

## ИСТОРИЯ В ЛИЦАХ



### На языке C++

Вячеслав КОНОНЕНКО, инженер-программист, сотрудник Отдела программного обеспечения KASE, один из разработчиков торговой системы

---

В конце 1995 года нас троих взяли на работу на Казахстанскую валютную биржу (тогда еще КМВБ). Насколько я знаю и помню, взяли нас в связи с требованием поставщика закупленной тогда программной системы для торговли ценными бумагами о наличии пяти программистов для ее поддержки. Так как двое программистов на бирже уже были, то дополнительно взяли нас троих.

Наняли нас, возможно, в последний момент, поскольку, не проработав и недели, мы отправились на курсы в Москву, в учебный центр компании Hewlett-Packard, которая являлась поставщиком компьютерного оборудования. Причем, что довольно специфично для этой компании (насколько я знаю, немногие компании поставляют компьютерное оборудование столь широкого спектра), поставлены были как сервер и клиентские ПК, так и сетевое оборудование этого производства. Причем сеть была специфичная – 100VG Anylan, разработанная HP, и эту сеть я ни до, ни после больше нигде не встречал.

Но, следует признать, к работе сети нареканий никаких не было: как и другое оборудование этой компании, оно показало себя только с лучшей стороны. Хотя протестировали мы его на «износ». Это была довольно занимательная история.

Дело в том, что хотя HP предоставляла широкий спектр оборудования, но источники бесперебойного питания UPS они не делали. Поэтому UPS были поставлены производства фирмы Merlin Gerin. Вследствие ошибки монтажа поставщиков системы, при выходе UPS в аварийный режим (так называемый BYPASS, когда питание передается напрямую со входа) происходил перекоп фазы в сторону самой нагруженной. И на двух других появлялось напряжение 380В (при нормальных 220В), что привело к каскадному срабатыванию защиты блоков питания в торговых ПК. Но, повторяю, оборудование HP показало себя с наилучшей стороны, даже после такого сбоя ни один блок питания компьютера не вышел из строя, а блоки питания мониторов потребовали только замены детали защиты. Тем не менее поволновались мы изрядно, да и времени после работы было потрачено немало для выявления этой проблемы.

Курсы, которые проводила компания HP, касались либо их собственной версии Unix – HP UX, либо программирования на C и C++. На этих языках была написана

торговая система. Курсы были хорошо организованы, дали много полезной информации и новых знаний, хотя большинство из нас уже программировали на С++ и имели опыт. Откровением (возможно, только для меня) была женщина-преподаватель, проводившая курсы по программированию. Поскольку встречать программистов не мужского пола до этого просто не доводилось (никакого шовинизма, просто наблюдение), поначалу встретили ее довольно скептически. Но уровень ее знаний и преподавания предмета оказались на высоте, что мы довольно быстро поняли, когда к середине курса, к нашему огромному сожалению, преподаватель заболела. Вместо нее преподавать С++ прислали какого-то ученого. На вопрос, знает ли он С++, он ответил, что именно С++ не особо, но работает в институте, где разрабатывают языки программирования, поэтому знает их все и быстро разберется. Однако его оптимизма хватило ненадолго, первый день он худо-бедно провел, подвергся критике и больше не появлялся. Этот курс мы, к сожалению, так и не дослушали...

Торговая система, которую мы должны были сопровождать, уже была установлена и успешно работала на ММВБ. Серверную часть, работавшую на сервере HP UX с использованием СУБД Informix, написала австралийская компания на языке С. Клиентскую часть (рабочие места трейдеров) написала московская компания на языке С++ под MS Windows, она уже инсталлировала и обучала нас работе с системой на месте.

Названий фирм я за давностью лет не помню, да и не думаю, что это важно. Система была сделана для торговли ценными бумагами и стоила в районе миллиона долларов. С этим, кстати, связана довольно курьезная ситуация. Как-то водитель рассказывал, что ездил на таможню получать товар, высланный этой фирмой. Ему выдали кассету с магнитной лентой (так называемый стриммер, для архивного хранения информации на сервере) и спросили, не боится ли он везти ее один, поскольку заявленная стоимость этой кассеты 900 тысяч долларов. Он сказал, что не боится, и привез кассету без приключений. А заявленная стоимость и передача кассеты были чистой формальностью, вся информация к тому времени у нас уже была. Мало того, переданная кассета даже не подходила к стриммеру, который стоял на сервере, и ценность имела нулевую.

Но я отвлекся. Итак, система была разработана для торговли ценными бумагами, но на бирже также проходили торги валютой. И система там использовалась, написанная на СУБД «Clipper». Она была рассчитана для работы на локальном компьютере и не соответствовала требованиям быстрого действия и возможности работы с количеством трейдеров, которые тогда уже были и должны были вскоре появиться. Мы узнали, что велись переговоры на адаптацию приобретенной системы под валютные торги, и что была запрошена сумма в несколько сотен тысяч долларов. На это мы заявили, что напишем такую же, и даже лучше. И это обойдется бирже в сумму на порядок меньше. После обсуждения и небольшой торговли было достигнуто согласие, и мы вдвоем, Саша Гаврилюк и я, приступили к написанию новой торговой системы для валютных торгов.

Поскольку опыта написания подобных систем у нас не было, мы решили отталкиваться от того, что уже работает, но при этом улучшить эти идеи. Здесь, конечно, не идет речи о воровстве кода. Да это в большинстве своем

малопродуктивно. Но за основу мы взяли идеи и организацию существующей системы и написали свою с нуля. Как я уже говорил, серверная часть была написана на языке С для операционной системы HP UX и составляла три части – загрузочную, рабочую и выгружающую. Первой запускалась загрузочная программа, которая создавала разделяемую память (shared memory) фиксированного размера, читала данные базы данных Informix, загружала их в эту память и затем завершалась. Далее запускалась рабочая часть, которая работала с данными в разделяемой памяти, осуществляла торги и по их окончании завершалась.

Затем запускалась последняя часть, загружавшая результаты обратно в базу данных. Эта система позволяла обслуживать большое количество клиентов (рабочая часть содержала несколько процессов, по одному на клиента), но имела существенные недостатки. Первый – фиксированный размер структур данных, который задавался конфигурацией. Что требовало аккуратного подхода и было весьма неудобно: если памяти было выделено недостаточно, система просто переставала добавлять такие данные (например, переставала принимать заявки на торги). Второй проблемой было то, что выгрузка данных из памяти в базу данных происходила отдельным процессом, поэтому данные для бэкофиса были недоступны до конца торгов, что тоже порождало неудобство – при наличии данных по ходу торгов можно было бы производить выборки по базе и собирать статистику. Внедряя же эту функциональность в саму торговую систему нецелесообразно – усложняется код, и возрастает вероятность появления ошибки в самой торговой системе без особой на то причины.

Поэтому после многочисленных обсуждений и обдумывания было решено хотя и загружать данные в разделяемую память, но делать это динамически, без предварительного жесткого лимита. Также выгрузку в базу данных сделать отдельным клиентским процессом, который постоянно выгружает данные по мере их поступления, и со скоростью, какую позволяет база данных. Отдельно стоит упомянуть, что данная версия HP UX (9.0, если мне не изменяет память) не поддерживала многопоточность (но, насколько я знаю, она уже появилась в то время), а мы имели дело с многопроцессорной системой, то данные необходимо было хранить в разделяемой памяти для доступа к данным нескольких процессов одновременно. При этом один процесс был «главным»: он вносил изменения, но с клиентами по сети не общался. Остальные процессы общались с клиентами, отвечали на их запросы на чтение, а запросы на запись передавали главному процессу. Отдельный процесс в качестве «клиента» имел базу данных Informix и сбрасывал изменившиеся данные по мере поступления. Язык написания – С++, тогда еще HP UX, использовала технологию front-end – код на С++ компилировался в код на языке С, а тот уже компилировался в исполняемый код компилятором С (который должен быть на каждом unix).

Работать над проектом нам приходилось по вечерам и выходным, поскольку от основных обязанностей нас никто не освобождал. Очень помогало желание заработать на собственный автомобиль, цену которого должно было покрыть вознаграждение, ну и, конечно, было просто интересно построить такую систему. Причем последнее даже мотивировало сильнее.

С благодарностью вспоминаю руководство биржи, которое приняло решение предоставить транспорт для нашего возвращения домой, поскольку засиживались мы частенько допоздна, и добираться домой на общественном транспорте после столь напряженной работы было выше наших сил. Отдельно с благодарностью вспоминаю водителей биржи, которые довольно терпеливо допоздна ждали, когда мы закончим работу, не проявляя недовольства, и относились с пониманием к нашей ситуации. Конечно, стоит особо упомянуть долготерпение моей супруги, которая в тот период почти не видела меня – только либо поздно вечером, либо иногда на выходных.

С постоянной переработкой по ночам тоже был связан довольно курьезный случай. Голова на тот момент была забита программированием системы, поэтому зачастую я программировал даже во сне, что было достаточно утомительно. Но в одном из таких снов я увидел ошибку в программе и, проснувшись и приехав на работу, я эту ошибку с удивлением обнаружил. Причем ошибка была не из приятных – проявляла бы себя только в определенных условиях, поэтому ее вылавливание и исправление обычным способом отняло бы много времени и сил.

Сам момент запуска нашей системы мне, честно говоря, за давностью лет не запомнился. После победной эйфории наступило время кропотливой работы по выявлению и устранению ошибок, так как любая новая программная система не может их не содержать. Запомнился победный момент – купленную торговую систему для ценных бумаг заменили на нашу! Но радость от победы сменилась замешательством – наша система такую нагрузку просто не тянула.

Переход торгов ценными бумагами на нашу систему заморозили, пришлось разбираться. Дело оказалось в том, что в валютных торгах участвовало счетное количество финансовых инструментов (две или три валюты). Каждый финансовый инструмент содержал несколько групп связанных с ним данных – сделки, заявки и проч. Поэтому при добавлении финансовых инструментов количество групп возрастало пропорционально, а в торговле ценными бумагами инструментов было на несколько порядков больше. Пришлось срочно оптимизировать программу – писать имплементацию сбалансированного двоичного дерева. К сожалению, современной библиотеки языка C++ (STL) у нас тогда не было. Второй переход на нашу систему уже был удачным и более волнительным.

Опыт написания этой системы, конечно, был бесценен, а идеи, заложенные в систему, выдержали проверку временем. Сейчас, глядя на систему с позиций многолетнего опыта создания систем на C++, некоторые вещи я бы, конечно, сделал по-другому. Но приходилось работать с тем, что есть, и благодаря поддержке людей, окружавших нас тогда на работе и дома, удалось внедрить этот амбициозный проект в жизнь.