

# Refinitiv Contributions Channel

## Development Quick Start Guide v3.1

Contributions Channel can be accessed using the Refinitiv Real-Time C++ and JAVA APIs.

Namely, these are the Enterprise Message API (EMA) and the Enterprise Transport API (ETA). Both Linux and Windows clients are supported.

Alternatively, Websocket access is also available for programming languages that support Websocket and JSON such as Python, Perl, Ruby and CSharp using .Net framework.

## AUTHENTICATION

Clients subscribing to the service will be provided with new machine account IDs. These credentials will be authenticated against our authentication platform during login.

Separate credentials are provided for A-side and B-side service DNS endpoints and must be used for correct operation.

The passwords supplied with these credentials will be 30 characters long. Passwords should be sent in clear text but should be obfuscated by the client for secure storage.

Passwords must be changed every 90 days and a request should be made via the helpdesk to have this performed. We recommend that the password change is made outside of market hours.

Please contact us through the following link <https://my.refinitiv.com/content/mytr/en/helpandsupport.html>

## CONNECTION SEMANTICS

Contributions Channel is an OMM Provider that supports MarketPrice OMM data.

Connecting applications are OMM Consumers that must form Transport Layer Security (TLS) encrypted OMM Tunnel Stream connections to the Contributions Channel on port 443.

Connecting applications must form connections to both the A-side and B-side addresses within the same region to provide resiliency and high availability. Whilst connections are made to both a-side and B-side endpoints data must only be sent to one at any given time. That is, one of the connections must be live/primary and the other should be used as standby/secondary to switch over in case of a connection or service issue on the live/primary.

If Refinitiv requires the connected application to switch channels a service degraded status message will be provided. At this point no more data should be sent to the degraded service channel and applications should send data to the viable service channel instead.

Connecting applications should implement a structured backoff in their reconnect code. We recommend that connection recovery should start at 10 seconds and on each login denial, double the connection recovery interval (10, 20, 40, 80, 160 etc) until a maximum back off time of 600 seconds is reached, where it will continue at that value. Once the reconnection login succeeds, the recovery interval should be reset back to the 10 second default value.

Tunnel Stream bandwidth available to the connecting application will be advertised by Contributions Channel at stream formation and throughout its lifetime. Adherence to the available bandwidth limit is managed by the API.

The rate at which an application can publish updates per second is determined by the advertised message rate.

Contributions Channel Connectivity Guide, VIP service addresses and Product Change Notifications can be found on MyRefinitiv <https://my.refinitiv.com/content/mytr/en/product/thomson-reuters-contributions-channel.html>

You will need to be authenticated on the developer portal to be able to access the links provided below and may register yourself for an account.

## Refinitiv Real-Time SDK

Refinitiv Real-Time SDK 2.0.0.L1 and above is required to access Contributions Channel (or Elektron APIs 3.1.1 and above available in SDK 1.1.1).

Please read the APIs description to determine which API best suits your needs.

C++ <https://developers.refinitiv.com/en/api-catalog/elektron/elektron-sdk-cc>

Java <https://developers.refinitiv.com/en/api-catalog/elektron/elektron-sdk-java>

API Quick Start Guides are available on the Developer Community Website and are a recommended starting point.

C++ <https://developers.refinitiv.com/en/api-catalog/elektron/elektron-sdk-cc/quick-start>

Java <https://developers.refinitiv.com/en/api-catalog/elektron/elektron-sdk-java/quick-start>

Tutorials for posting data to Contributions Channel with supporting source code can be obtained at the following links.

C++ Enterprise Message API [EMA](#)

C++ Enterprise Transport API [ETA](#)

Java Enterprise Message API [EMA](#)

Java Enterprise Transport API [ETA](#)

## Websocket

Websocket access is available for programming languages that support Websocket and JSON such as Python, Perl, Ruby and CSharp using .Net framework.

Contributions Channel currently supports a single service endpoint address in each region for Websocket clients.

Tutorial for contributing data to Contributions Channel using Websocket with required libraries and example code can be obtained at the following links.

Python Tutorial <https://developers.refinitiv.com/en/api-catalog/elektron/refinitiv-websocket-api/tutorials#contributing-data-to-refinitiv-contributions-channel>

Refinitiv Contributions Channel WebSocket Example [https://developers.refinitiv.com/content/dam/devportal/api-families/elektron/web-socket-api/downloads/websocket\\_authentication\\_rcc.zip](https://developers.refinitiv.com/content/dam/devportal/api-families/elektron/web-socket-api/downloads/websocket_authentication_rcc.zip)

Python Quick Start Guide and links to Required libraries <https://developers.refinitiv.com/en/api-catalog/elektron/refinitiv-websocket-api/quick-start>

Refinitiv Contributions Channel uses the Market Price Domain model (examples of field values supported can be found in the WebSocket API Specifications).

[https://developers.refinitiv.com/content/dam/devportal/refinitivrealtimeapi\\_pdfs/websocket\\_api\\_protocol\\_specification.pdf](https://developers.refinitiv.com/content/dam/devportal/refinitivrealtimeapi_pdfs/websocket_api_protocol_specification.pdf) -

Contributions Channel service endpoint addresses and Product Change Notifications can be found on MyRefinitiv <https://my.refinitiv.com/content/mytr/en/product/thomson-reuters-contributions-channel.html>

## TEST TOOL

A simple API Client Contribution Tool is available for download from the Refinitiv Real-Time SDK Tools Downloads area and the direct link is also shown below.

[https://developers.refinitiv.com/content/dam/devportal/api-families/elektron/elektron-sdk-c/downloads/api\\_client\\_contribution\\_b\\_1\\_0.zip](https://developers.refinitiv.com/content/dam/devportal/api-families/elektron/elektron-sdk-c/downloads/api_client_contribution_b_1_0.zip)

When run, this application will form a tunnel stream connection to Contributions Channel and update fields for a given instrument.

All configuration settings including instrument name, fields and data values, host address and credentials are to be configured within the included "SampleContributionsChannelPublisher.ini" file.

## SUPPORT

For support, please contact us through MyRefinitiv.

<https://my.refinitiv.com/content/mytr/en/helpandsupport.html>

In addition, you can contact an API support specialist via our Developer Community.

<https://community.developers.refinitiv.com/index.html>

Your Account Manager can provide information about membership to the Refinitiv Developer Community if you are not already a member.