



QNX SOFTWARE SYSTEMS LTD.



ОС реального времени QNX[®] NEUTRINO[®] ОСРВ НА ОСНОВЕ МИКРОЯДРА

НАДЕЖНОСТЬ, ПРОВЕРЕННАЯ ВРЕМЕНЕМ.

**МАСШТАБИРУЕМОСТЬ, ОТКРЫВАЮЩАЯ
НОВЫЕ ВОЗМОЖНОСТИ.**

**ПРОИЗВОДИТЕЛЬНОСТЬ,
НЕ ИМЕЮЩАЯ АНАЛОГОВ.**

QNX Neutrino способна управлять любыми системами — от крошечных встраиваемых устройств до мощных SMP-архитектур и огромных распределенных вычислительных сред.

В ТЕЧЕНИЕ БОЛЕЕ 20 ЛЕТ
ТЕХНОЛОГИЯ QNX ПОМОГАЛА
РАЗРАБОТЧИКАМ СОЗДАВАТЬ
САМЫЕ НАДЕЖНЫЕ В МИРЕ
ПРИЛОЖЕНИЯ –
ОТ МЕДИЦИНСКИХ ПРИБОРОВ
ЖИЗНЕОБЕСПЕЧЕНИЯ И
АВТОМОБИЛЬНЫХ
ТЕЛЕМАТИЧЕСКИХ СИСТЕМ
ДО РАСПРЕДЕЛЕННЫХ
ПРОМЫШЛЕННЫХ СИСТЕМ
УПРАВЛЕНИЯ. ТЕПЕРЬ САМАЯ
НАДЕЖНАЯ ОС РЕАЛЬНОГО
ВРЕМЕНИ ПОДДЕРЖИВАЕТ ЕЩЕ
И САМЫЙ ПРОГРЕССИВНЫЙ
ИНСТРУМЕНТАРИЙ.

СОДЕРЖАНИЕ

АРХИТЕКТУРА НА ОСНОВЕ МИКРОЯДРА.....	2
ОБМЕН СООБЩЕНИЯМИ	3
ПОДДЕРЖКА POSIX	3
РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ.....	4
СИММЕТРИЧНЫЕ МУЛЬТИПРОЦЕССОРНЫЕ АРХИТЕКТУРЫ (SMP).....	5
ГРАФИЧЕСКАЯ ОБОЛОЧКА QNX PHOTON MICROGUI.....	6
ФАЙЛОВЫЕ СИСТЕМЫ.....	7
ПОДДЕРЖКА СЕТЕЙ TCP/IP.....	7
ДРАЙВЕРЫ УСТРОЙСТВ.....	8
ДИАГНОСТИЧЕСКАЯ ВЕРСИЯ МИКРОЯДРА.....	9
АДМИНИСТРАТОР СИСТЕМ ВЫСОКОЙ ГОТОВНОСТИ.....	10
СРЕДА ИСПОЛНЕНИЯ JAVA.....	10
ПРОИЗВОДИТЕЛЬНОСТЬ РЕАЛЬНОГО ВРЕМЕНИ.....	11
ПОДДЕРЖИВАЕМЫЕ ЦЕЛЕВЫЕ ПРОЦЕССОРЫ.....	11
КОМПЛЕКТ РАЗРАБОТЧИКА QNX MOMENTICS.....	12
ПРОФЕССИОНАЛЬНЫЕ СЕРВИСЫ.....	13
КРАТКИЙ ОБЗОР QNX NEUTRINO.....	14



ОСРВ QNX Neutrino: истинная мощь микроядерной технологии

В течение более 20 лет разработчики использовали технологию QNX для построения систем исключительной надежности и производительности: медицинских приборов, маршрутизаторов последнего поколения, call-центров 911, автомобильных телематических устройств, комплексов промышленной автоматизации - и даже систем управления для Международной Космической Станции. И вне зависимости от их масштаба и сложности, эти системы объединяет одно: все они работают непрерывно по 24 часа в день, 365 дней в году, год за годом.

Как QNX Neutrino делает это возможным? Ответ прост - это ОС, четко следующая принципам микроядерной архитектуры. В QNX Neutrino ядро отвечает только за базовые примитивы ОС (сигналы, таймеры, планирование, и т.п.). Все остальные компоненты системы: драйверы, файловые системы, стеки протоколов, пользовательские приложения - выполняются вне пределов ядра как отдельные процессы, каждый в своем защищенном адресном пространстве. Такая схема изначально обладает исключительной, "встроенной", отказоустойчивостью.

Однако, и это еще не все. Все компоненты QNX Neutrino используют для общения друг с другом единый, четко детерминированный механизм - обмен сообщениями. Он образует между компонентами системы виртуальную "программную шину", позволяющую подключать к ней или, наоборот, отключать любой компонент "на лету". Мало того, сообщения могут свободно передаваться между узлами вычислительной сети, предоставляя прозрачный доступ к любому ресурсу, где бы он ни находился.

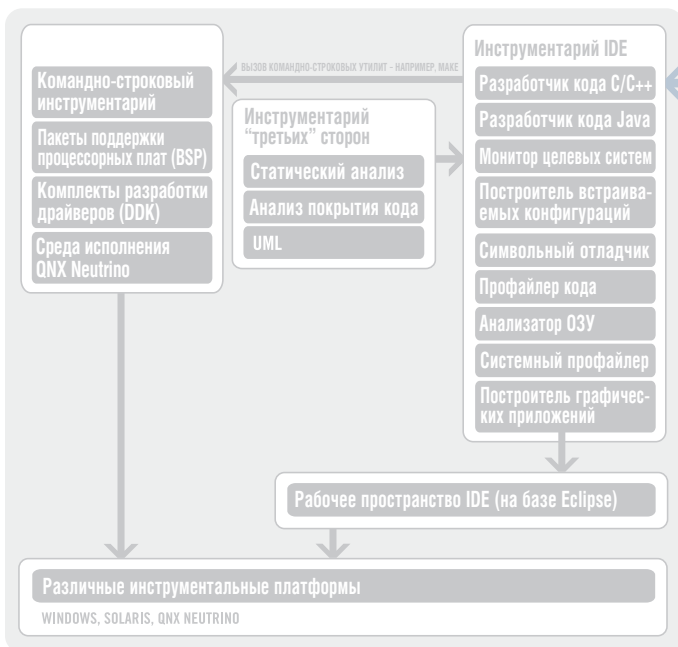
С QNX Neutrino вы можете:

Создавать системы, способные к самовосстановлению – в QNX Neutrino любой компонент в случае отказа может быть перезапущен динамически, не нарушая работу микроядра и других компонентов. Например, если драйвер попытается обратиться к памяти за пределами своего адресного пространства (что для большинства ОС является фатальной ошибкой), QNX Neutrino корректно завершит этот драйвер и освободит все занятые им ресурсы. Вы сможете даже автоматически перезапустить этот драйвер, используя администратор систем высокой готовности QNX Neutrino.

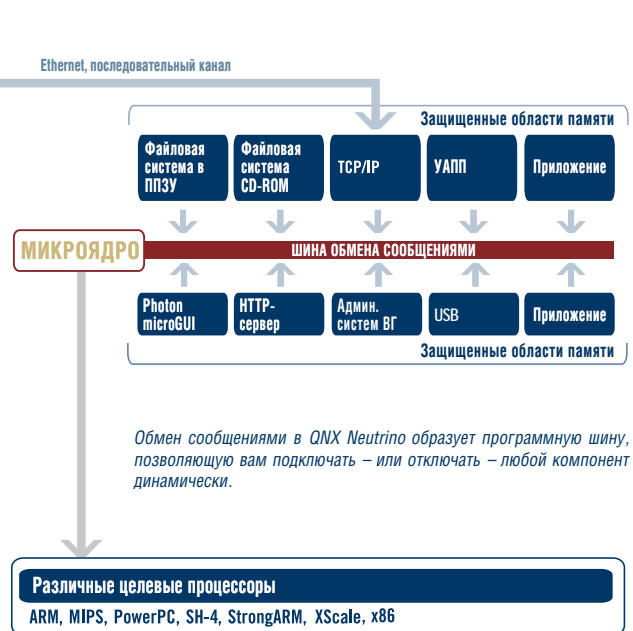
Использовать одну и ту же ОС во всей своей линейке продуктов – благодаря исключительной модульности QNX Neutrino, любые уже испытанные и проверенные компоненты - драйверы, приложения, дополнительные сервисы ОС - вы можете использовать повторно в других своих продуктах. Фактически, вы можете создать универсальный набор бинарных модулей, а затем применять его либо в однопроцессорном устройстве, либо в SMP-системе, либо в вычислительном кластере. Вне зависимости от масштаба и сложности вашей системы, вы будете использовать одну и ту же ОС, один и тот же интерфейс прикладного программирования (API) и один и тот же инструментарий разработчика.

Производить апгрейды "на лету" – поскольку любой компонент в QNX Neutrino может быть добавлен или удален динамически, ваша система может продолжать работать даже в процессе замены или добавления в нее новых приложений, драйверов или стеков протоколов.

ИНТЕГРИРОВАННЫЙ КОМПЛЕКТ РАЗРАБОТЧИКА QNX MOMENTICS



ОС РЕАЛЬНОГО ВРЕМЕНИ QNX NEUTRINO



ОБМЕН СООБЩЕНИЯМИ – ОСНОВА ГИБКОСТИ

В QNX Neutrino обмен сообщениями – это гораздо больше, чем просто форма межзадачного взаимодействия. Это один из фундаментальных механизмов, делающих системы на основе QNX Neutrino столь гибкими и простыми в разработке. Этот механизм, в частности:

- автоматически синхронизирует выполнение взаимодействующих компонентов;
- избавляет вас от необходимости следить за очередностью доставки данных;
- позволяет вам разбить сложное приложение на четко разграниченные функциональные блоки, которые можно разрабатывать и тестировать по отдельности;
- придает системам изящность, делающую их легкими в эксплуатации и обслуживании;
- действует по всей сети, предоставляя вашим приложениям прозрачный доступ к сервисам и ресурсам удаленных узлов.

Обмен сообщениями в QNX также чрезвычайно эффективен, поскольку каждая операция происходит непосредственно между отправителем и получателем. Соответственно, не происходит никакого промежуточного копирования данных, и не требуется никаких дополнительных действий по синхронизации.

Мало того, нет никакой необходимости реализовывать дополнительные уровни для обработки сложных сообщений. Чтобы обмениваться сообщениями с системными сервисами, приложения могут использовать стандартные вызовы POSIX.

Как это работает? В QNX Neutrino каждая программа, предоставляющая некий сервис (например, драйвер), может зарегистрировать в пространстве имен путей "файл" или "каталог". Потом любое приложение может соединиться с этим драйвером, применив к данному файлу или каталогу стандартную операцию *open()*. Результатом будет обычный файловый дескриптор, через который приложение сможет обращаться к сервисам драйвера при помощи вызовов POSIX, предназначенных для работы с файловыми дескрипторами – *read()*, *write()*, *lseek()* и т.п. Библиотека языка Си автоматически преобразует эти вызовы в соответствующие сообщения и передает их драйверу. Например, когда приложение вызывает функцию *read()*, чтобы считать готовые данные, библиотека преобразует этот вызов в сообщение "запрос на чтение". Реально, в QNX Neutrino приложения используют обмен сообщениями каждый раз, когда работают с файловыми дескрипторами или указателями на файлы.

Такой подход позволяет вам:

Упростить обслуживание систем – поскольку пространство имен путей четко отделяет сервисы от клиентских приложений, обновление для ваших систем становится элементарной задачей. Любой сервис можно заменить на его новую версию, в том числе в процессе эксплуатации – и клиенты сами автоматически найдут его.

Расширять ОС для нестандартных задач – QNX Neutrino предоставляет разработчикам *библиотеку администратора ресурсов*, которая позволяет сервисным программам регистрировать свои имена в пространстве имен путей и обрабатывать запросы от клиентских приложений.

В дополнение ко всему, в QNX Neutrino любые системные сервисы, включая драйверы, являются программами пользовательского кольца, а значит, разрабатываются точно так же, как и любые другие приложения. В результате вы получаете возможность легко расширять ОС совершенно новыми, специфичными для ваших приложений возможностями.

Обратите внимание, что обмен сообщениями можно использовать напрямую, при помощи функций *MsgSend()*, *MsgReceive()* и *MsgReply()*. В QNX Neutrino доступны также и традиционные формы межзадачного взаимодействия – программные каналы, FIFO, очереди сообщений POSIX, разделяемая память и сигналы.

АРХИТЕКТУРА НА ОСНОВЕ МИКРОЯДРА: КЛЮЧ К МОДУЛЬНОЙ СТРУКТУРЕ ПРИЛОЖЕНИЙ



РАСШИРЯЕМОСТЬ "ПО ОПРЕДЕЛЕНИЮ"

Некоторые ОС называют микроядерными за их небольшую ресурсоемкость. Такое определение, строго говоря, не совсем корректно – архитектура на основе микроядра подразумевает совершенно другой подход к самой структуре ОС. Этот подход дает не только исключительную надежность, но и гораздо большую гибкость, позволяющую удовлетворять требованиям самых различных областей применения.

В ОС на основе микроядра драйверы, стеки протоколов и прочие системные сервисы выполняются вне пределов ядра как опциональные пользовательские процессы. В результате, расширить ОС дополнительным сервисом, специфичным для данного приложения или данной индустрии, не составляет никакого труда – просто напишите свой собственный сервис, используя обычный инструментарий разработки приложений. В отличие от других ОС, вам не придется изучать специализированные средства или нанимать дорогостоящих специалистов для программирования на уровне ядра.

Поскольку дополнительные сервисы ОС выполняются в защищенных адресных пространствах как пользовательские процессы, они не могут нарушить работу микроядра. Таким образом, вы можете расширять QNX Neutrino, одновременно сохраняя основу ее надежности. Фактически, поскольку микроядро не требует внесения никаких изменений, вы можете всегда быть уверены, что используете именно тот бинарный модуль микроядра, который был проверен в лабораториях QNX Software Systems и на многих объектах, где работает QNX Neutrino. Никакая другая ОС не предоставляет такого мощного сочетания расширяемости и надежности.

"ВСТРОЕННАЯ" ПОДДЕРЖКА POSIX

Многие операционные системы реального времени, претендующие на POSIX-совместимость, часто на самом деле поддерживают только его небольшие подмножества. QNX Neutrino, в отличие от них, была изначально задумана как POSIX-совместимая ОСРВ - в ней реализована "встроенная" поддержка POSIX. Этот подход исключает необходимость в сложных промежуточных уровнях, позволяя вам достигнуть лучшей производительности и сэкономить на объеме памяти.

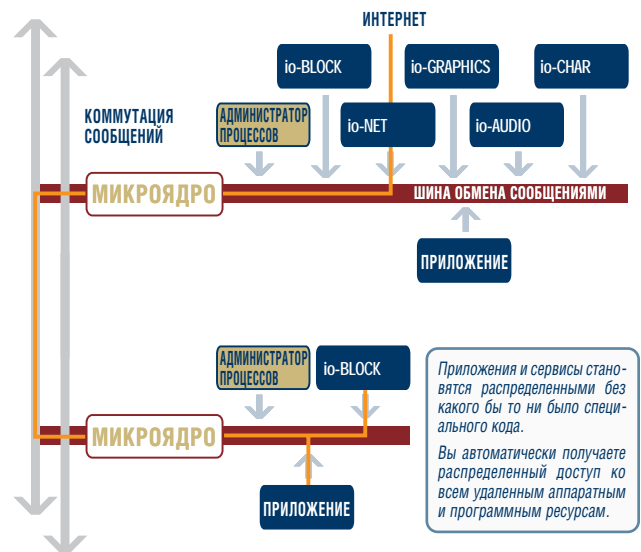
QNX Neutrino соответствует POSIX 1003.1-2001, включая многопоточность, расширения реального времени и ряд других опций (см. ниже). В результате, в QNX Neutrino очень просто переносить программы с открытым исходным текстом из, например, UNIX или Linux - в большинстве случаев перенос сводится к перекомпиляции и компоновке с библиотеками QNX Neutrino. Более того, опыт показывает, что программисты с опытом работы в UNIX или Linux осваиваются в QNX Neutrino практически мгновенно.

Список поддерживаемых примитивов POSIX 1003.1-2001 включает в себя:

- BAR:** Барьеры
- CS:** Выбор часов реального времени
- CX:** Расширение ISO C
- FSC:** Синхронизация файлов
- IP6:** IPv6
- MON:** Монотонные часы
- MPT:** Защита памяти
- PIO:** Приоритетный ввод/вывод
- PS:** Планирование процессов
- RTS:** Сигналы реального времени
- SEM:** Семафоры
- SHM:** Объекты разделяемой памяти
- SIO:** Синхронизированный ввод/вывод
- SPI:** Спин-блокировки
- SS:** Спорадическое планирование процессов
- THR:** Потoki
- TMO:** Тайм-ауты
- TMR:** Таймеры
- TPI:** Наследование приоритетов потоков
- TPP:** Защита приоритетов потоков
- TPS:** Планирование выполнения потоков
- TSA:** Адрес стека потока
- TSF:** Функции, совместимые с многопоточностью
- TSH:** Синхронизация потоков в процессе
- TSP:** Спорадическое планирование потоков
- TSS:** Размер адреса стека потока

РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ

Поддержка распределенных вычислений в QNX Neutrino была заложена изначально. Таким образом, чтобы обращаться к сервисам на удаленных узлах, вашему приложению не нужно выполнять каких-либо специальных действий. Оно будет посылать такие же сообщения, которые использовались бы для доступа к локальному сервису, и эти сообщения будут автоматически перенаправлены по сети на нужный узел. Что это дает? Любые удаленные ресурсы – диски, сетевые адаптеры, стеки протоколов, и т.п. – становятся доступны так, как будто они находятся на локальной машине.



Сообщения в QNX Neutrino могут свободно передаваться между узлами вычислительной сети, предоставляя прозрачный доступ к любому ресурсу, где бы он ни находился.

Используя распределенные возможности QNX Neutrino, вы можете:

Уменьшить затраты на оборудование – при использовании распределенных вычислений узлы сети могут совместно использовать ресурсы вместо их дублирования. Например, если на одном узле расположена большая файловая система в ПЗУ, другим узлам иметь такую же не обязательно – они смогут использовать файловую систему того узла, на котором она уже есть. Аналогично, если на одном узле запущен стек TCP/IP, все остальные узлы смогут использовать этот узел как TCP/IP-шлюз, исключая необходимость в настройке нескольких IP-адресов.

Добавлять вычислительную мощность без дополнительных разработок – чтобы добавить в распоряжение вашего приложения больше вычислительных мощностей или больше физических интерфейсов, достаточно просто вставить дополнительную процессорную карту или добавить в сеть еще один компьютер. Приложения на уже имеющихся узлах смогут пользоваться ресурсами нового узла без внесения в них каких-либо изменений.

Упростить проектирование отказоустойчивых кластеров – поскольку обмен сообщениями в QNX Neutrino предоставляет прозрачный доступ к сервисам вне зависимости от их местоположения, приложения могут полностью абстрагироваться от принятия решений о том, кто будет обрабатывать запрос от клиента, где этот сервис расположен, и есть ли другие сервисы, способные обработать этот запрос (например, в случае дублирования сервиса на нескольких узлах для обеспечения отказоустойчивости или балансировки нагрузки).

Увеличить пропускную способность сети резервированными соединениями – в QNX Neutrino сообщения могут передаваться по нескольким соединениям одновременно, увеличивая пропускную способность и повышая надежность связи. Например, при отказе одного из соединений QNX Neutrino может перенаправить поток данных по одному или нескольким альтернативным маршрутам. Вы также настроите QNX Neutrino на балансировку сетевого трафика между всеми доступными соединениями, повысив тем самым суммарную пропускную способность.

Использовать любой транспорт – поскольку механизм распределенных вычислений QNX Neutrino расположен над транспортным уровнем, он одинаково хорошо работает через локальные сети, объединительные панели, коммутируемые сети (switch fabrics) и шинные интерфейсы типа CAN и MOST.

СИММЕТРИЧНЫЕ МУЛЬТИПРОЦЕССОРНЫЕ СИСТЕМЫ (SMP)

QNX Neutrino – единственная коммерческая ОС реального времени, в полной мере поддерживающая SMP: любой поток любого процесса в ней можно запланировать на выполнение на любом из доступных процессоров. С QNX Neutrino вам не придется "учить" ваше приложение,

драйвер или стек протоколов работать с SMP – если ваш процесс является многопоточным, его потоки будут автоматически распланированы по процессорам SMP-платы.

С QNX Neutrino вы можете:

Использовать наиболее подходящие SMP-платы – на ваш выбор широкий спектр плат на базе процессоров MIPS, PowerPC и x86.

Настраивать производительность, используя родственность процессоров – для оптимизации использования процессорного кэша, QNX Neutrino будет всегда пытаться запланировать поток на процессоре, где он выполнялся в последний раз, если это допустимо. Чтобы иметь возможность дополнительно оптимизировать работу кэша, микроядро QNX Neutrino предоставляет "маску родственности" процессоров, позволяя закрепить поток за одним или несколькими выбранными процессорами.

Использовать максимум возможностей каждого процессора – поскольку QNX Neutrino может запланировать любой поток на любом процессоре, все процессоры можно использовать по максимуму, получая наибольший выигрыш в производительности. При этом, в отличие от традиционных ядер ОС, чтобы поддерживать SMP, микроядру QNX Neutrino не требуется множества модификаций кода, снижающих производительность. Микроядро с поддержкой SMP всего на несколько килобайт больше стандартного.

Строить отказоустойчивые кластеры, обладающие колоссальной вычислительной мощностью – комбинируя SMP-возможности QNX Neutrino с ее встроенным механизмом распределенных вычислений, вы можете легко конструировать массивные отказоустойчивые кластеры, включающие в себя сотни одно- и многопроцессорных систем. Используя QNX Neutrino, вы получаете уникальную возможность выполнять один и тот же набор бинарных модулей вашего приложения как на однопроцессорных, так и на SMP- и кластерных целевых системах.

ВСЯ МОЩЬ НАСТОЯЩЕГО SMP

SMP-система состоит из двух или более тесно связанных процессоров, разделяющих общие ресурсы – например, память и шины. В настоящей SMP-системе любой процессор может выполнять любой поток, включая код ядра, код приложения или код обработчика прерывания. Симметрия, обеспечивающая максимальную загрузку процессоров, достигается через использование одинаковых процессоров и связей между ними, а также с помощью специально спроектированного ядра ОС. Предлагая поддержку SMP, QNX Neutrino предлагает огромный выигрыш в производительности для сетевых устройств, современных систем обработки изображений, многопоточных серверов и многих других приложений, требующих больших вычислительных мощностей.

SMP для QNX Neutrino позволяет в полной мере пользоваться параллельной архитектурой, позволяя множеству приложений и системных сервисов (драйверов, стеков протоколов и т.п.) выполняться одновременно.



QNX PHOTON MICROGUI

Только в QNX Neutrino существует Photon microGUI®, модульная графическая оболочка, способная обеспечить даже самому маленькому встраиваемому устройству профессиональный, современный графический интерфейс. По аналогии с самой QNX Neutrino, Photon основан на компактном микроядре и предоставляет большинство своих сервисов посредством опциональных процессов, работающих в защищенных областях памяти. Результат – отказоустойчивая и динамически масштабируемая графическая среда.

В отличие от функционально ограниченных графических библиотек, традиционных для других ОСРВ, Photon представляет собой настоящую оконную систему, позволяющую вам:

Строить сложные многослойные дисплеи – Photon предоставляет высокотехнологичные функции типа внеэкранный растеризации (off-screen rendering), перекрытия изображений, канала прозрачности (alpha-blending), подстановки текстуры и прямого режима. В результате, вы можете создавать плавно работающие многослойные дисплеи, способные отображать комбинацию из графики и живого видео – идеальное решение для систем динамической навигации, телеприставок с функцией "картинка в картинке", и т.п.

Создавать уникальный облик – используя механизм, называемый *стилями виджетов (widget styles)*, вы можете настраивать внешний вид кнопок, меню, окон и других элементов интерфейса – как индивидуально, так и на глобальном уровне. Для полного контроля над пользовательским интерфейсом вашей системы вы можете даже менять стили "на ходу", в процессе выполнения.

Отображать и вводить текст на нескольких языках одновременно – Photon поддерживает стандарт Unicode, благодаря чему пользователи могут, например, вводить китайские, японские или корейские символы, в то время как система отображает текст на английском или любом другом языке.

Отображать высококачественные шрифты на дисплеях любого размера – администратор шрифтов Photon поддерживает множество шрифтовых форматов, включая растровые и TrueType, а также компактные штриховые шрифты для азиатских языков. Вы можете масштабировать шрифты до любого размера и отображать их с использованием технологии сглаживания, добиваясь четко читаемых изображений на ЖКИ и других устройствах с низким разрешением.

Подключать какие угодно мультимедийные форматы – расширяемая архитектура медиапроигрывателя Photon предоставляет готовую поддержку множества форматов данных, включая CD-аудио, MP3, потоки MPEG-1, WAV, AIFF, IFF, AU и прочие. Вам нужно добавить поддержку нового формата? Никаких проблем. Используя систему плагинов и высокоуровневый API, вы легко сможете добавить в систему ваши собственные мультимедийные компоненты.

Выбрать наиболее подходящий веб-браузер – Photon поддерживает Mozilla, мощный веб-браузер с открытым исходным текстом, основанный на технологии Netscape 7, а также браузеры от "третьих" производителей.

Обновлять графический интерфейс "на лету" – в Photon большинство графических сервисов (видеодрайверы, менеджеры окон, драйверы устройств ввода и т.п.) реализованы в виде плагинов. Таким образом, вы можете динамически добавлять, заменять или обновлять практически любой компонент вашего интерфейса – перезагрузка для этого не потребуется.

Создавать полноценные интерфейсы, не написав ни единой строчки кода – для упрощения разработки графических интерфейсов, Photon поддерживает PhAB™, визуальное средство разработки, позволяющее вам создавать интерфейсы целиком по принципу "укажи и нажми". Более подробно о PhAB вы можете прочитать в описании комплекта разработчика QNX Momentics®.

> PHAB: PHOTON APPLICATION BUILDER

При помощи PhAB приложения для Photon microGUI создаются практически моментально. Просто укажите и щелкните "мышью" для добавления или изменения свойств кнопок, окон, меню, индикаторов и других компонентов интерфейса.

Переведите ваш интерфейс на несколько языков с полной поддержкой Unicode.

Подключите вызовы функций для автоматического открытия окон, диалогов и меню – не нужно писать никакого кода для "склеивания" компонентов.



Используйте встроенные редакторы ресурсов для быстрого изменения облика и поведения виджетов.

Щелкните, чтобы добавить любой виджет в ваше приложение.

ФАЙЛОВЫЕ СИСТЕМЫ

В традиционных ОС файловые системы встроены в ядро. В QNX Neutrino файловые системы расположены вне пределов ядра и выполняются в отдельных защищенных областях памяти как пользовательские процессы. В результате, вы можете запустить, остановить или обновить поддержку той или иной файловой системы "на лету", без необходимости в перезагрузке.

В дополнение, несколько файловых систем: дисковая, встраиваемая в ППЗУ, CD-ROM, CIFS и т.д. – могут выполняться одновременно на одной и той же целевой системе. Они даже могут работать совместно, расширяя возможности друг друга. Например, файловая система со сжатием может работать совместно со встраиваемой файловой системой, существенно снижая потребности вашего устройства в объеме ППЗУ.

ОБЗОР ФАЙЛОВЫХ СИСТЕМ

Встраиваемые	Дисковые	Специальные	Сетевые
Образная ROM/Flash Execute-in-place	QNX POSIX	Со сжатием Разворачивание на "лету"	NFS Совместимость с Unix
В ОЗУ Временное хранилище	Linux Ext2	Пакетная Обновления и откаты на "лету"	CIFS Совместимость с Microsoft
NOR flash Линейное flash-ППЗУ	DOS FAT 12, 16, 32		
NAND flash Страничное flash-ППЗУ	CD-ROM ISO9660, Joliet		

ВСТРАИВАЕМЫЕ ФАЙЛОВЫЕ СИСТЕМЫ

Образная – поддерживает "выполнение с места" (XIP) для уменьшения потребностей в ОЗУ и для поддержки сверхбыстрой загрузки устройств высокой готовности.

В ОЗУ – для систем, требующих быстродействующего временного хранилища данных с доступом по чтению и записи.

В линейном flash-ППЗУ – предоставляет автоматическое восстановление после сбоев, фоновое стирание и освобождение, а также полную вытесняемость для обеспечения производительности в реальном времени. Все операции записи упорядочиваются, чтобы гарантировать целостность данных.

В страничном flash-ППЗУ – позволяет стандартным файловым системам (QNX, Linux или DOS) выполняться непосредственно поверх "сырых" разделов ППЗУ. Поддерживает динамическое распределение и освобождение блоков, с полной коррекцией ошибок на основе ECC для продления срока жизни.

ПРИМЕЧАНИЕ: файловые системы как в линейном, так и в страничном flash-ППЗУ являются полнофункциональными, с поддержкой иерархии каталогов, символьных связей и прав доступа в стандарте POSIX.

ДИСКОВЫЕ ФАЙЛОВЫЕ СИСТЕМЫ

Все нижеперечисленные файловые системы поддерживают IDE- и SCSI-диски больших размеров:

QNX – полностью реализует семантику POSIX. Подходит для инструментальных рабочих станций и для встраиваемых целевых систем, использующих диски или страничное flash-ППЗУ.

Linux и DOS – для прозрачного доступа к другим разделам жесткого диска, а также для обмена данными с другими машинами посредством сменных носителей.

CD-ROM / DVD – поддерживает все соответствующие стандарты, включая ISO9660, Joliet и Rock Ridge.

СПЕЦИАЛЬНЫЕ ФАЙЛОВЫЕ СИСТЕМЫ

Используйте эти файловые системы для расширения возможностей других файловых систем:

Со сжатием – позволяет разворачивать "на лету" сильно сжатые файлы. Плотность сжатия колеблется от 35% до 55% для исполняемых файлов и до 95% для текста. Используйте эту файловую систему:

- для уменьшения расхода flash-памяти;
- снижения загрузки сети при доступе к удаленным файлам через NFS или CIFS.

Пакетная – использует определение пакета на основе XML для быстрой установки и деинсталляции программного обеспечения.

СЕТЕВЫЕ ФАЙЛОВЫЕ СИСТЕМЫ

Используйте эти файловые системы для организации прозрачного доступа к файлам на стороне Windows, Unix и Linux. Поддерживаются как CIFS, так и NFS.

ПОДДЕРЖКА СЕТЕЙ TCP/IP

В QNX Neutrino все сетевые сервисы выполняются вне пределов ядра как отдельные процессы с защищенными адресными пространствами, поэтому вы можете запускать, останавливать или обновлять любой драйвер или протокол "на лету". Сверх того, ваше приложение может пользоваться любым сетевым сервисом через один и тот же стандартный POSIX API, применяемый для работы со всеми остальными сервисами. Вы также можете:

- сочетать любое количество сетевых протоколов, включая TCP/IP и распределенную сеть QNX;
- создавать многочисленные виртуальные сети, запуская несколько копий стека TCP/IP на одном и том же физическом интерфейсе;
- использовать богатую базу сетевого кода "третьих" производителей, основанного на POSIX и BSD API.

ВЫБЕРИТЕ НАИБОЛЕЕ ПОДХОДЯЩИЙ СТЕК ПРОТОКОЛОВ

QNX Neutrino предлагает несколько вариантов стека TCP/IP:

Стек NetBSD TCP/IP – поддерживает маршрутизацию IP, сокеты с управлением маршрутизацией, широковещательные и многоабонентские передачи, ARP, ICMP и IGMP, а также CIFS, DHCP, DNS, NFS, PPP, PPPoE, UDP, а также включает в себя встраиваемый веб-сервер.

Усовершенствованный стек NetBSD TCP/IP с поддержкой IPSec и IPv6 – включает в себя все возможности стандартного стека, плюс функции, предназначенные для мобильных и безопасных коммуникаций нового поколения. Предоставляет полную поддержку IPv6 и IPSec за счет расширений KAME, а также поддержку VPN через туннели IPSec. Также содержит в себе оптимизированный код маршрутизации для увеличения производительности.

Встраиваемый стек TCP/IP для систем с ограниченным объемом памяти – несмотря на свой небольшой объем (менее 80Кб), этот стек предоставляет поддержку IP, TCP и UDP через интерфейсы Ethernet, PPP и PPPoE. Поскольку в приложениях используется интерфейс BSD-сокетов, для переключения между встраиваемым стеком и стеком NetBSD не нужно перекомпилировать код.

ПРОСТОТА РАЗРАБОТКИ ДРАЙВЕРОВ

QNX Neutrino содержит множество готовых драйверов для различных плат и периферийных устройств. Однако, если вам придется написать драйвер для вашего собственного нестандартного устройства, вы найдете эту задачу на удивление простой, благодаря библиотеке администратора ресурсов.

Эта библиотека предоставляет вам простой интерфейс для регистрации вашего драйвера в пространстве имен путей и обработки запросов от клиентов. Однако, применение этой библиотеки не ограничивается только драйверами - упрощая процессы установления и разрыва соединений, она помогает значительно ускорить разработку любого сервиса, с которым работают множество клиентских приложений. Если клиент захочет установить соединение с каким-нибудь (любым!) сервисом, ему потребуется сделать один-единственный вызов - POSIX *open()*.

С QNX Neutrino вы можете:

Использовать пакеты разработки драйверов (DDK) для ускорения разработки – для еще большего упрощения разработки драйверов в состав комплекта разработчика QNX Momentics входят пакеты разработки

драйверов (DDK) для устройств различного типа, включая аудио-, графические и сетевые адаптеры, терминальные устройства, устройства ввода, принтеры и USB-устройства. В состав пакетов входит детальная документация, исходные тексты, а также готовый программный каркас, в котором весь высокоуровневый аппаратно-независимый код уже реализован в библиотеках – таким образом, вы сможете сразу сосредоточиться на написании аппаратно-зависимого кода, специфичного для вашего устройства.



Пакеты разработки драйверов (DDK) сводят усилия по написанию драйверов к минимуму – в множестве случаев большая часть работы уже сделана.

Отлаживать драйверы на уровне исходного текста, используя обычный инструментарий – поскольку драйверы и прочие администраторы ресурсов выполняются как обычные пользовательские процессы, вы можете отлаживать их при помощи тех же интегрированных инструментов разработки приложений, которые входят в состав QNX Momentics. Вам не потребуются отладчики на уровне ядра, которые могут "завешивать" всю систему, в результате скрывая ошибки.

Тестировать новые драйверы без перезагрузки – вы можете тестировать изменения в коде ваших драйверов без перезагрузки системы, и даже не начиная новую отладочную сессию – просто перекомпилируйте и перезапустите драйвер. Мало того, поскольку все драйверы представляют собой обычные процессы, вы можете тестировать и отлаживать их прямо на инструментальном компьютере, еще до того как будет готова ваша целевая аппаратура.

Программировать многопоточные сервисы без написания больших объемов кода – чтобы сделать многопоточные драйверы проще и меньше по объему, библиотека администратора ресурсов включает в себя функции управления пулами потоков, которые автоматически регулируют число потоков в системе в зависимости от загрузки. Драйвер, использующий пулы потоков, изначально поддерживает SMP и может легко масштабироваться в многопроцессорных системах.

ДИАГНОСТИЧЕСКАЯ ВЕРСИЯ МИКРОЯДРА

ОПТИМИЗИРУЙТЕ ПОВЕДЕНИЕ ВАШЕЙ СИСТЕМЫ

Используя диагностическую версию микроядра, вы можете разрешать конфликты синхронизации, обнаруживать взаимные блокировки потоков, проследить корни семантических ошибок, выявлять скрытые дефекты программного и аппаратного обеспечения - причем как в однопроцессорных, так и в многопроцессорных системах.

Диагностическая версия микроядра может отслеживать и протоколировать каждое системное событие, включая вызовы ядра, прерывания, сообщения, смену состояний потоков и планирование. Фактически, она представляет собой логический анализатор для всей вашей системы: если что-то сработает не так, ядро поможет вам выяснить, когда произошло событие, какие программные модули были в нем задействованы, что они делали, и, самое главное, как интерпретировать это событие.

Будучи альтернативой стандартному микроядру QNX Neutrino, диагностическая версия абсолютно аналогична ему по функциональности, так что вам не придется изменять свой код. Более того, ее можно использовать в системах, находящихся в эксплуатации, поскольку она лишь немногим медленнее обычного микроядра. Влияние на производитель-

ность системы проявляется только в процессе трассировки событий, но и здесь вы можете свести это влияние к минимуму, производя динамическое протоколирование и фильтруя информацию по уровням важности.

ВИЗУАЛИЗИРУЙТЕ СИСТЕМНЫЕ СОБЫТИЯ ДЛЯ БЫСТРОЙ ЛОКАЛИЗАЦИИ ПРОБЛЕМ

Обладая способностью генерировать огромные объемы системной информации, диагностическая версия микроядра одновременно позволяет вам четко контролировать, какие события и когда записывать, таким образом, вы можете уделять внимание в точности нужным областям. А чтобы сделать задачу интерпретации данных еще проще, ядро тесно интегрировано с системным профайлером QNX Momentics, позволяя визуализировать нужные участки и проследить сложные взаимодействия.

СОЗДАВАЙТЕ СВОИ СОБСТВЕННЫЕ ФИЛЬТРЫ СОБЫТИЙ

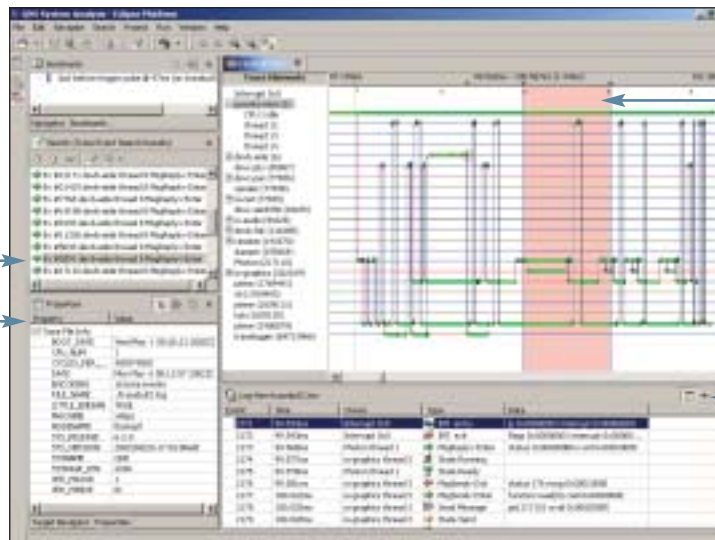
Если готовые фильтры событий, предоставляемые диагностической версией микроядра, не соответствуют ситуациям, которые вы хотели бы изучить, это не проблема: ядро поддерживает динамические определяемые пользователем фильтры для отслеживания сложных условий, характерных для конкретного приложения. Ваше приложение также может само вставлять специальные события в систему трассировки для предупреждающего влияния на процесс протоколирования.

> СИСТЕМНЫЙ ПРОФАЙЛЕР

Комбинация диагностической версии микроядра и системного профайлера поможет вам быстро обнаруживать проблемы синхронизации, семантические ошибки и прочие факторы, ухудшающие производительность.

Ищите события по типам и переходите прямо к нужному событию.

Анализируйте состояние системы в моменты, когда были записаны события.



Увеличьте нужный временной диапазон до полного экрана, выберите нужный процесс и создайте удобное представление.

Просматривайте краткую сводку деталей по событиям, включая продолжительность, владельца и тип.

АДМИНИСТРАТОР СИСТЕМ ВЫСОКОЙ ГОТОВНОСТИ

УМЕНЬШИТЕ ВРЕМЯ ВОССТАНОВЛЕНИЯ ВАШЕЙ СИСТЕМЫ

Принятый в QNX Neutrino подход к обеспечению высокой готовности очень прост: время, расходуемое на перезапуск одного компонента, гораздо меньше времени перезагрузки всей системы. Например, драйвер или стек протоколов, в котором возникла проблема, может быть немедленно выгружен и перезапущен, зачастую за единицы миллисекунд. Перезагрузка системы не требуется. Именно такой тонкий подход к изоляции сбоев позволяет QNX Neutrino обеспечивать гораздо меньшее, чем у других ОС, среднее время восстановления (MTTR).

В дополнение к этому, вы можете расширить встроенные возможности QNX Neutrino, применив администратор систем высокой готовности, "интеллектуальный сторожевой таймер", позволяющий вашей системе в случае сбоя самовосстанавливаться автоматически.

Администратор систем высокой готовности обеспечивает:

Мгновенные уведомления об отказах – в администраторе систем высокой готовности реализован механизм квитаций работоспособности, следящий за состоянием каждого компонента системы и позволяющий обнаруживать отказы на самой ранней стадии. Если администратор обнаруживает определенное стечение обстоятельств или отказ, он может автоматически мгновенно оповестить об этом другие компоненты.

Настраиваемые сценарии восстановления – используя библиотеку администратора систем высокой готовности, ваше приложение может явно указать администратору, какие действия по восстановлению и в каком порядке предпринять в случае сбоя.

Автоматическое восстановление соединений – администратор систем высокой готовности также предоставляет клиентскую библиотеку, которая позволяет вашей системе в случае отказа моментально восстановить прерванные соединения. Эта библиотека содержит замену для стандартных функций ввода/вывода из библиотеки языка Си.

"Посмертный" анализ – если процесс завершается некорректно, администратор систем высокой готовности может сохранить его образ для последующей обработки. Анализируя этот образ, вы сможете моментально определить, какая строка кода вызвала сбой, а также узнать содержимое переменных, чтобы точно определить, что именно произошло.

Администратор систем высокой готовности обладает способностью к самовосстановлению и поэтому устойчив к внутренним сбоям. Если он по какой-либо причине завершается некорректно, он моментально и полностью восстанавливает свое предыдущее состояние.

СРЕДА ИСПОЛНЕНИЯ JAVA

QNX Neutrino предоставляет два варианта среды исполнения Java. Оба они основаны на виртуальной машине J9 от IBM:

WebSphere Embedded Environment – среда исполнения Java в стандарте "Java Powered", совместимая с Java 2 Micro Edition (J2ME).

WebSphere Custom Environment – модульная, настраиваемая среда исполнения с поддержкой predefined подмножеств спецификации Java 2 Standard Edition (J2SE), а также ее произвольных подмножеств и супермножеств.

Оба варианта среды исполнения оптимизированы для QNX Neutrino, и оба предоставляют очень быструю интерпретацию байт-кода, а также компиляцию JIT и "сбор мусора".

ПРОЗРАЧНЫЙ ДОСТУП К РАСШИРЕННЫМ СЕРВИСАМ ОС

Обычно в классических OSCP Java-приложения являются "гражданами второго сорта". В QNX Neutrino, напротив, Java-приложения могут получать доступ к тем же сервисам ОС, что и обычные приложения. Например, им доступны:

Полный доступ к распределенным функциям и SMP – как и любое обычное приложение QNX Neutrino, Java-приложение может получить доступ к любому системному сервису на любом узле сети. А поскольку потоки Java реализуются как обычные потоки QNX Neutrino, многопоточное Java-приложение может быть автоматически распределено по процессорам SMP-платы. Такое распределение потоков также гарантирует, что Java-приложения будут выполняться в соответствии со своими приоритетами.

Одновременное выполнение в защищенной памяти нескольких виртуальных машин – в QNX Neutrino, каждая виртуальная машина выполняется в своем собственном защищенном адресном пространстве. Мало того, несколько виртуальных машин могут выполняться одновременно, позволяя вам запускать параллельно несколько Java-приложений, а не просто апплетов.

Полный доступ к штатной графической оболочке – в отличие от ограниченных графических библиотек, предлагаемых в классических OSCP, Photon microGUI позволяет множеству как обычных приложений, так и виртуальных машин Java, пользоваться экраном одновременно. Кроме того, Java-приложения могут выглядеть точно так же, как и стандартные приложения Photon. Вы можете написать весь интерфейс на Java, если это необходимо.

ПРОВЕРЕННАЯ ПРОИЗВОДИТЕЛЬНОСТЬ РЕАЛЬНОГО ВРЕМЕНИ

ДОБЕЙТЕСЬ ПРЕДСКАЗУЕМОГО ВРЕМЕНИ РЕАКЦИИ, ДАЖЕ В СЛОЖНЫХ СИСТЕМАХ

Будучи основанной на микроядре и используя вытесняющую приоритетную многозадачность, QNX Neutrino предоставляет одновременно быстрое и предсказуемое время реакции на событие. Таким образом, потоки с высоким приоритетом всегда выполняют свои задачи вовремя, при любом уровне загрузке системы.

QNX Neutrino предоставляет высочайшую производительность реального времени, поскольку в ней реализованы:

Сверхмалые задержки обработки прерывания и переключения контекста – с временем переключения контекста в 600 наносекунд на процессорах класса Motorola PowerPC 7450, QNX Neutrino позволяет "выжать" максимум производительности из вашей аппаратуры.

Распределенный механизм наследования приоритетов – в QNX Neutrino драйверы, файловые системы и прочие сервисы могут выполняться с приоритетом клиента, запросившего обслуживание, даже если он расположен на другом узле сети. Такое наследование приоритетов при обмене сообщениями дает гарантию, что задача, выполняемая по заказу низкоприоритетного клиента, всегда будет вытеснена задачей от высокоприоритетного клиента. Инверсия приоритетов исключается.

Свобода выбора дисциплины планирования потоков – QNX Neutrino не просто предоставляет несколько дисциплин планирования (адаптивная, FIFO, карусельная, спорадическая), она позволяет вам назначать каждому потоку свою дисциплину. Таким образом, вы не привязаны к выбору одной дисциплины для всех потоков на узле.

Гарантированная доступность процессора для задач с жестким графиком – используя спорадическое планирование, вы можете назначать лимит времени выполнения для потоков в пределах определенного интервала. В результате, эти потоки будут готовы обработать нерегулярно (асинхронно) возникающие события, не рискуя нарушить график выполнения других процессов и потоков. Эта дисциплина особенно полезна при реализации алгоритма RMA (Rate Monotonic Analysis) в системе, обрабатывающей одновременно периодические и аperiodические события.

Автоматическая синхронизация системных компонентов – синхронизация, предоставляемая механизмом обмена сообщениями в QNX Neutrino, значительно упрощает реализацию поведения системы в реальном времени. Во многих других ОС такое поведение приходится реализовывать при помощи двухуровневого планирования и с большими накладными расходами.

Вложенные прерывания – предоставляя поддержку вложенных прерываний, в сочетании с фиксированной верхней границей времени реак-

ции, QNX Neutrino позволяет вашей системе обрабатывать наиболее приоритетные прерывания первыми и за предсказуемое время.

БОГАТЫЙ ВЫБОР ЦЕЛЕВЫХ ПРОЦЕССОРОВ

ВЫБЕРИТЕ НАИБОЛЕЕ ПОДХОДЯЩИЙ ПРОЦЕССОР

Используя QNX Neutrino, вы можете реализовывать свои целевые системы на широком спектре процессоров с архитектурой ARM, MIPS, PowerPC, SH-4, StrongARM, XScale или x86. Кроме того, QNX Neutrino предоставляет функции и макросы, позволяющие писать процессорно-независимые драйверы и приложения. В ряде случаев вам достаточно будет просто указать тип процессора, чтобы автоматически сгенерировать нужный код – можно даже компилировать код для нескольких процессоров одновременно. В результате, вы можете тестировать и отлаживать ваши приложения еще до выбора целевого процессора, или использовать несколько разных процессоров для достижения оптимального соотношения цены и производительности.

Перечень поддерживаемых процессоров и процессорных плат постоянно пополняется. Посетите <http://www.qnx.com>, чтобы ознакомиться с самым свежим списком.

QNX NEUTRINO ПРИЗНАНА ЛИДЕРОМ ПО РЕЗУЛЬТАТАМ НЕЗАВИСИМОГО ИССЛЕДОВАНИЯ

В независимом исследовании четырех встраиваемых ОС, компания Dedicated Systems Experts наградила QNX Neutrino высочайшими оценками за архитектуру ОС и производительность: 9 из 10 возможных. В одном из наиболее жестких тестов, QNX Neutrino успешно обработала миллиард прерываний, генерированных с периодом в 9 микросекунд. Ни одно прерывание не было пропущено.

ОСРВ QNX Neutrino



Источник: Dedicated Systems Experts

ИНТЕГРИРОВАННЫЙ КОМПЛЕКТ РАЗРАБОТЧИКА QNX MOMENTICS

Комплект разработчика QNX Momentics™ Professional Edition поддерживает различные языки программирования, различные инструментальные ОС и различные целевые процессоры. Полнофункциональный и в то же время простой в использовании, он способен значительно ускорить разработку вашего проекта, вне зависимости от его сложности и масштаба. В состав комплекта входят:

- тесно интегрированный инструментарий для анализа памяти, профилирования кода, анализа событий трассировки, управления версиями, удаленной диагностики, генерации графических интерфейсов, и т.д.;
- мастера, позволяющие вам экономить время при создании новых проектов, оптимизации целевых образов и организации сессий удаленной отладки;
- богатый выбор пакетов поддержки процессорных плат, библиотек и исходных текстов драйверов различного типа.

Используя QNX Momentics, вы можете писать на C, C++, встраиваемом C++ или Java, работать в среде Windows, Solaris или QNX Neutrino, и компилировать код для процессоров ARM, MIPS, PowerPC, SH-4, StrongARM, XScale или x86 - и все это из одной и той же IDE. Вы даже можете работать с различными языками и процессорами одновременно.

Вы убедитесь, что QNX Momentics очень легок в изучении, поскольку все инструменты его IDE используют один и тот же пользовательский интерфейс. Кроме того, IDE построена на открытом стандарте Eclipse, поэтому вы можете легко добавлять в нее плагины "третьих" производителей или разрабатывать ваши собственные. В любом случае, новый инструментарий органично интегрируется с уже существующим в IDE.

> ИНТЕГРИРОВАННЫЙ КОМПЛЕКТ РАЗРАБОТЧИКА QNX MOMENTICS

Монитор целевой системы предоставляет динамическое наблюдение за атрибутами потоков, использованием процессора и памяти, открытых файловых дескрипторах, блокировках потоков и множеством других параметров

Профайлер приложений позволяет вам анализировать использование процессора для множества процессов и множества целевых систем.

Анализатор ОЗУ дает вам возможность быстро определить переполнения буферов, некорректные освобождения памяти и прочие типичные ошибки

QNX Momentics предлагает набор визуальных средств для анализа и настройки производительности вашей системы.

В состав QNX Momentics входят:

Редакторы кода C, C++ и Java – редакторы предоставляют стандартные функции типа "вырезать" и "вставить", добавление комментариев, отступы, характерные для каждого языка, а также ряд расширенных функций, таких как дополнение кода, шаблоны, контекстно-зависимая справка и список деклараций, позволяющий сразу перейти к любому идентификатору в проекте.

Символьный отладчик – позволяет вам отлаживать несколько процессов одновременно, даже если они написаны на разных языках и распределены по нескольким процессорам различной архитектуры. Отладчик также может динамически подключаться к выполняющимся процессам, производить "посмертный" анализ dump-файлов и отлаживать многопоточные приложения, отслеживая каждый поток и передачу управления между ними.

Построитель встраиваемых систем – берет на себя работу по созданию загрузочных образов и образов ППЗУ для вашей целевой системы. Включает в себя браузер компонентов, помогающий вам быстро выбрать нужные модули, и оптимизатор, выясняющий зависимости модулей и делающий разделяемые библиотеки максимально компактными.

Монитор целевых систем – предоставляет большой объем информации о системе и процессах, как динамически, так и "посмертно". Вы можете анализировать переменные окружения, атрибуты потоков, состояние сигналов, карты памяти процессов, файловые дескрипторы, блокировки.

Профайлер приложений – позволяет вам быстро обнаруживать неэффективный код и алгоритмы. Профайлер поддерживает как статистическое профилирование для пассивного анализа, так и инструментированное профилирование для точных измерений. Он также может работать на уровне исходного текста, показывая строки кода, потребляющие больше всего ресурсов процессора.

Анализатор ОЗУ – графически отображает использование памяти вашими программами и позволяет вам отслеживать ошибки типа переполнения буферов, некорректного использования памяти и двойного освобождения.

Системный профайлер – работает с диагностической версией микроядра, помогая вам анализировать поведение всей системы в целом. Предоставляет разнообразные фильтры событий и опции отображения, позволяющие вам получить общую картину взаимодействия компонентов системы и в подробностях рассмотреть критические участки.

Визуальный построитель графических приложений PhAB – позволяет вам создавать сложные графические интерфейсы с легкостью щелчка мыши.

ПРОФЕССИОНАЛЬНЫЕ СЕРВИСЫ QNX

Вся деятельность компании QNX Software Systems направлена на то, чтобы помочь вам разрабатывать надежные и конкурентоспособные

продукты в кратчайшие сроки. Поэтому QNX Software Systems в дополнение к своим продуктам предоставляет широкий спектр сервисов для содействия вам на всех этапах вашего проекта, от разработки до сопровождения. В перечень предоставляемых сервисов, например, входят:

- заказные разработки пакетов поддержки процессорных плат (BSP);
- заказные разработки драйверов;
- консалтинг и рецензирование проектов.

QNX Software Systems также предоставляет профессиональное обучение по самой разнообразной тематике: написание драйверов, программирование многопоточных задач, организация файловой системы в ППЗУ, и многое другое. Мало того, если у вас нет возможности посетить учебные центры QSS, это тоже не проблема: QSS обеспечивает обучение в том числе и на территории заказчика.

ПРЕДНАЗНАЧЕННАЯ ДЛЯ РАЗРАБОТЧИКОВ ВСТРАИВАЕМЫХ СИСТЕМ

ОСРВ QNX Neutrino включает в себя все, что необходимо для разработки надежных, масштабируемых, высокопроизводительных встраиваемых систем: отказоустойчивую архитектуру на основе микроядра, динамическую расширяемость, распределенные вычисления, полнофункциональную графическую оболочку, гарантированное время реакции, переносимость на основе POSIX, богатый выбор поддерживаемых процессоров и периферийного оборудования, мощную и при этом простую в освоении IDE – и все это из рук истинных профессионалов, по-настоящему ценящих своих клиентов. Свяжитесь с нами сегодня, чтобы узнать, как QNX Neutrino поможет обеспечить вашему новому проекту исключительную надежность, масштабируемость и производительность.

О КОМПАНИИ QNX SOFTWARE SYSTEMS

Основанная в 1980 году, компания QNX Software Systems на сегодняшний день занимает позицию лидера в области технологий операционных систем реального времени, основанных на микроядре. Присущая ОСРВ QNX Neutrino надежность, масштабируемость и малые размеры делают выбор этой технологии наиболее предпочтительным для разработки передовых встраиваемых приложений. Компания имеет по всему миру крупнейших партнеров, таких как Cisco, Delphi, Siemens, Alcatel, Texasco и Ford, и устойчивый рынок сбыта своей продукции во многих отраслях промышленности, в частности в телекоммуникационной сфере, в сфере медицинского оборудования, транспортирования, систем безопасности, робототехники, промышленной автоматике и других, в которых применяются ответственные приложения. Компания QNX Software Systems имеет штаб-квартиру в Оттаве, Канада, и представительства в Северной Америке, Европе и Азии. Продукция компании распространяется в более чем 100 странах по всему миру.

КРАТКИЙ ОБЗОР QNX NEUTRINO

АРХИТЕКТУРА НА ОСНОВЕ МИКРОЯДРА

Динамически обновляемые сервисы и приложения
Изоляция сбоев и автоматическое самовосстановление
Обмен сообщениями для обеспечения модульности

РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ

Прозрачный доступ к удаленным ресурсам
Простота проектирования отказоустойчивых кластеров

ПОДДЕРЖКА SMP НА УРОВНЕ МИКРОЯДРА

Поддержка тесно связанного SMP для MIPS, PowerPC и x86
Автоматическое масштабирование многопоточных приложений

ОТЛАДОЧНАЯ ВЕРСИЯ МИКРОЯДРА

Анализ производительности и оптимизация всей системы в целом
Быстрое выявление ошибок синхронизации и скрытых дефектов

QNX PHOTON MICROGUI

Настраиваемый вид и поведение
Сложные многоуровневые дисплеи
Расширяемая поддержка мультимедиа

СТЕКИ ПРОТОКОЛОВ

NetBSD (IPSec, IPv6) и встраиваемый стек TCP/IP

ФАЙЛОВЫЕ СИСТЕМЫ

Образная, в ОЗУ, в flash-ППЗУ, QNX, Linux, DOS, CD-ROM, DVD,
NFS, CIFS, со сжатием, пакетная

ДРАЙВЕРЫ УСТРОЙСТВ

Аудио, терминальные, диски, графика, ввод, сеть, параллельные,
последовательные, принтеры, USB

АДМИНИСТРАТОР СИСТЕМ ВЫСОКОЙ ГОТОВНОСТИ

Квитанции работоспособности для ранней диагностики отказов
Интеллектуальное восстановление отказавших компонентов

ПОДДЕРЖКА JAVA

Среда исполнения Java в стандарте Java Powered
Полный доступ к сервисам ОС для Java-приложений

ПРЕДСКАЗУЕМОСТЬ И ПРОИЗВОДИТЕЛЬНОСТЬ РЕАЛЬНОГО ВРЕМЕНИ

Вытесняющий планировщик с выбором дисциплины планирования
Распределенное наследование приоритетов

ПОДДЕРЖКА POSIX

POSIX 1003.1-2001, с многопоточностью и расширениями
реального времени

ВЫБОР ЦЕЛЕВОГО ПРОЦЕССОРА

ARM, MIPS, PowerPC, SH-4, StrongARM, XScale, x86

КОМПЛЕКТ РАЗРАБОТЧИКА QNX MOMENTICS

Графическая IDE, BSP, DDK, инструментарий GNU
Поддержка множества инструментальных и целевых платформ и
языков программирования

В ТЕЧЕНИЕ БОЛЕЕ 20 ЛЕТ
ТЕХНОЛОГИЯ QNX ПОМОГАЛА
РАЗРАБОТЧИКАМ СОЗДАВАТЬ
САМЫЕ НАДЕЖНЫЕ В МИРЕ
ПРИЛОЖЕНИЯ –
ОТ МЕДИЦИНСКИХ ПРИБОРОВ
ЖИЗНЕОБЕСПЕЧЕНИЯ И
АВТОМОБИЛЬНЫХ
ТЕЛЕМАТИЧЕСКИХ СИСТЕМ
ДО РАСПРЕДЕЛЕННЫХ
ПРОМЫШЛЕННЫХ СИСТЕМ
УПРАВЛЕНИЯ. ТЕПЕРЬ САМАЯ
НАДЕЖНАЯ ОС РЕАЛЬНОГО
ВРЕМЕНИ ПОДДЕРЖИВАЕТ ЕЩЕ
И САМЫЙ ПРОГРЕССИВНЫЙ
ИНСТРУМЕНТАРИЙ.



QNX SOFTWARE SYSTEMS LTD.



QNX SOFTWARE SYSTEMS LTD.



ЗАО «СВД Софтвр»
официальный дистрибьютор
компании QSSL



**PREMIER
DISTRIBUTOR**

196135, Россия, Санкт-Петербург
пр. Ю.Гагарина, 23
тел. +7 (812) 373-0260, 102-0833
факс +7 (812) 373-0497
e-mail: info@swd.ru, <http://www.swd.ru>

ОС реального времени **QNX[®] NEUTRINO[®]** ОСРВ НА ОСНОВЕ МИКРОЯДРА

НАДЕЖНОСТЬ, ПРОВЕРЕННАЯ ВРЕМЕНЕМ.

**МАСШТАБИРУЕМОСТЬ, ОТКРЫВАЮЩАЯ
НОВЫЕ ВОЗМОЖНОСТИ.**

**ПРОИЗВОДИТЕЛЬНОСТЬ,
НЕ ИМЕЮЩАЯ АНАЛОГОВ.**