

Whilst market data volume continues to grow, the nirvana for traders remains unchanged: identify a trading opportunity before your competitors. To achieve this, it has become more and more likely that you will need to develop an algorithm. But how do you develop such an algorithm?

You may use something like Microsoft Excel with software to connect direct to the exchange or broker. This works for basic algorithms but raises problems as they become complex.

First, there is latency. Your spreadsheet runs on a desktop machine, connecting to another piece of software converting Excel commands into calls to a market adapter. This adapter converts the message for each execution destination and having checked it against your risk limits sends it to the market. With the multitude of steps here and other software you may be running, you are not getting optimal execution by any means.

Next there is functionality. Spreadsheets are designed to provide real-time answers to real-time questions, but do not understand the concept of time, and in particular historic time which is essential for algorithmic trading.

The key to algorithmic trading

Darren Simons, sales account manager for RTD Tango and backtester, RTS Realtime Systems and Steffen Gemuenden, co-CEO of RTS Realtime Systems Group examine the key criteria for developing an effective algorithmic solution.

Tick the box

So what tool should you use? There are a number of key criteria your algorithmic trading platform needs to support:

Performance – the system must have minimum latency and be highly scalable. To minimise latency, the system should run on a server perhaps hosted at the exchange itself. It should also speak the same language as the exchange to reduce unneeded latency cost. Having a ‘killer algorithm’ but submitting the order two seconds late is analogous to buying a lottery ticket today for last week’s winning numbers. Scalability is relevant since you are unlikely to be running a single strategy – if your strategy monitors 200 instruments against generating orders based on a statistical analysis you need a platform that can support it.

Flexibility – the development environment you use needs to be geared towards algorithmic trading. Out goes Excel add-ins, and funky drag and drop boxes and circles;



Darren Simons, sales account manager, RTD Tango and backtester, RTS Realtime Systems



Steffen Gemuenden, co-CEO, RTS Realtime Systems Group

in comes adaptive scripting languages designed for algorithms – things like OHLC over specified time period, VWAP calculation, real-time market depth analysis are the tools for your trade here.

Multiple venue support – as algorithmic trading becomes further adopted the winners must look at many markets concurrently. Thus your system must connect to a multitude of exchanges across multiple asset classes.

Simulation – a trading idea, which works first time, is great but very unlikely. Therefore your system must offer realistic trade simulation (what would happen if I ran this strategy right now) and what would have happened historically with various strategy parameters (backtest).

The successful algorithmic trader will need both the vision and the toolset to be successful in this marketplace. ■

RTS

