New Change FX - API Manual

The following document provides a description of the complete set of NCFX API's.

NCFX has made our data and calculations available via API to allow our data and analytics to be machine readable. This ensures that NCFX data and processes can be automated and performed in real time.

NCFX APIs are organised into thematic families of Benchmarks:

- Spot Data
- Forward Data
- Calculations
- Volatility
- Interpolation
- Transaction Cost Analysis

Version 1.0
12th March 2021

Spot Data

NCFX spot data APIs provide access to the New Change FX benchmark and reference price databases. The APIs support 4 types of functionality, each with their own API end point. NCFX produces and records a rate up to 20 times per second.

- Timeseries - returns all mid-rates published between two specified time stamps including the start time and excluding the end time.
- Last Rate - returns a single value for the last NCFX Mid-Rate before the specified time.
- OHLC - returns the open high low close at a user defined interval between two datetime stamps.
- TWAP - returns the time weighted average price between two timestamps over the entire period, or over user defined intervals during the chosen period.

Forward Data

NCFX forwards data APIs provide access to the NCFX database of standard tenors and NDFs. The APIs support two types of functionality, each with their own endpoint. Forwards are reported in true decimals while NDFs are reported as outright prices.

- Timeseries - returns all the delta change updates to the forward mid-rates including the start time and excluding the end time for all standard tenors up to 1 year, and depending on the currency pair out to as much as 5 years.
- Last Rate – returns a single value for the last NCFX Mid-Rate before the specified timestamp for the required tenor.

Calculations

The Calculations APIs allow users to run the same calculations that are used in NCFX standard TCA outputs. All calculations are run on NCFX underlying benchmark or reference data. The calculations APIs support three critical Transaction Cost Analysis (TCA) functions callable via individual endpoints:

- Implementation Shortfall - Calculates the change in portfolio value through the lifecycle of a trade. We propose 4 time slices that clients can measure their transactions against. These time slices are meant to capture the creation of an FX exposure, the moment that exposure is recognised on a trading desk, the moment that order is externalised to the market and the moment the order is executed. If a client only has one time stamp such as the execution timestamp, we propose default values that each client can change manually as appropriate. All costs are recalculated into USD at the prevailing USD exchange rate at each time slice.

- Decay – Calculates mark-outs of transaction costs, comparing the execution rate to a series of mid-rates, at specified intervals, before and after the execution time stamp. Rising USD costs before the execution timestamp may signal information leakage or trading in the direction of the prevailing trend. Falling USD costs represent a potential for the market maker to clear the trade at a profit, in addition to the instantaneous spread between the execution rate at the mid-rate prevailing at the execution time stamp.

- Price Impact – Calculates the effective spread given from the arrival time to the execution time. (In Europe this is the regulatory definition of transaction costs). Price impact is further broken down into its constituent parts:
  - **Permanent impact** – this is the change in mid-rates over time.
  - **Temporary impact** – this is the instantaneous spread between the Bid or Offer and the prevailing mid-rate at the execution timestamp.

Volatility

The intuition behind NCFX proprietary Unit Cost of Volatility (UCV) model comes from the Grinold Kahn model of market impact. At its simplest, the inventory risk approach to market impact says that it should cost one day's volatility to trade one day's volume. The USD cost of volatility applies this logic to intra-day periods and calculates the USD cost of volatility as: the amount traded x volatility over the time taken to complete the trade, converted into USD at the average USD conversion rate. Realized volatility is calculated using Yang Zhang volatility estimates from interval based Open High Low Close data. NCFX currently offers two APIs in the volatility theme.

- USD Cost of Volatility – This calculates the inventory risk of liquidating a currency position over a user specified period. The realized volatility is standardised into USD terms. Because we cannot know what volatility will be in the future, USD cost of volatility forms the basis of pre-trade cost expectations.

- Unit Cost of Volatility – Normalises transaction costs for the risk taken to complete the transaction. The API calculates the effective spread and divides this by the corresponding USD cost of volatility. This enables users to compare transaction costs across different currency pairs and market conditions.

Interpolation

Linear (straight-line) interpolation is a tried and tested methodology for calculating a rate between two standard tenors, except on particular days, such as the year end, quarter end or on days where a change in rate is expected, such as for scheduled central bank announcements.

NCFX uses a screen process to determine whether a particular day used in the calculation period falls on a "special" day and applies a weighted average technique to correct for the step change or jump in the curve. Where no special days have been identified, NCFX uses straight-line interpolation. NCFX provides an Forwards Calculator to calculate the value of broken dates. We support 5 API endpoint functions.

- Interpolator Last Rate – Calculates a single broken date value.
- Interpolator Curve – Calculates to a specific broken date and every day for a user specified number of days. A classic use case might be to calculate the curve between IMM dates.
- Interpolator Timeseries – calculates all the values from the underlying timeseries database of the pillar dates.
- Spread – Calculates the forward forward rate, comparing two value dates, such as two IMM dates.
- Spread Timeseries – Calculates the evolution of the fwd fwd rate from two value dates, from the underlying timeseries database.

Transaction Cost Analysis

The Transaction Cost Analysis (TCA) API allows users to write directly to a TCA engine, to run large trade files, and perform instant, machine readable TCA.

- TCA API – performs analysis on trade files. Calculates the transaction cost from a single timestamp, reducing transaction costs into a single number for spread costs.

- Instant TCA – performs instantaneous calculations on a limited number of trades (up to 4 trades) per call.