



Павел Корякин

Управляющий активами ИФК «Опцион»

О НЕКОТОРЫХ ОСОБЕННОСТЯХ ИСПОЛЬЗОВАНИЯ БИРЖЕВЫХ ТОРГОВЫХ СИСТЕМ

Торговые роботы — это, прежде всего, программы часто не сильно отличающиеся от любого офисного приложения, используемого ежедневно. В такие программы «зашивается» некоторая логика, в которой условно можно выделить три компонента:

1. обработка входящих данных и расчет тех или иных показателей по ним (в том числе накопленная позиция, размер гарантийного обеспечения и т. д.);
2. принятие решений в зависимости от соотношения рассчитанных показателей;
3. выставление/снятие заявок, мониторинг их состояния и формирование позиций к исполнению.

Варианты реализации роботизированных систем сильно зависят от инфраструктуры, в которую он будет встраиваться. Инфраструктура, в свою очередь, находится в зависимости от возможностей участника торгов.

Большинство торговых терминалов могут экспортировать данные по протоколу *DDE*, в том числе в формате *Excel*. В зависимости от навыков можно сделать либо простой расчет каких-то индексных показателей, либо, написав соответствующие макросы, расчет кумулятивных индикаторов. После расчета индикаторов отрабатывается логика принятия того или иного решения. Но в таком варианте роботизированной программы никакие заявки автоматически отправляться не будут. Однако робототорговец может «научить» *Excel* сигнализировать о той или иной возможности, например, подсвечивая ячейку на мониторе нужным цветом.

В таком варианте заявки можно генерировать автоматически, но для их отправки в торговый терминал будет необходимо писать макросы.

Более опытные в программировании инвесторы имеют значительные преимущества. Они могут не ограничивать себя рамками возможностей *Excel* и использовать все возможности языков программирования высокого уровня¹.

Наиболее часто используется схема взаимодействия роботов с торговыми терминалами. Для того чтобы была возможность связать разрабатываемого торгового робота с торговым терминалом, последний должен обладать двумя функциями:

1. экспорт биржевой информации;
2. импорт заявок.

И то, и другое может реализовываться несколькими путями. Экспорт может быть осуществлен во внешнюю базу данных, из которой робот может забирать информацию по протоколу *DDE*, непосредственно в алгоритм работы робота или архаично через текстовые файлы. То же самое касается импорта данных. Если терминал предоставляет хотя бы один из вариантов, можно вооружиться языком программирования и создавать саму логику.

Разумеется, помимо информации из биржевого терминала создатель робота может реализовать в логике и опрос новостных *web*-ресурсов, и парсинг, и анализ текста на предмет поиска каких-нибудь ключевых слов, которые будут учитываться при принятии торговых решений.

Первый вариант простейшего робота, созданного при помощи *Excel*, имеет следующие преимущества:

- гибкость языка программирования;
- многократное преимущество в скорости;
- масштабируемость, возможность разделения на процессы, службы и проч.;
- возможность использования готовых библиотек;
- возможность создания произвольного пользовательского интерфейса.

Однако этот вариант нельзя назвать оптимальным с точки зрения скорости передачи данных. Другое дело, когда подключение роботизированных систем к торговой системе биржи производится через т. н. биржевой шлюз. Как правило, такой подход используется профессиональными участниками и не используется частными инвесторами. Это обусловлено и

регламентными ограничениями работы бирж, и ценами на услуги.

Так или иначе, информация из торговой системы биржи уходит через биржевой шлюз. Однако чтобы попасть к конечному пользователю, данные из шлюза сначала попадают на сервер брокера, откуда потом рассылаются подписчикам. Очевидно, что задержки при получении данных и выставления заявок будут обусловлены наличием промежуточного звена — сервера брокера. Кроме того, терминал частного инвестора подключается к серверу брокера через интернет, а значит появляются дополнительные промежуточные узлы.

Но брокер, являющийся профессиональным участником, получает биржевые данные на свой сервер напрямую через шлюз биржи не по интернету, а по выделенной сети, что многократно повышает надежность и скорость. С точки зрения биржи, не важно что находится на другой стороне — сервер брокера или торговый автомат, который работает напрямую через шлюз. И сервер, и робот — это лишь программы, общающиеся со шлюзом биржи.

Роботизированные торговые системы используются всеми классами инвесторов — от крупнейших инвестиционных институтов до частных инвесторов, управляющих небольшими портфелями. Уровень технологических возможностей для создания «автоматов» зависит лишь от материальных возможностей и опыта программирования.

Если частные инвесторы накладывают свою логику на торговые терминалы и не могут себе позволить играть в быстрые игры вроде межбиржевого арбитража, то крупный инвестиционный банк может располагать свои сервера в здании биржи, максимизируя скорость передачи данных.

Частные инвесторы, как правило, создают роботов, логика которых подразумевает, что скорость выставления заявок не велика. Это могут быть «автоматы», занимающиеся прогнозированием движения цен на основе численного анализа временных ря-

дов. Или роботы, оптимизирующие опционный портфель в той или иной метрике.

Интересный факт

Итак, скорость передачи данных и расчетов — один из важнейших факторов при создании роботизированных торговых систем. Потребность в скорости породила интересные решения при построении вычислительных кластеров и, как следствие, особенности спроса на рынке труда. Дело в том, что только на первый взгляд кажется, что современные персональные компьютеры могут решить любую вычислительную задачу за доли секунды. На самом деле это не так. С ростом размерности задач (например, при добавлении лишнего актива в оптимизируемый портфель) вычислительная нагрузка растет значительно быстрее. Скажем, в процессе принятия решения может потребоваться регулярно перемножать матрицы размером 1000x1000. Операция кажется тривиальной, только если ее нужно осуществить один раз. Если такие операции необходимо проводить непрерывно, то при каждой новой сделке это практически полностью поглощает процессорное время.

Решить вопрос возможно с использованием кластеров. Но это, во-первых, дорого, а во-вторых, проблематично с точки зрения поддержки. Альтернативное решение нашлось. Дело в том, что подобные задачи легко «распараллеливаются» при использовании многоядерных процессоров, необходимых для нормального функционирования видеокарт. А вся графика в компьютере — это перемножение огромного количества матриц. И один компьютер с многоядерным процессором и снабженный четырьмя видеокартами, по вычислительной мощности легко заменяет кластер из нескольких сотен машин. Поэтому в последнее время на рынке труда в секторе создания роботизированных торговых программ начали пользоваться спросом специалисты, умеющие создавать программное обеспечение для видеокарт. ■

¹ Субъективное мнение автора: самыми популярными языками, на которых создаются роботы являются C++ и C#.