id expires after a period of inactivity.

RESPONSE

Possible Outputs

Status	Description
200	Success
400	Malformed request
401	Unauthorized
403	No subscription available for this data
404	Requested resource was not found
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The instrument reference is used to retrieve information about all the security identifiers which are linked to an issuer. A request can filter results based on identifiers list. Response will contain attributes as 'data_request_id' and 'batch_id' that can be used for getting response for specific batch. For batch request, request body will have only two parameters as 'batch_id' and 'data_request_id'. Maximum of 50 identifiers can be sent in a single request. The response will contain parameters as 'total_number_of_instrument_identifiers', 'data_request_id' and 'total_number_of_batches'. 'total_number_of_instrument_identifiers' implies the total number of security identifiers that are available for all the issuers.

Sample Output

Possible outputs for status: 200

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "65688ff0e3ef0296ee51d58c1be42a2c",
  "timestamp": "2023-11-30T13:36:48Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_instrument_identifiers": 11,
      "total_number_of_unresolved_instrument_identifiers": 0,
      "total_number_of_batches": 12576,
      "current_batch": 1,
      "data_request_id": "3-4c6b1012-cee5-4965-98bc-ca0355398b9f-2023-12-01T13:36:48Z",
      "data_request_id_expiration_time": "2023-12-0113:36:48.492652451"
    },
    "instrument_identifiers": [
      {
        "requested_id": "IID0000000001842895",
        "data": [

```

```

        "ISIN": "US70338A1034",
        "CUSIP": "70338A103",
        "SEDOL": "2260222",
        "TICKER": "PATRNL",
        "ISSUER_ID": "IID000000001842895",
        "LOOKUP_ID": "UIF000000001587679",
        "ASSET_TYPE": "CORP",
        "MARKET_STATUS": "ISS",
        "FIGI": "BBG00007NF04",
        "SECURITY_COUNTRY": "US",
        "ISSUER_NAME": "PATRONS' LEGACY"
    },
    {
        "ISIN": "US70337U1007",
        "CUSIP": "70337U100",
        "SEDOL": "2268549",
        "TICKER": "PATRNL",
        "ISSUER_ID": "IID000000001842895",
        "LOOKUP_ID": "UIF000000001626782",
        "ASSET_TYPE": "CORP",
        "MARKET_STATUS": "ISS",
        "FIGI": "BBG00001Y5H9",
        "SECURITY_COUNTRY": "US",
        "ISSUER_NAME": "PATRONS' LEGACY"
    }
],
{
    "requested_id": "IID000000001981141",
    "data": [
        {
            "ISIN": "US86311PAA57",
            "CUSIP": "86311PAA5",
            "TICKER": "STRATS",
            "ISSUER_ID": "IID000000001981141",
            "LOOKUP_ID": "UIF0000000010058755",
            "ASSET_TYPE": "CORP",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG00008Q716",
            "SECURITY_COUNTRY": "US",
            "ISSUER_NAME": "STRATS"
        }
    ]
},
{
    "requested_id": "IID000000001786293",
    "data": [
        {
            "ISIN": "US4372962054",
            "CUSIP": "437296205",
            "SEDOL": "2541734",
            "TICKER": "HOMOWN",
            "ISSUER_ID": "IID000000001786293",
            "LOOKUP_ID": "UIF000000008133931",
            "ASSET_TYPE": "PRFD",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG000006154",
            "SECURITY_COUNTRY": "US",
            "ISSUER_NAME": "HOME OWNERSHIP FUNDING
CORPORATION"
        }
    ]
},
{
    "requested_id": "IID000000001787718",
    "data": [
        {
            "ISIN": "US8667492032",
            "CUSIP": "866749203",
            "SEDOL": "2242662",
            "TICKER": "SHG",

```

```

        "ISSUER_ID": "IID000000001787718",
        "LOOKUP_ID": "UIF000000009478875",
        "ASSET_TYPE": "PRFD",
        "MARKET_STATUS": "ISS",
        "FIGI": "BBG000002BP4",
        "SECURITY_COUNTRY": "US",
        "ISSUER_NAME": "SUN FINANCING"
      }
    ],
    {
      "requested_id": "IID000000001832643",
      "data": [
        {
          "ISIN": "US60686HAA59",
          "CUSIP": "60686HAA5",
          "TICKER": "MMCAPS",
          "ISSUER_ID": "IID000000001832643",
          "LOOKUP_ID": "UIF000000008041940",
          "ASSET_TYPE": "CORP",
          "MARKET_STATUS": "ISS",
          "FIGI": "BBG00000R0Z0",
          "SECURITY_COUNTRY": "KY",
          "ISSUER_NAME": "MMCAPS FUNDING LIMITED/MMCAPS
FUNDING INC"
        }
      ]
    },
    {
      "requested_id": "IID000000001817947",
      "data": [
        {
          "ISIN": "US434803AA08",
          "CUSIP": "434803AA0",
          "TICKER": "HOLDEN",
          "ISSUER_ID": "IID000000001817947",
          "LOOKUP_ID": "UIF000000008010986",
          "ASSET_TYPE": "CORP",
          "MARKET_STATUS": "ISS",
          "FIGI": "BBG00002TMR5",
          "SECURITY_COUNTRY": "KY",
          "ISSUER_NAME": "HOLDEN FUNDING CORPORATION"
        }
      ]
    },
    {
      "requested_id": "IID000000001789005",
      "data": [
        {
          "ISIN": "US19074VAA08",
          "CUSIP": "19074VAA0",
          "TICKER": "COTRS",
          "ISSUER_ID": "IID000000001789005",
          "LOOKUP_ID": "UIF000000034432295",
          "ASSET_TYPE": "CORP",
          "MARKET_STATUS": "ISS",
          "SECURITY_COUNTRY": "US",
          "ISSUER_NAME": "COBALTS Trust for Sprint
Capital Notes"
        }
      ]
    },
    {
      "requested_id": "IID000000001789974",
      "data": [
        {
          "ISIN": "US7244818743",
          "CUSIP": "724481874",
          "TICKER": "PBI",
          "ISSUER_ID": "IID000000001789974",
          "LOOKUP_ID": "UIF000000020221284",

```



```

    "ASSET_TYPE": "PRFD",
    "MARKET_STATUS": "ISS",
    "FIGI": "BBG0001JLSD7",
    "SECURITY_COUNTRY": "US",
    "ISSUER_NAME": "PITNEY BOWES INTERNATIONAL
HOLDINGS INC"
  }
]
},
{
  "requested_id": "IID000000001832386",
  "data": [
    {
      "ISIN": "US114802AA98",
      "CUSIP": "114802AA9",
      "TICKER": "BROSIS",
      "ISSUER_ID": "IID000000001832386",
      "LOOKUP_ID": "UIF000000008040933",
      "ASSET_TYPE": "CORP",
      "MARKET_STATUS": "ISS",
      "FIGI": "BBG00095VC98",
      "SECURITY_COUNTRY": "US",
      "ISSUER_NAME": "BROSIS FINANCE LLC"
    }
  ]
},
{
  "requested_id": "IID000000001850218",
  "data": [
    {
      "ISIN": "GB00B1347K44",
      "CUSIP": "EG3187275",
      "TICKER": "LCHLTD",
      "ISSUER_ID": "IID000000001850218",
      "LOOKUP_ID": "UIF0000000021765758",
      "ASSET_TYPE": "CORP",
      "MARKET_STATUS": "ISS",
      "FIGI": "BBG0000H6KY3",
      "SECURITY_COUNTRY": "GB",
      "ISSUER_NAME": "LONDON CLEARING HOUSE LIMITED"
    }
  ]
}
]
}
}

```

CODE SAMPLES

cURL

```

curl -i -H "Accept: application/json"
-X GET https://api.msci.com/esg/data/v3.0/instrument-reference

```

Python

```

import requests url = "https://api.msci.com/esg/data/v3.0/instrument-reference"; headers = {
"Accept": "application/json" } response = requests.request("POST", url, headers=headers)
print(response.text)

```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse response = Unirest.get("https://api.msci.com/esg/data/v3.0/instrument-reference") .header("Accept", "application/json") .asString();
```

C#

```
var client = new RestClient("https://api.msci.com/esg/data/v3.0/instrument-reference"); var request = new RestRequest(Method.GET); request.AddHeader("Accept", "application/json"); IRestResponse response = client.Execute(request);
```

HTTP Method:

POST

ENDPOINT DESCRIPTION

In addition to requesting instrument reference data via a GET request, users can also create a POST request where the POSTed data consists of the various query parameters that are available on the GET request. The POST form of this endpoint allows for potentially large amounts of data to be specified. For example, a client may have a list of identifier IDs that they wish to retrieve data for. Specifying the ids in a GET request may exceed the maximum length allowed for a GET request. A POST request has no such limit.

It is important to note that any POST request specify the Content-Type being used for submission. For example, if JSON is being used to send data, the Content-Type should be application/json, otherwise the request may be rejected with a 400 status code and an 'Invalid formatting' error message.

REQUEST

Service URLs:

<https://api.msci.com/esg/data/v3.0/instrument-reference>

Body payload

Name	Type	Mandatory	Default value	Accepted values	Description
identifier_list [array-item]	Array	No			A caller can specify one or more identifiers for which they want to retrieve identifiers data. Data will be returned for the specified identifiers. In case if none of the identifiers are passed, response will return all the available security identifiers for all the issuers in paginated format. And contains values up to 50
factor_name_list [array-item]	Array	No			A caller can specify one or more factors name for which they want to retrieve identifiers data.
num_of_issuer_identifiers_in_batch	Integer	No			A caller can specify the number of issuer identifiers to be returned in a single batch. num_of_issuer_identifiers_in_batch must be of a multiple of 10 and fall within the range of 10 – 50. This parameter can be used independently or only along with factor_name_list. In all other cases, it will be ignored.

Name	Type	Mandatory	Default value	Accepted values	Description
data_request_id	String	If batch_id is passed			<p>The data_request_id is an identifier that is assigned to a data request. It acts as a key to retrieve additional batches of information. The key remains valid for as long as data is being requested but will expire at some point like if data_request_id is generated before the daily data load process.</p> <p>The data_request_id is always displayed in the response_metadata section of the response.</p>
batch_id	Integer	If data_request_id is passed			<p>The batch_id parameter is used to identify which batch of data to retrieve. Multiple requests with the same data_request_id and same batch_id will return the same results. This will be true until the data_request_id expires after a period of inactivity.</p>

RESPONSE

Possible Outputs

Status	Description
200	Success
400	Malformed request
401	Unauthorized
403	No subscription available for this data
404	Requested resource was not found
500	Unexpected Server Error
503	Service Unavailable

Output Encoding

application/json

Output Description

The instrument reference is used to retrieve information about all the security identifiers which are linked to an issuer and the result is returned in JSON format only.

Sample Output

Possible outputs for status: 200

```
{
  "status": "OK",
  "code": 200,
  "trace_id": "65688ff0e3ef0296ee51d58c1be42a2c",
  "timestamp": "2023-11-30T13:36:48Z",
  "messages": [],
  "result": {
    "response_metadata": {
      "total_number_of_instrument_identifiers": 11,
      "total_number_of_unresolved_instrument_identifiers": 0,
      "total_number_of_batches": 12576,
      "current_batch": 1,
      "data_request_id": "3-4c6b1012-cee5-4965-98bc-ca0355398b9f-2023-12-01T13:36:48Z",
      "data_request_id_expiration_time": "2023-12-0113:36:48.492652451"
    },
    "instrument_identifiers": [
      {
        "requested_id": "IID000000001842895",
        "data": [
          {
            "ISIN": "US70338A1034",
            "CUSIP": "70338A103",
            "SEDOL": "2260222",
            "TICKER": "PATRNL",
            "ISSUER_ID": "IID000000001842895",
            "LOOKUP_ID": "UIF000000001587679",
            "ASSET_TYPE": "CORP",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG00007NF04",
            "SECURITY_COUNTRY": "US",
            "ISSUER_NAME": "PATRONS' LEGACY"
          },
          {
            "ISIN": "US70337U1007",
            "CUSIP": "70337U100",
            "SEDOL": "2268549",
            "TICKER": "PATRNL",
            "ISSUER_ID": "IID000000001842895",
            "LOOKUP_ID": "UIF000000001626782",
            "ASSET_TYPE": "CORP",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG00001Y5H9",
            "SECURITY_COUNTRY": "US",
            "ISSUER_NAME": "PATRONS' LEGACY"
          }
        ]
      },
      {
        "requested_id": "IID000000001981141",
        "data": [
          {
            "ISIN": "US86311PAA57",
            "CUSIP": "86311PAA5",
            "TICKER": "STRATS",
            "ISSUER_ID": "IID000000001981141",
            "LOOKUP_ID": "UIF0000000010058755",
            "ASSET_TYPE": "CORP",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG00008Q716",
            "SECURITY_COUNTRY": "US",
            "ISSUER_NAME": "STRATS"
          }
        ]
      },
      {
        "requested_id": "IID000000001786293",
```

```

    "data": [
      {
        "ISIN": "US4372962054",
        "CUSIP": "437296205",
        "SEDOL": "2541734",
        "TICKER": "HOMOWN",
        "ISSUER_ID": "IID000000001786293",
        "LOOKUP_ID": "UIF000000008133931",
        "ASSET_TYPE": "PRFD",
        "MARKET_STATUS": "ISS",
        "FIGI": "BBG000006154",
        "SECURITY_COUNTRY": "US",
        "ISSUER_NAME": "HOME OWNERSHIP FUNDING
CORPORATION"
      }
    ],
    {
      "requested_id": "IID000000001787718",
      "data": [
        {
          "ISIN": "US8667492032",
          "CUSIP": "866749203",
          "SEDOL": "2242662",
          "TICKER": "SHG",
          "ISSUER_ID": "IID000000001787718",
          "LOOKUP_ID": "UIF000000009478875",
          "ASSET_TYPE": "PRFD",
          "MARKET_STATUS": "ISS",
          "FIGI": "BBG000002BP4",
          "SECURITY_COUNTRY": "US",
          "ISSUER_NAME": "SUN FINANCING"
        }
      ],
      {
        "requested_id": "IID000000001832643",
        "data": [
          {
            "ISIN": "US60686HAA59",
            "CUSIP": "60686HAA5",
            "TICKER": "MMCAPS",
            "ISSUER_ID": "IID000000001832643",
            "LOOKUP_ID": "UIF000000008041940",
            "ASSET_TYPE": "CORP",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG00000R0Z0",
            "SECURITY_COUNTRY": "KY",
            "ISSUER_NAME": "MMCAPS FUNDING LIMITED/MMCAPS
FUNDING INC"
          }
        ],
        {
          "requested_id": "IID000000001817947",
          "data": [
            {
              "ISIN": "US434803AA08",
              "CUSIP": "434803AA0",
              "TICKER": "HOLDEN",
              "ISSUER_ID": "IID000000001817947",
              "LOOKUP_ID": "UIF000000008010986",
              "ASSET_TYPE": "CORP",
              "MARKET_STATUS": "ISS",
              "FIGI": "BBG000002TMR5",
              "SECURITY_COUNTRY": "KY",
              "ISSUER_NAME": "HOLDEN FUNDING CORPORATION"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    "requested_id": "IID000000001789005",
    "data": [
      {
        "ISIN": "US19074VAA08",
        "CUSIP": "19074VAA0",
        "TICKER": "COTRS",
        "ISSUER_ID": "IID000000001789005",
        "LOOKUP_ID": "UIF000000034432295",
        "ASSET_TYPE": "CORP",
        "MARKET_STATUS": "ISS",
        "SECURITY_COUNTRY": "US",
        "ISSUER_NAME": "COBALTS Trust for Sprint
Capital Notes"
      }
    ],
    {
      "requested_id": "IID000000001789974",
      "data": [
        {
          "ISIN": "US7244818743",
          "CUSIP": "724481874",
          "TICKER": "PBI",
          "ISSUER_ID": "IID000000001789974",
          "LOOKUP_ID": "UIF000000020221284",
          "ASSET_TYPE": "PRFD",
          "MARKET_STATUS": "ISS",
          "FIGI": "BBG0001JLSD7",
          "SECURITY_COUNTRY": "US",
          "ISSUER_NAME": "PITNEY BOWES INTERNATIONAL
HOLDINGS INC"
        }
      ],
      {
        "requested_id": "IID000000001832386",
        "data": [
          {
            "ISIN": "US114802AA98",
            "CUSIP": "114802AA9",
            "TICKER": "BROSIS",
            "ISSUER_ID": "IID000000001832386",
            "LOOKUP_ID": "UIF000000008040933",
            "ASSET_TYPE": "CORP",
            "MARKET_STATUS": "ISS",
            "FIGI": "BBG00095VC98",
            "SECURITY_COUNTRY": "US",
            "ISSUER_NAME": "BROSIS FINANCE LLC"
          }
        ],
        {
          "requested_id": "IID000000001850218",
          "data": [
            {
              "ISIN": "GB00B1347K44",
              "CUSIP": "EG3187275",
              "TICKER": "LCHLTD",
              "ISSUER_ID": "IID000000001850218",
              "LOOKUP_ID": "UIF000000021765758",
              "ASSET_TYPE": "CORP",
              "MARKET_STATUS": "ISS",
              "FIGI": "BBG0000H6KY3",
              "SECURITY_COUNTRY": "GB",
              "ISSUER_NAME": "LONDON CLEARING HOUSE LIMITED"
            }
          ]
        }
      ]
    }
  ]
}

```

CODE SAMPLES

cURL

```
curl -i -H "Accept: application/json" -d "{}" -X POST
https://api.msci.com/esg/data/v3.0/instrument-reference
```

Python

```
import requests import json url = " https://api.msci.com/esg/data/v3.0/instrument-reference";
payload = json.dumps( {} ) headers = { "Accept": "application/json" } response =
requests.request("POST", url, data=payload, headers=headers) print(response.text)
```

Java

The following example uses Unirest for Java, a lightweight HTTP request library.

```
HttpResponse response = Unirest.post("https://api.msci.com/esg/data/v3.0/instrument-
reference") .header("Accept", "application/xml") .body("{}") .asString();
```

C#

```
var client = new RestClient("https://api.msci.com/esg/data/v3.0/instrument-reference"); var
request = new RestRequest(Method.POST); ); request.AddHeader("Accept", "application/json");
IRestResponse response = client.Execute(request);
```

CONTACT US

esgclientservice@msci.com

AMERICAS

+ 1 212 804 5299

EUROPE, MIDDLE EAST & AFRICA

+ 44 20 7618 2510

ASIA PACIFIC

+ 612 9033 9339

ABOUT MSCI ESG RESEARCH PRODUCTS AND SERVICES

MSCI ESG Research products and services are provided by MSCI ESG Research LLC, and are designed to provide in-depth research, ratings and analysis of environmental, social and governance-related business practices to companies worldwide. ESG ratings, data and analysis from MSCI ESG Research LLC. are also used in the construction of the MSCI ESG Indexes. MSCI ESG Research LLC is a Registered Investment Adviser under the Investment Advisers Act of 1940 and a subsidiary of MSCI Inc.

ABOUT MSCI

For more than 40 years, MSCI’s research-based indexes and analytics have helped the world’s leading investors build and manage better portfolios. Clients rely on our offerings for deeper insights into the drivers of performance and risk in their portfolios, broad asset class coverage and innovative research.

Our line of products and services includes indexes, analytical models, data, real estate benchmarks and ESG research.

MSCI serves 99 of the top 100 largest money managers, according to the most recent P&I ranking.

For more information, visit us at www.msci.com.

NOTICE AND DISCLAIMER

This document and all of the information contained in it, including without limitation all text, data, graphs, charts (collectively, the “Information”) is the property of MSCI Inc. or its subsidiaries (collectively, “MSCI”), or MSCI’s licensors, direct or indirect suppliers or any third party involved in making or compiling any Information (collectively, with MSCI, the “Information Providers”) and is provided for informational purposes only. The Information may not be modified, reverse-engineered, reproduced or disseminated in whole or in part without prior written permission from MSCI.

The Information may not be used to create derivative works or to verify or correct other data or information. For example (but without limitation), the Information may not be used to create indexes, databases, risk models, analytics, software, or in connection with the issuing, offering, sponsoring, managing or marketing of any securities, portfolios, financial products or other investment vehicles utilizing or based on, linked to, tracking or otherwise derived from the Information or any other MSCI data, information, products or services.

The user of the Information assumes the entire risk of any use it may make or permit to be made of the Information. NONE OF THE INFORMATION PROVIDERS MAKES ANY EXPRESS OR IMPLIED WARRANTIES OR REPRESENTATIONS WITH RESPECT TO THE INFORMATION (OR THE RESULTS TO BE OBTAINED BY THE USE THEREOF), AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, EACH INFORMATION PROVIDER EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES (INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF ORIGINALITY, ACCURACY, TIMELINESS, NON-INFRINGEMENT, COMPLETENESS, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) WITH RESPECT TO ANY OF THE INFORMATION.

Without limiting any of the foregoing and to the maximum extent permitted by applicable law, in no event shall any Information Provider have any liability regarding any of the Information for any direct, indirect, special, punitive, consequential (including lost profits) or any

Index returns do not represent the results of actual trading of investible assets/securities. MSCI maintains and calculates indexes, but does not manage actual assets. Index returns do not reflect payment of any sales charges or fees an investor may pay to purchase the securities underlying the index or Index Linked Investments. The imposition of these fees and charges would cause the performance of an Index Linked Investment to be different than the MSCI index performance.

The Information may contain back tested data. Back-tested performance is not actual performance, but is hypothetical. There are frequently material differences between back tested performance results and actual results subsequently achieved by any investment strategy.

Constituents of MSCI equity indexes are listed companies, which are included in or excluded from the indexes according to the application of the relevant index methodologies. Accordingly, constituents in MSCI equity indexes may include MSCI Inc., clients of MSCI or suppliers to MSCI. Inclusion of a security within an MSCI index is not a recommendation by MSCI to buy, sell, or hold such security, nor is it considered to be investment advice.

Data and information produced by various affiliates of MSCI Inc., including MSCI ESG Research LLC and Barra LLC, may be used in calculating certain MSCI indexes. More information can be found in the relevant index methodologies on www.msci.com.

MSCI receives compensation in connection with licensing its indexes to third parties. MSCI Inc.’s revenue includes fees based on assets in Index Linked Investments. Information can be found in MSCI Inc.’s company filings on the Investor Relations section of www.msci.com.

MSCI ESG Research LLC is a Registered Investment Adviser under the Investment Advisers Act of 1940 and a subsidiary of MSCI Inc. Except with respect to any applicable products or services from MSCI ESG

other damages even if notified of the possibility of such damages. The foregoing shall not exclude or limit any liability that may not by applicable law be excluded or limited, including without limitation (as applicable), any liability for death or personal injury to the extent that such injury results from the negligence or willful default of itself, its servants, agents or sub-contractors.

Information containing any historical information, data or analysis should not be taken as an indication or guarantee of any future performance, analysis, forecast or prediction. Past performance does not guarantee future results.

The Information should not be relied on and is not a substitute for the skill, judgment and experience of the user, its management, employees, advisors and/or clients when making investment and other business decisions. All Information is impersonal and not tailored to the needs of any person, entity or group of persons.

None of the Information constitutes an offer to sell (or a solicitation of an offer to buy), any security, financial product or other investment vehicle or any trading strategy.

It is not possible to invest directly in an index. Exposure to an asset class or trading strategy or other category represented by an index is only available through third party investable instruments (if any) based on that index. MSCI does not issue, sponsor, endorse, market, offer, review or otherwise express any opinion regarding any fund, ETF, derivative or other security, investment, financial product or trading strategy that is based on, linked to or seeks to provide an investment return related to the performance of any MSCI index (collectively, "Index Linked Investments"). MSCI makes no assurance that any Index Linked Investments will accurately track index performance or provide positive investment returns. MSCI Inc. is not an investment adviser or fiduciary and MSCI makes no representation regarding the advisability of investing in any Index Linked Investments.

Research, neither MSCI nor any of its products or services recommends, endorses, approves or otherwise expresses any opinion regarding any issuer, securities, financial products or instruments or trading strategies and MSCI's products or services are not intended to constitute investment advice or a recommendation to make (or refrain from making) any kind of investment decision and may not be relied on as such. Issuers mentioned or included in any MSCI ESG Research materials may include MSCI Inc., clients of MSCI or suppliers to MSCI, and may also purchase research or other products or services from MSCI ESG Research. MSCI ESG Research materials, including materials utilized in any MSCI ESG Indexes or other products, have not been submitted to, nor received approval from, the United States Securities and Exchange Commission or any other regulatory body.

Any use of or access to products, services or information of MSCI requires a license from MSCI. MSCI, Barra, RiskMetrics, IPD, and other MSCI brands and product names are the trademarks, service marks, or registered trademarks of MSCI or its subsidiaries in the United States and other jurisdictions. The Global Industry Classification Standard (GICS) was developed by and is the exclusive property of MSCI and Standard & Poor's. "Global Industry Classification Standard (GICS)" is a service mark of MSCI and Standard & Poor's.