Trader Tools
The eFXperts

e-FX paves the road for FX algorithmic trading

by Mitchell Mittman, President and Co-founder of TraderTools LLC

The advent of e-FX has made algorithmic trading possible in the FX world by providing efficient distribution of market rates and the ability to automatically execute trades according to algorithmic parameters.

Ever-increasing efficiency of the e-FX part of the trading process allows for greater improvements in algorithmic systems. The result? Increased trade flow through electronic FOREX execution systems and the continued growth of FX trading volumes globally.

Algorithmic trading

Before discussing the effect of e-FX on algorithmic trading, a definition of what algorithmic trading is in order.

> **The Tower Group defines algorithmic trading as:**
> Taking a buy or sell order of a defined quantity, and placing it into a quantitative model that automatically generates the timing and size of orders based on the specific goals of the algorithm. This approach is typically oriented around trading to a specific benchmark, price or time.

By combining a number of factors (including market price, wait-time, order-type and other characteristics) into the buy or sell decision, algorithmic trading "shapes" a position or strategy far more effectively than pure program trading. Algorithmic trading requires reliable market data and efficient execution of market orders; without electronic execution, algorithmic trading systems are futile. Algorithmic trading relies solely on the execution of orders by direct computer-to-computer systems.

Algorithmic trading crosses the boundaries of both buy- and sell-side. Large financial institutions tout their algorithms to their institutional customers.

The buy side uses their own algorithmic trading platforms to make efficient use of multiple lines of liquidity and execution systems. The flexibility of algorithmic trading platform allows the CTA to precisely "shape" the position by increasing the probability of precise executions.

Market data – the evolution

Not too long ago, the only streaming source for real-time data in FX for most institutions was a blended rate made available from sources such as Reuters, Bridge and others. This rate represented a "guesstimate" of the market and its direction. The customer then contacted their banks to get a "real" or executable price.

This process was telephone-based and moved to a chat-like environment where prices were quoted on the screen but here too, the process was strictly request for quote, or RFQ.

Algorithmic systems that rely on a constant stream of executable information could not have been deployed until the advent of true *streaming executable rates*.

Automated execution venues

Currently there are three primary execution venues available within the FX world:

1. Banks' proprietary platforms
2. Portal (e.g. FXall, Currenex, HotspotFX, etc)
3. Direct application programming interface (API) from bank to customer

For each of the above channels, a method of transferring information to and from the bank to the customer is necessary. While this could be defined as an API, the requirements for algorithmic

TraderTools LLC
30 Rockefeller Plaza
Suite 4244
New York, NY 10112
USA

Tel +1-212-333-3770
Fax +1-212-333-3790

For additional information, please visit our website
www.tradertools.com

trading are much more stringent.

These requirements include:

- Reliable price transmission
- Tagged price for execution
- Trade execution in a narrowly defined time period
- Reliable and timely trade confirmations
- Comprehensive error reporting

APIs have only just recently matured to incorporate the above requirements thus paving the way for algorithmic trading platforms.

Who's got the "Best Price"?

The process of algorithmic trading

A decision that any architect of an algorithmic trading system must make is whether to rely on a single liquidity provider's rates or to receive a multiple number of streams and funnel them through a "best price" mechanism.

Each path has its advantages and disadvantages. The single liquidity provider is simpler to implement and manage; the algorithm must respond to a smaller number of inputs and can therefore execute trades faster with a higher probability of successful completion due to less time spent in calculation. In the event of system malfunction, there is one rates provider to solve the problem.

The multi-stream approach is far more complex to design. There are many more inputs to the filtering system, which have an adverse affect on the overall performance. These additional factors include multiple sources of prices, tolerance checks on more inputs, and a more complicated setup for defining source inputs. For example, a "funnel" is required to monitor all liquidity streams and update the "best price" for any given currency pair, using rules for tolerances.

On the other hand, multiple liquidity providers give the algorithmic trading platform a wider range of potential prices. Byproducts of multiple liquidity providers are aggregation and built-in fault tolerance in the event of feed failures.

Once implemented, the algorithmic platform receives a "best price" rate (either from a single liquidity source or from the funnel of multiple liquidity providers) and then executes trades based on the signals generated by the algorithms.

Obstacles along the road

The majority of the liquidity providers require trading on any given "price" within a very short period, typically measured in hundreds of milliseconds. When calculating the roundtrip to process a rate from the liquidity provider's network to the client's WAN, very little time remains (potential latency) for applying a complicated algorithmic system.

This latency can be broadly attributed to the efficiency of the algorithmic trading platform itself and of the data delivery mechanism.

By nature, algorithmic systems are sensitive to the quantity of data received in a given period. Foreign exchange rates update very rapidly and if the system handles multiple data feeds, the algorithmic trading system can easily be overloaded.

The requirement for partial fills and aggregation further increases the complexity of the system and the potential for latency.

The primary delivery mechanism for these streams of liquidity is the public Internet. While many parts of the globe have adequate Internet coverage, many parts do not. Even areas that do have broadband Internet access experience outages during parts of the day at peak conditions.

Additionally, the financial institutions themselves often do not have adequate WAN connectivity between branches. This can add to the time lag when executing trades received remotely.

The sheer complexity of an algorithmic system combined with the inherent latency of the data transport medium (usually the public Internet) leads to a high potential rate of failure. This is a recognized problem and calls for inventive solutions such as better compression and adaptive transfers.

Overcoming the obstacles

Internet latency

One of the difficulties of the majority of the liquidity providers is that information is streamed using some version of XML. XML is ASCII-based and is particularly inefficient in terms of network bandwidth.

All solutions to this issue involve some sort of compression technique. XML tends to compress very well. As in all solutions using compression, it is imperative that the system monitor whether the compression actually improves the throughput or not. Most of the available solutions in network compression techniques are adaptive by nature.

Since the data distribution medium is the Internet, in most cases TCP is used as the underlying protocol. TCP is not particularly efficient in recognizing and handling network bottlenecks.

TCP's default behavior is to throttle output and only slowly return to a "normal" level of performance. A number of companies offer a combined hardware and software service that provides "network-based route optimization" to address this problem. This solution places servers at various points on the Internet backbones, providing a multiplex approach to data delivery. This requires hardware and software to be installed at the delivery points, resulting in capital expenditures.

Based on research, it is clear that there is a market niche available for those companies that recognize the opportunity to streamline Internet traffic, especially in cases of high-volume, low-packet traffic.

Internal system latency

Once the price packet enters the customer's system, the "funnel" must be flexible enough to adapt to market conditions. This requires a multi-threaded architecture, with a central server monitoring failure rates, best price, and number of updates per currency pair across liquidity streams. Weights to the foregoing are assigned and those streams that are more likely to succeed are granted preferential status over those that have had or are having stale conditions.

In essence, the "best price engine" should not only choose the best price available, it should choose the best price that has the highest probability of a successful execution. A successful implementation of this type of best price engine, coupled with a high degree of Internet throughput, will result in an algorithmic system that generates successful executed trades.

Summary

Algorithmic trading will play an increasingly important role in the overall FX world in the near future. This is primarily due to the confluence of a number of factors, including increasing efficiency of data transmission, the role of the major liquidity providers in providing access to their rates, and the sophisticated requirements of the buy side.

Technology issues must be solved before widespread adaptation of algorithmic trading platforms can take place. This article has mentioned a few potential solutions to these obstacles.