

**ФИО:** Сенькина Е.В., Романов Д.В., Виноградов А.И.

**Город:** Магнитогорск

**Место учебы:** Магнитогорский государственный университет им. Г.И. Носова, АВ-01-2

**Специальность:** программное обеспечение ВТиАСУ

**E-mail:** vt-pm@magtu.ru

## QNX как она есть

Е.В. Сенькина, Д.В. Романов, А.И. Виноградов

Название операционной системы QNX знакомо многим. Для одних оно прочно ассоциируется с демонстрационной дискетой, на которой умещались ядро системы, графический интерфейс пользователя и Web-браузер. Для других – со сложными системами управления промышленным и медицинским оборудованием, сетевыми маршрутизаторами, сотовыми телефонами.

Кто-то, наверное, вспомнит и кое-какие подробности: QNX – масштабируемая, встраиваемая операционная система «жесткого» реального времени. Применяется в наиболее критичных областях. Девиз фирмы QNX Software Systems – "Build a more reliable world", что значит «сделаем мир надежнее». И действительно, на протяжении всего двадцати трехлетнего существования компании ее продукты для многих являются образцом стабильности и качества.

QNX – компания с именитым и таинственным прошлым и настоящим, как, впрочем, и сама система. Таинственным – потому что продукция этой компании, операционная система QNX, использовалась и используется в самых критических и стратегических областях, таких как аэрокосмическая промышленность, энергетика, в том числе и ядерная, а также (насколько об этом можно судить) в военных разработках. Именитой же QNX можно назвать хотя бы по той простой причине, что в течение двадцати лет у этой системы реального времени нет и никогда не было серьезных конкурентов. Встроенные Linux и DOS, не говоря о Windows, даже и не приближались к лидеру в плане надежности кода и устойчивости интерфейсов.

Долгие годы QNX была системой стратегического значения, не предназначавшейся не только для экспорта, но даже и для того, чтобы быть темой открытой для общественности. Считалось, что «все, кому нужно, знают все, что им нужно». Но рынок, даже для критических систем, существует, и конкуренты не оставляют попыток занять ваше место. Поэтому, особенно по окончании холодной войны, QNX стала выходить из тени. Осуществлено встраивание операционной системы QNX в сетевое оборудование, медицинские приборы, автомобильную электронику, электронные киоски, телефонные аппараты и разнообразную бытовую технику. На сегодняшний день код и среда разработки доступны для обозрения мировому сообществу.

Официально QNX называется RTOS, Real Time Operating System (Операционная система реального времени). Операционные системы реального времени (ОС РВ) – это специальный класс программного обеспечения нижнего уровня, на базе которого разрабатываются так называемые системы реального времени (СРВ). Несмотря на то, что СРВ могут представлять собой приложения различного масштаба – от встроенного в часы электронного секретаря-ассистента до глобальной системы банковских транзакций – их общей отличительной чертой является способность дать

правильный ответ на любое разрешенное событие за время, не превосходящее некоторый, заранее определенный интервал («реальное время»).

В основе QNX, как и раньше, лежит компактное и стабильное микроядро. Поскольку система рассчитана на 100-процентно бесперебойную работу, то архитектура ядра и окружения предполагает динамическую конфигурацию – установка и удаление драйверов, сетевых протоколов, файловых систем, оконных интерфейсов и т. п. происходит в режиме "hot plug", без прекращения работы. Все перечисленные компоненты работают в отдельных адресных пространствах, без типичных для других систем «оптимизаций» по занесению подключаемых компонент в монолитное или составное ядро. Как следствие – QNX умеет восстанавливать себя практически после любого программного сбоя, любой компонент может быть «признан недействительным» и повторно запущен. Существуют точки восстановления, «неразрушающей перезагрузки», после которой система блокирует участки кода, которые вызвали сбой, и продолжает работу без потери состояния, очистки памяти и инициализации процессора.

Что включает в себя микроядро? Предельно ограниченный набор функций – девизом разработчиков было «что мы еще можем оттуда убрать», а не «что мы еще можем туда вставить». Оставшиеся сервисы: управление потоками выполнения, взаимодействие потоков (IPC), примитивы синхронизации. Все остальное, включая драйверы, сетевые протоколы и приложения любого уровня, работает как отдельные процессы, сообщаясь с ядром с помощью синхронных сообщений.

Сообщения передаются через «шину сообщений», по которой проходят все сигналы по подключению и отключению программных компонент «на лету». Следствие из общей шины сообщений – это экстраполяция локального IPC на процессорные кластеры. Если одно ваше приложение вызывает другое через системный вызов (а иных возможностей фактически и нет), то IPC прозрачно транслируется на другой процессор – и в результате приложения общаются тоже прозрачно. Применение кластерного механизма – доступ к ресурсам другого процессора, такого как файловые системы, устройства или сетевые подключения, без включения сетевых протоколов (кроме встроенного в ядро протокола Qnet). На основе такого доступа можно строить экстремально «тонкие» клиенты, все ресурсы получающие по сети. Протокол Qnet будет осуществлять маршрутизацию и баланс сетевого трафика – прозрачно для приложений. Сам Qnet работает практически через любую среду, например Ethernet или Internet.

Помимо процессорных кластеров, QNX поддерживает также и SMP для отдельных потоков – естественно, что приложение должно быть спроектировано для многопоточной среды, чтобы получить выгоду от работы на нескольких процессорах. Использование SMP можно комбинировать с Qnet для построения кластеров как одно-, так и многопроцессорных систем – перекомпиляции приложений не потребуется.

Перечисленные выше возможности QNX – это, так основные для системы характеристики. Но, вместе с тем, разработчикам стремились сделать QNX популярной и коммерчески выгодной.

Для этого был создан приятный и самодостаточный интерфейс пользователя – Photon MicroGUI. Он предоставляет все основные примитивы графической оконной системы: окна, кнопки, меню и так далее.

Первоначально QNX 6 задумывалась как POSIX-совместимая система – POSIX API был реализован на самом раннем этапе, так что не существует «библиотеки POSIX-совместимости». Если вы привыкли к программированию в Unix или Linux, можете продолжать и под QNX, проблем не возникнет.

Известно несколько попыток создать «открытую» среду разработки.

Одним из самых перспективных в сфере разработки ПО был и остается Eclipse – IDE-среда разработки, созданная на базе оболочки Websphere Studio Workbench компании IBM. Eclipse написан на Java – таким образом, проблема переносимости и универсальности этой среды решена изначально. Модульная архитектура Eclipse позволяет разработчикам и потребителям этой технологии свободно модифицировать, встраивать и добавлять возможности без перекомпиляции всего кода, адаптируя возможности Eclipse своим требованиям.

Правильным решением, с точки зрения разработчиков, стало использование Eclipse в качестве новой среды для разработки под QNX (QNX Momentics). Eclipse – открытая платформа для интеграции инструментария, поддерживаемая большим – и постоянно расширяющимся – сообществом компаний-производителей. Чтобы инструменты могли работать вместе, Eclipse предоставляет им четко определенные интерфейсы, как следствие, весь инструментарий QNX Momentics выдержан в одном стиле. Таким образом, нет необходимости изучать пользовательский интерфейс каждого отдельного инструмента. Eclipse также предоставляет расширяемую архитектуру подключаемых модулей, позволяющую QNX Momentics работать практически с любым типом информационного содержания. Например, в состав QNX Momentics входит множество подключаемых модулей для разработки и анализа встраиваемых образов, исходных текстов на C/C++ и прочих объектов, характерных для встраиваемых систем. Можно разрабатывать свои собственные подключаемые модули для работы с любыми другими объектами или использовать готовые модули.

Особое отличие среды Momentics (как и Eclipse или Sun ONE Studio) – это немодальная отладка. То есть вы можете выполнять один или несколько фрагментов кода, причем они могут выполняться на разных «целевых» платформах и быть написанными на разных языках. Дополнительные опции – подключение к уже запущенному процессу и анализ дампа «свалившегося» процесса. Естественно, поддерживаются все типы отладки: точки останова, пошаговое прохождение, просмотр стека и переменных.

Визуальные возможности – список "ToDo", формируемый на основе возникших ошибок компиляции, подсказки при наведении курсора на переменные, сохранение рабочей области и т.д. Кроме того, в наличии все средства редактирования: раскраска, автоматические отступы, отображение структуры проекта (файлов, классов, модулей) и т.д.

Одним из самых критических приложений для разработчика является система управления базами данных, RDBS. QNX поддерживает RDBMS Empress (Empress Software). Эта RDBMS изначально создавалась для QNX – и, следовательно, была оптимизирована для встроенных приложений и ограниченных ресурсов. Тем не менее, система поддерживает методы доступа SQL и ODBC.

Использование в настоящее время QNX в качестве ОС для организации АСУ и АСУТП крупных предприятий требует подготовки квалифицированных кадров, знакомых с устройством и принципами работы системы. Использование QNX в учебном процессе, помимо выполнения задачи подготовки кадров, выгодно еще и тем, что обучающиеся получают представление об иной парадигме построения операционных систем, знакомятся с архитектурой ОС, принципиально отличной от большинства ОС для PC-совместимых платформ.

В качестве заключения отметим, что ОС PV QNX имеет уже длительный путь развития. Он характеризуется постоянным улучшением качеств системы реального времени (производительность и предсказуемость), интенсивным добавлением функциональности (мультимедийные средства, протоколы Интернет и т.д.), а также стремлением ко все более полной реализации прикладных программных интерфейсов стандарта POSIX. Последнее свойство позволяет разработчику прикладной системы проще переносить программы из одной операционной системы в другую. В частности, тем, кто знаком с ОС Linux, легко перенести свой код в ОС PV QNX.

Кроме того, изменив схему лицензирования своей последней версии QNX 6.x, и предоставив бесплатно полноценный набор инструментальных средств и документации для платформы реального времени QNX Realtime Platform (QNX RTP), компания QNX Software System кардинальным образом расширила круг разработчиков, приверженных QNX. Теперь даже небольшие венчурные компании, имеющие интересные идеи для встроенных систем реального времени, могут попробовать создать действующий макет задуманного устройства с самыми минимальными затратами. Результатом будет быстрый выход на рынок большого количества новых устройств для различных областей человеческой деятельности. QNX уже широко применяется в таких областях как:

- 1) автоматизация процессов (АСУ ТП),
- 2) бытовая электроника,
- 3) медицинские приборы,
- 4) сетевые компьютерные устройства, периферия,
- 5) POS-терминалы для торговли,

- 6) научные исследования,
- 7) системы безопасности,
- 8) космонавтика,
- 9) телекоммуникации,
- 10) транспорт.

Ожидается, что в ближайшее время развитие встроенных компьютерных систем в автомобилестроении, персональной коммуникации и системах глобального мониторинга приобретет взрывной характер. Есть все основания полагать, что ОС РВ QNX сможет захватить около четверти рынка таких систем.

Итак, учитывая надежность, совместимость на уровне POSIX с другими операционными системами, мощную среду разработки, возможность построения на базе QNX системы РВ любой сложности, использование практически во всех областях промышленности, можно уверенно утверждать лидерство ОС РВ QNX на рынке ОС РВ и систем РВ, как проектов построенных на базе ОС РВ, и надеяться на ее дальнейшее внедрение во все новые области человеческой деятельности.