

Системы реального времени

Лекция 1. Введение. Начальные сведения о QNX

к. ф-м. н., доц. Вахтин А. А.

Литература

- Зыль, Сергей Николаевич. Операционная система реального времени QNX : от теории к практике / Сергей Зыль .— 2-е изд. — СПб. : БХВ-Петербург, 2004 .— 191 с.
- Практика работы с QNX / Д. Алексеев [и др.] .— М. : КомБук, 2004 .— 431 с.
- Операционная система реального времени QNX Neutrino 6.3. Системная архитектура : пер. с англ. — СПб. : БХВ-Петербург, 2006 .— 316 с.

Системы реального времени

Системами реального времени являются такие информационные системы, в которых корректность выходной информации зависит:

- от правильности применения алгоритмов;
- от времени появления результатов обработки информации.

Классификация операционных систем

- GPOS (General Purpose Operation Systems – операционные системы общего назначения)
- RTOS (Real Time Operation Systems – операционные системы реального времени)

Операционные системы реального времени

- *Жесткого реального времени* – гарантирует выполнение каких-то действий за определенный интервал времени.
- *Мягкого реального времени* – успевает выполнить заданные действия за заданное время с некоторой вероятностью.

QNX – операционная система жесткого реального времени

- QNX2 - до 1991 года система была запрещена к вывозу из Северной Америки как стратегический ресурс. Разработана для ЭВМ на базе процессоров Intel 286.
- QNX4 - самая распространенная на сегодняшний день ОС QNX в промышленности. Использует защищенный режим, применяется на процессорах не ниже Intel 386.
- QNX6 (или QNX Neutrino) - может использоваться не только на x86-совместимых ЭВМ. Обеспечивается максимальная переносимость в QNX исходных кодов, написанных для ОС Linux.

Категории ОС QNX

- *Инструментальная система (среда разработки)* – это ЭВМ со средствами, позволяющими формировать образ *целевой системы*.
- *Целевая система (среда исполнения)* предназначена для эксплуатации. Представляет собой инструментальную систему, из которой удалено все ненужное для данной прикладной задачи.

Целевая система QNX 6.2

- QNX Real Time Platform (QNX RTP) – полнофункциональная среда разработки, бесплатна для некоммерческого использования.
- QNX Networking Infrastructure Platform (QNX NIP) – коммерческий дистрибутив, представляющий собой расширение QNX RTP дополнительными программными пакетами, ориентированными на производителей сетевого оборудования.

Инструментальная система QNX

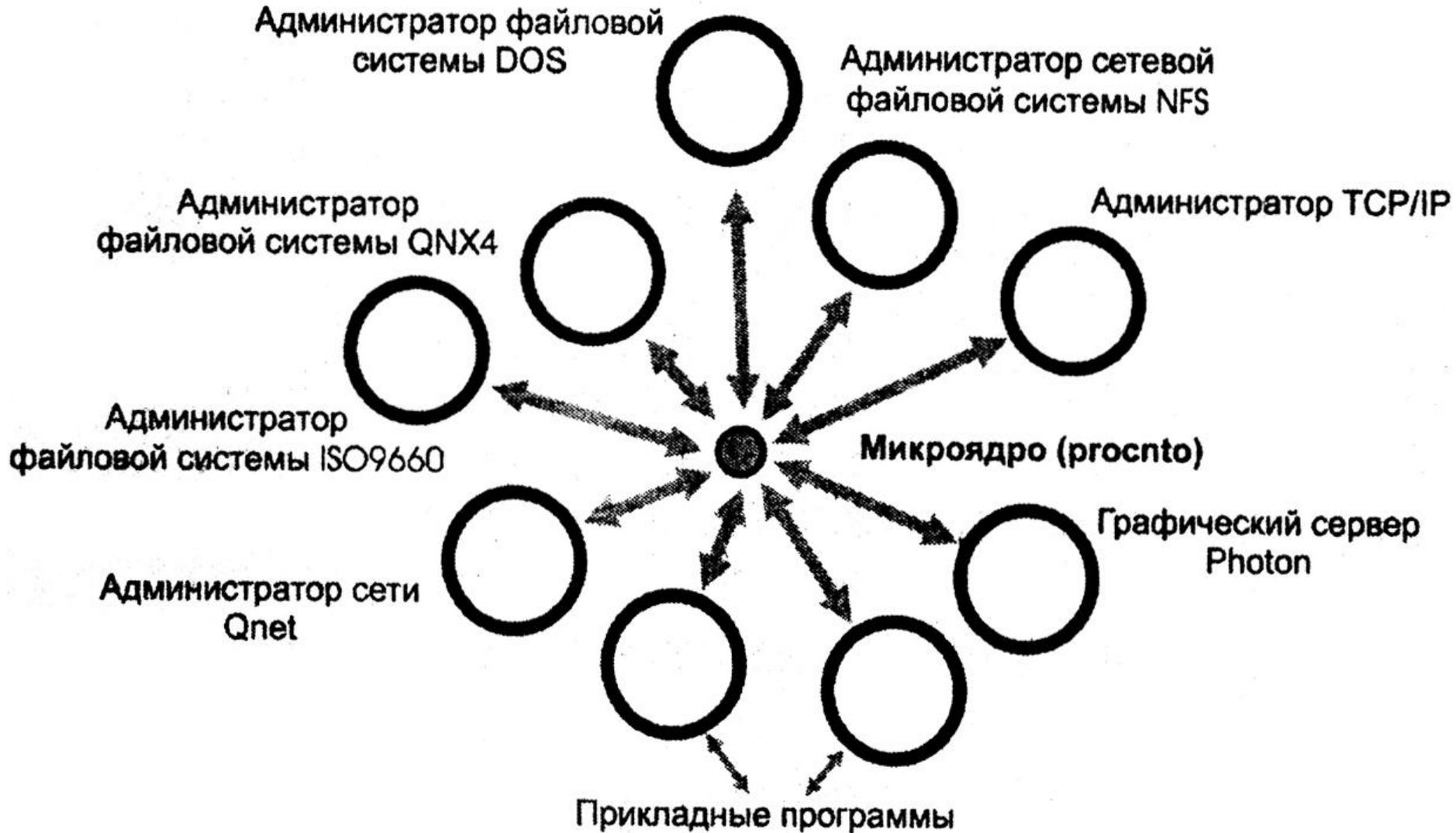
Momentics

- Non-Commercial Edition (NC) – ознакомительный комплект разработчика, бесплатный для некоммерческого использования;
- Standard Edition (SE) – пакет разработчика, позволяющий вести коммерческую разработку ПО, формировать целевые системы для разных платформ;
- Professional Edition (PE) – расширенный пакет разработчика, дополненный интегрированной средой разработки QNX IDE, основанной на технологии Eclipse, а также расширенной базой примеров в исходных текстах и рядом дополнительных компонентов.

POSIX-совместимость

POSIX (Portable Operation Systems Interface for Unix) – это развивающийся стандарт, призванный обеспечить переносимость исходных текстов программ между ОС разных производителей.

Архитектура QNX



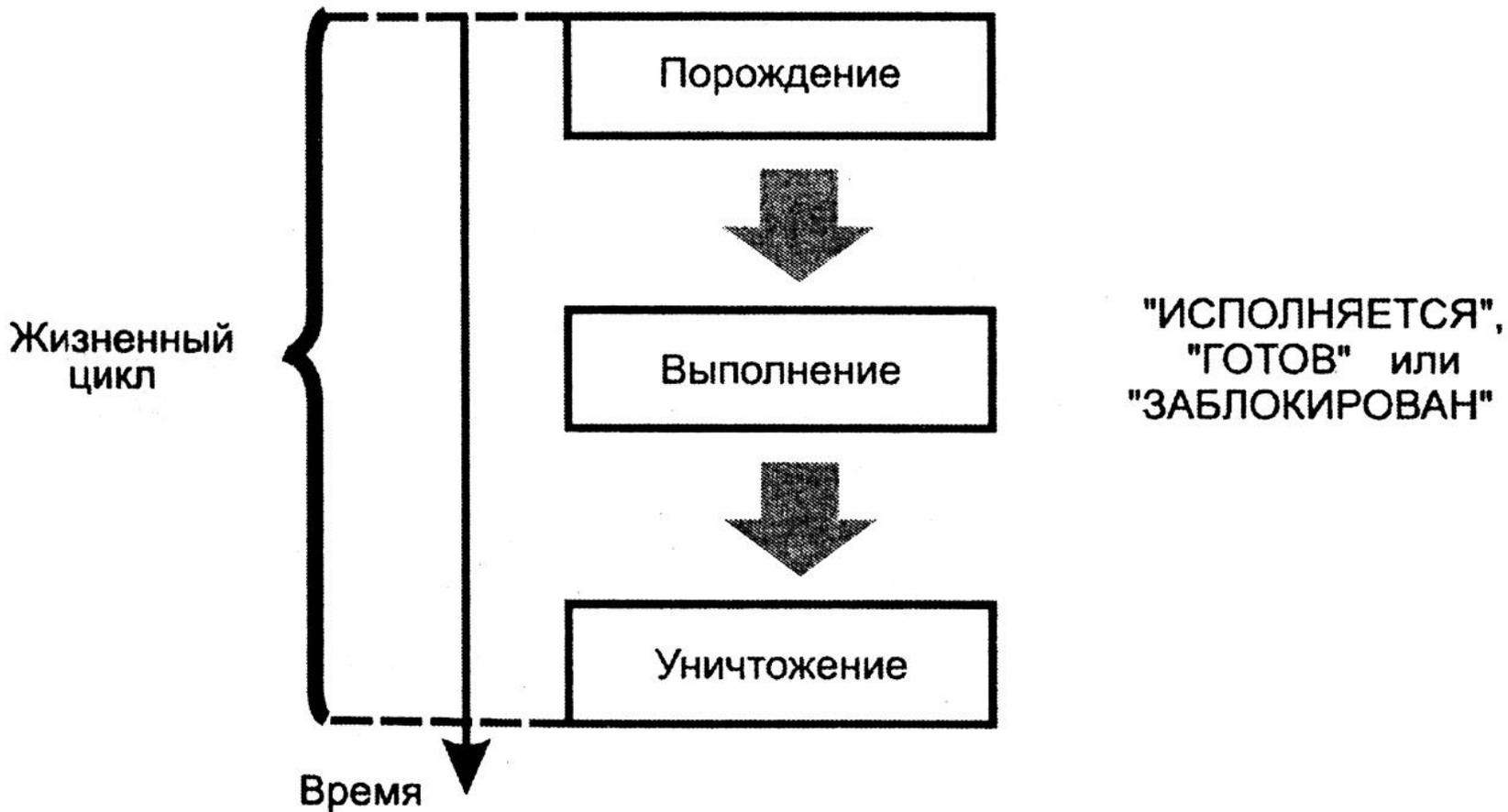
Применение многопоточности

- для распараллеливания задачи на многопроцессорных ЭВМ;
- для более эффективного использования процессора (например, когда один поток ожидает пользовательский ввод, другой может выполнять расчеты);
- для облегчения совместного использования данных (все потоки процесса имеют свободный доступ к данным процесса).

Функции микроядра Neutrino:

- создание и уничтожение потоков;
- диспетчеризация потоков;
- синхронизация потоков;
- механизмы IPC (Inter Process Communication);
- поддержка механизма обработки прерываний;
- поддержка часов, таймеров и таймаутов.

Жизненный цикл потока



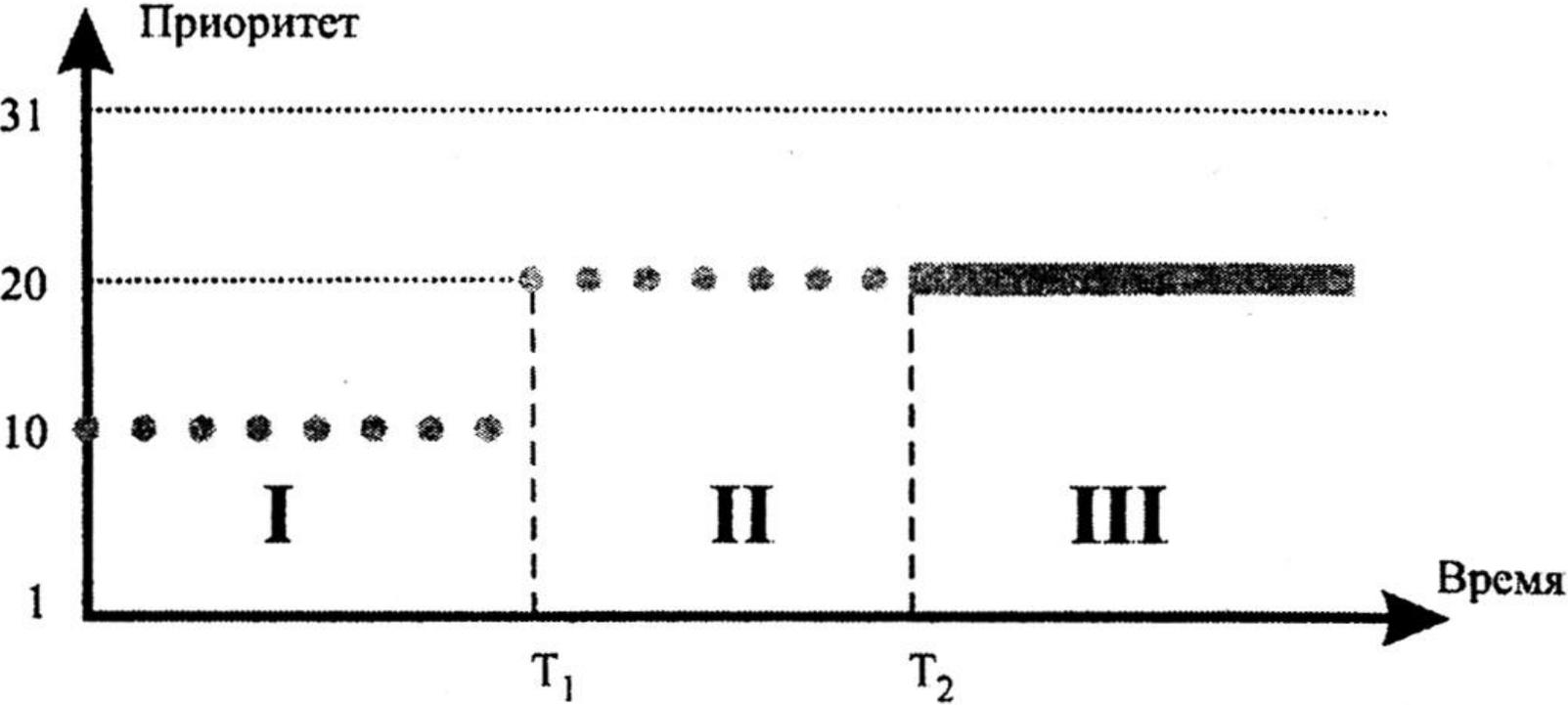
Условия выполнения диспетчеризации:

- исполняющийся на процессоре поток перешел в заблокированное состояние;
- поток с более высоким, чем у исполняющегося потока, приоритетом перешел в состояние готовности, т.е. происходит вытеснение потока (это свойство ОС называют вытесняющей многозадачностью);
- исполняющийся поток сам передает право исполнения процессора другому потоку (вызывает функцию *sched_yield()*).

Дисциплины диспетчеризации

- *FIFO (First In First Out – первый вошел – первый вышел)*
- *Карусельная диспетчеризация (Round Robin)*
- *Адаптивная диспетчеризация*
- *Спорадическая диспетчеризация*

Работа клиент-управляемого приоритета



Выполнение запроса клиента Б, имеющего приоритет 20



Выполнение запроса клиента А, имеющего приоритет 10

Синхронизация потоков

- взаимоисключающая блокировка (**Mutual exclusion lock** – mutex, *мутекс*)
- условная переменная (*condition variable*, или *condvar*)
- барьер
- ждущая блокировка
- блокировка чтения/записи (*rwlock*)
- семафор – это мутекс со счетчиком.

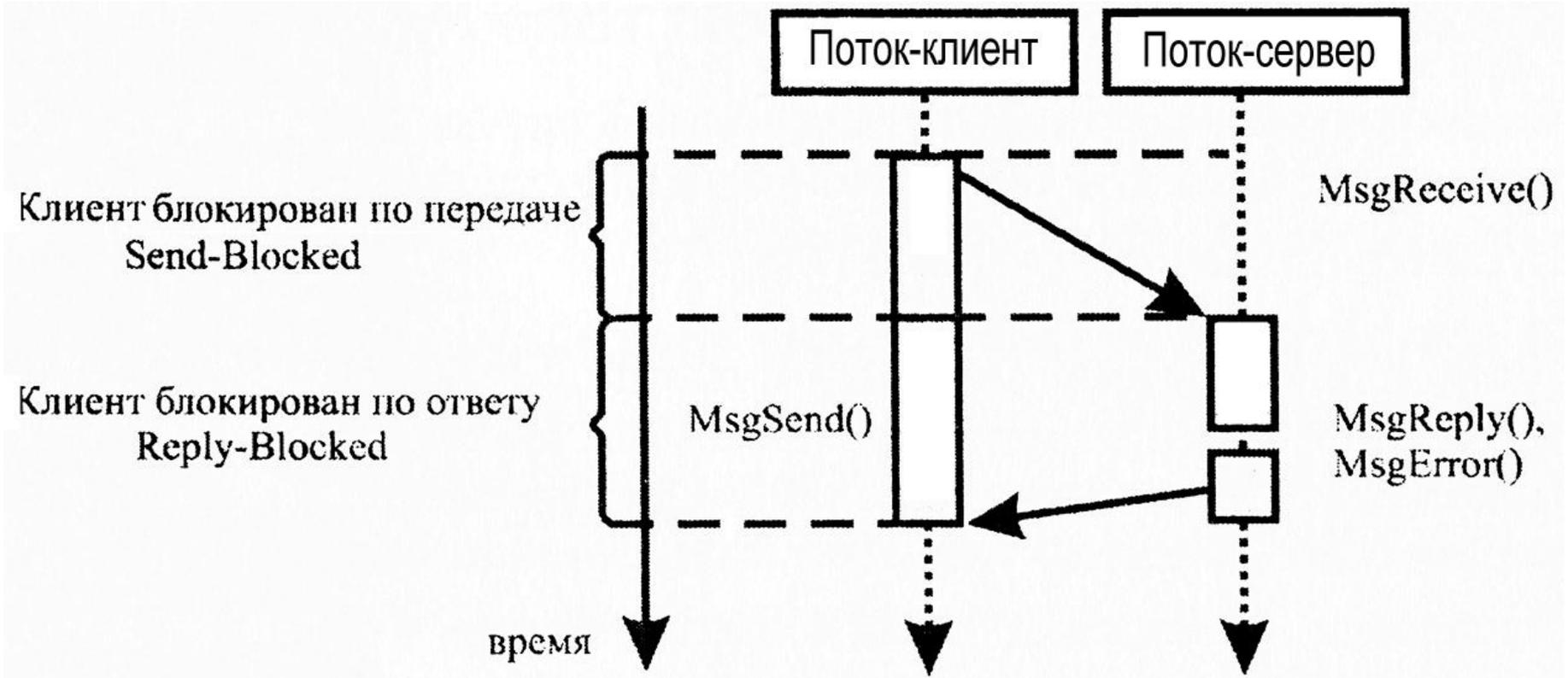
Межзадачное взаимодействие (IPC – Inter Process Communication) в микроядре Neutrino

- синхронные сообщения QNX;
- Pulses (импульсы);
- асинхронные сообщения QNX;
- сигналы POSIX (как простые, так и реального времени).

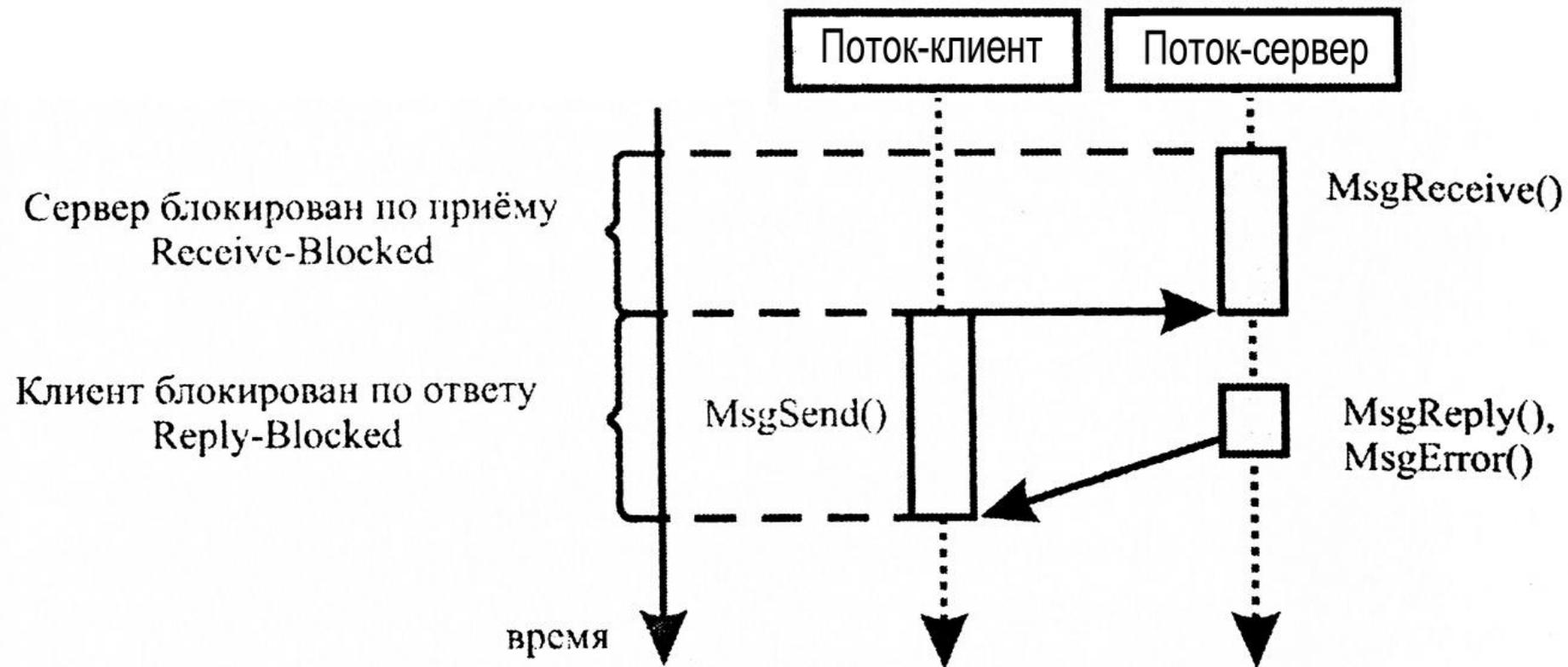
Межзадачное взаимодействие (IPC – Inter Process Communication) дополнительные

- очереди сообщений POSIX (реализованы в администраторе очередей *mqdque*);
- разделяемая память (реализована в администраторе процессов);
- именованные каналы (реализованы в администраторе файловой системы QNX4);
- неименованные каналы (реализованы в администраторе каналов *pipe*).

Поток-сервер еще не ждет сообщения от потока-клиента



Поток-сервер уже ждёт сообщения от потока-клиента



Pulses (импульсы)

- не блокирует отправителя;
- может быть получен сервером как обычное сообщение;
- ставится в очередь, если получатель не блокирован по приему (RECEIVE-Blocked).



Асинхронные сообщения

Asynctmsg_MsgSend()

Asynctmsg_MsgReceive()

Asynctmsg_MsgReply()

Asynctmsg_MsgError()

Сигналы POSIX

- Традиционные UNIX-сигналы;
- POSIX-сигналы реального времени;
- Специальные QNX-сигналы.

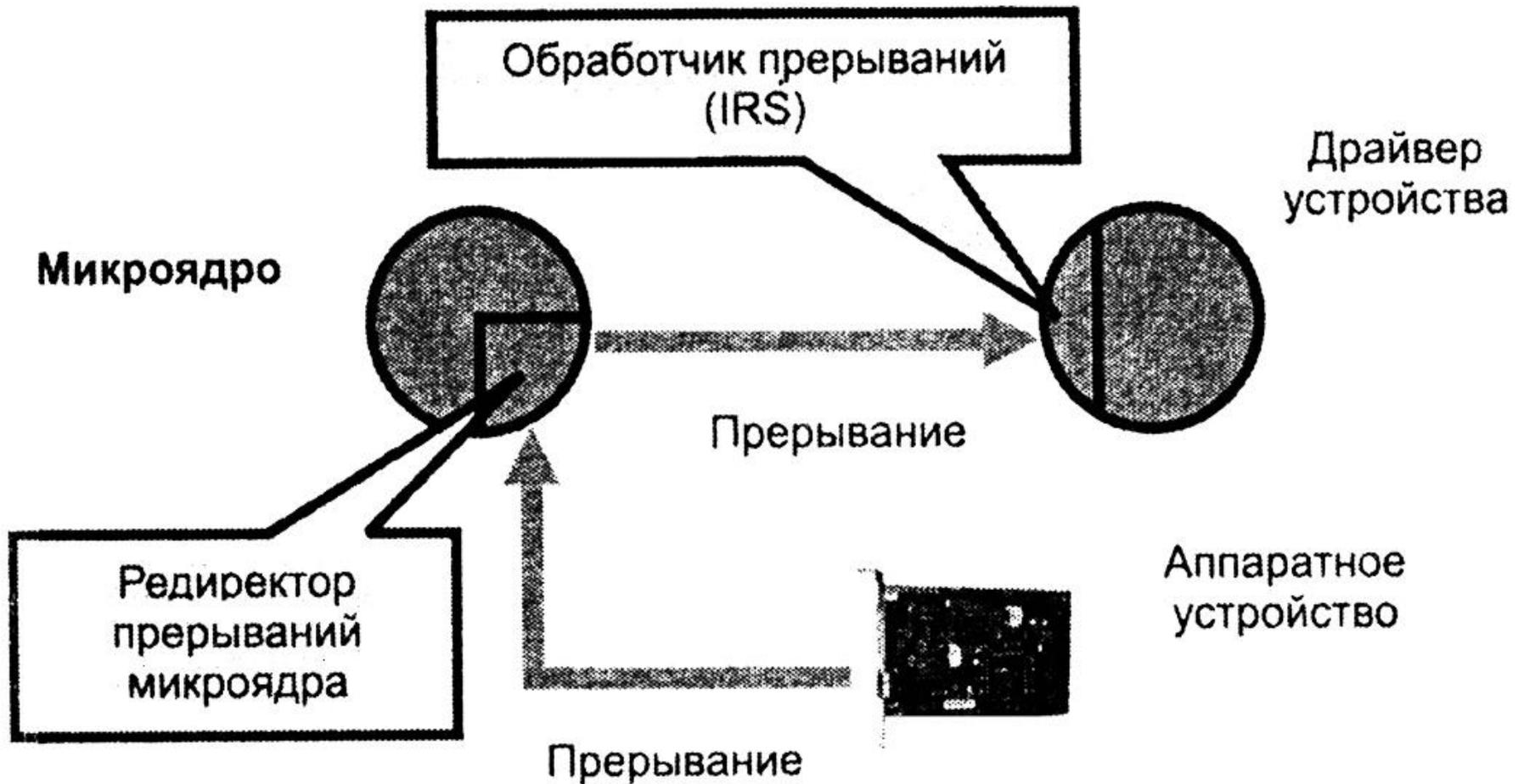
Классификация таймеров в QNX

- *по способу задания времени:* абсолютный (если указано время срабатывания) или относительный (если задан интервал срабатывания относительно настоящего момента);
- *по кратности срабатывания:* однократный или периодический.

Уведомление о срабатывании таймера

- послать импульс;
- послать сигнал;
- создать поток.

Обработка прерываний



Обработчик прерывания (ISR – Interrupt service routine)

- получает управление посредством так называемого *дальнего вызова*, т. е. не из процесса, к которому принадлежит функция;
- имеет приоритет выше, чем процессы;
- выполняется в контексте процесса, в который он включен;
- может маскироваться и блокироваться.

Администратор процессов

- управление процессами;
- управление механизмами защиты памяти;
- поддержка механизма разделяемой памяти и IPC на ее основе;
- управление пространством путевых имен.

Атрибуты процесса

- идентификатор процесса (process ID – *PID*);
- идентификатор родительского процесса (parent process ID – *PPID*);
- реальные идентификаторы владельца и группы (*UID* и *GID*);
- эффективные идентификаторы владельца и группы (*EUID* и *EGID*);
- текущий рабочий каталог;
- управляющий терминал;
- маска создания файлов (*umask*);
- номер приоритета;
- дисциплина диспетчеризации.

Жизненный цикл процесса

1. Создание.
2. Загрузка кода и данных процесса в ОЗУ.
3. Выполнение потоков.
4. Завершение.

Управление механизмами защиты памяти

Администратор процессов QNX
обеспечивает поддержку полной защиты
памяти (так называемую *виртуальную
память*) процессов.

Управление пространством путевых имен

Управление ресурсами ввода/вывода не встроено в микроядро, а реализуется посредством дополнительных процессов – *администраторов ресурсов*.

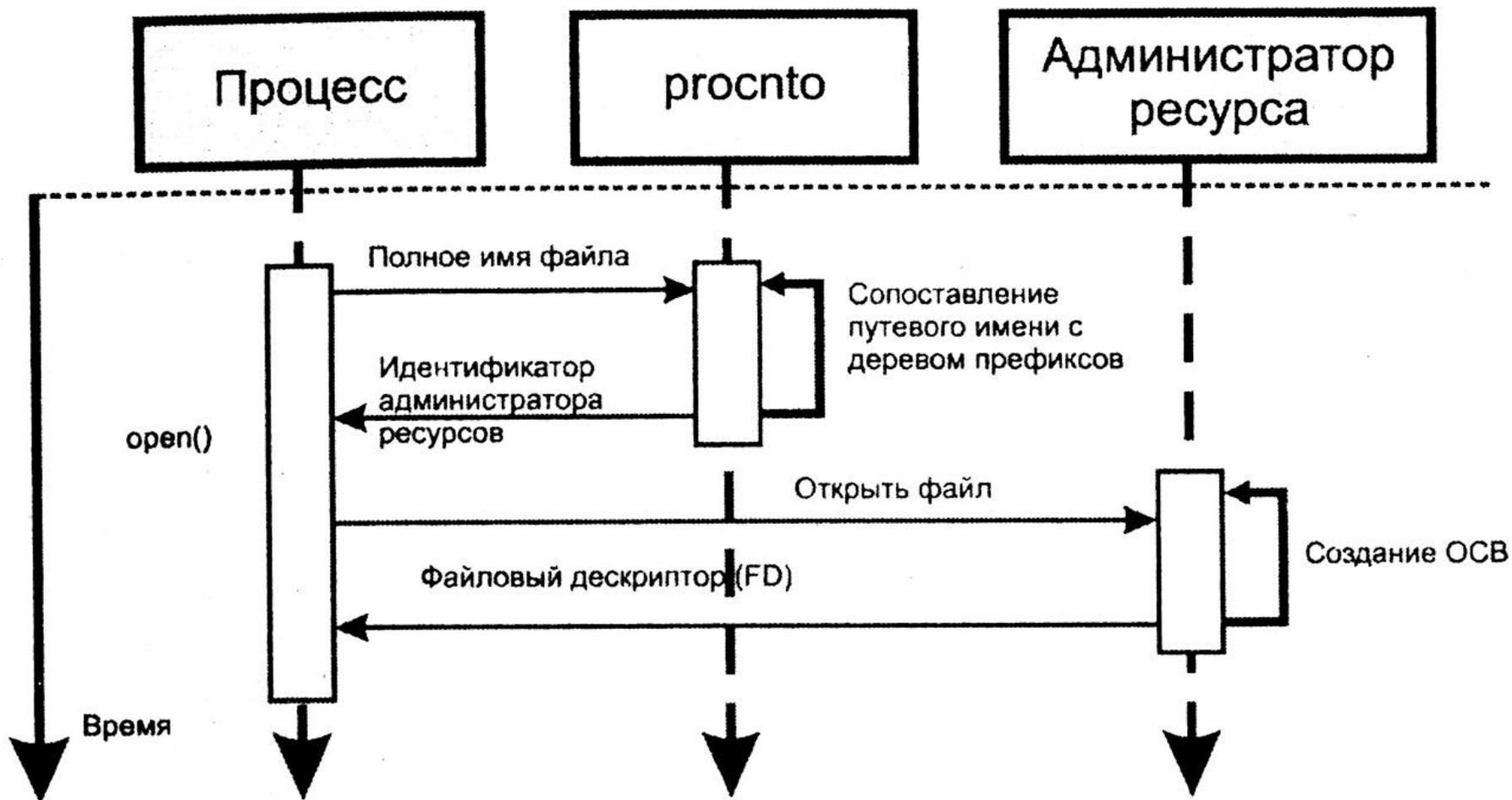
Записью файлов на диск управляет *администратор файловой системы*, отправкой данных по сети – *администратор сети*.

Администраторы ресурсов интегрируют свои услуги в ОС QNX с помощью *администратора процессов*, который предоставляет механизм пространства путевых имен.

procnto при загрузке регистрирует в пространстве имен несколько префиксов:

- / – корень (*root*) файловой системы, к которому монтируются все остальные префиксы;
- /*rgos*/ – каталог, в который отображается информация о запущенных процессах, представленных посредством их идентификаторов (PID);
- /*proc/boot*/ – каталог, в который в виде плоской файловой системы отображаются файлы, входящие в состав загрузочного образа QNX;
- /*dev/zero* – устройство, которое при чтении из него всегда возвращает нуль. Используется, например, для того, чтобы заполнить нулями страницы памяти;
- /*dev/mem* – устройство, представляющее всю физическую память.

Определение администратора ресурсов



Разделяемая память

Механизм разделяемой памяти реализован в *администраторе процессов* и предназначен для быстрого обмена большими объемами данных между процессами.

Загрузка динамической библиотеки компоновщиком

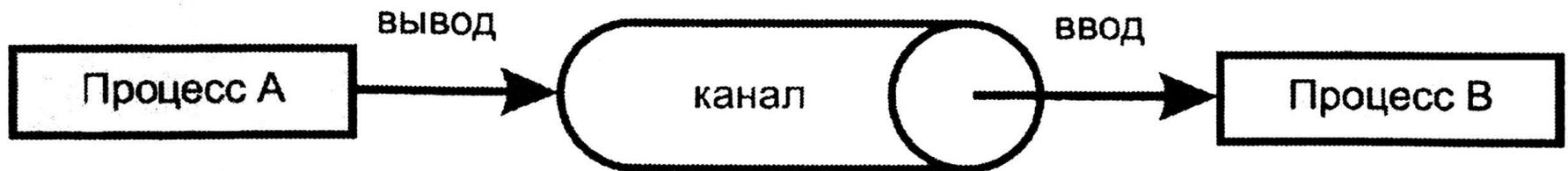
1. Компоновщик выполняет проверку, была ли загружена в память библиотека.
2. Если указано абсолютное путевое имя, то загружается именно указанная библиотека.
3. Если указано не абсолютное путевое имя, то динамический компоновщик осуществляет поиск библиотеки в каталогах, перечисленных в переменной окружения `LD_LIBRARY_PATH`.
4. Если разделяемая библиотека попрежнему не найдена и если динамическая секция исполняемого кода содержит тег `DT_RPATH`, то поиск производится и по пути, определенному в теге `DT_RPATH`.
5. Если разделяемая библиотека попрежнему не найдена то компоновщик выполняет поиск в каталогах, определенных в переменной окружения `LD_LIBRARY_PATH` модуля *procnto*.

Очереди сообщений POSIX

Очереди сообщений POSIX реализованы с помощью администратора очередей *mqueue*. Администратор *mqueue* регистрирует в пространстве имен путей префикс */dev/mqueue*, имеющий тип *каталог*. Очереди сообщений POSIX – это именованные объекты, поэтому данный механизм можно использовать для обмена данными между процессами как в рамках одной ЭВМ, так и между процессами, работающими на разных узлах сети.

Именованные и неименованные каналы

Назначение канала – обеспечить однонаправленную передачу данных от одного процесса к другому. При этом вывод одной программы соединяется с вводом другой.



Получение информации о процессах

- **ps** – основная POSIX-утилита для мониторинга процессов. Она включена в QNX как для совместимости POSIX, так и для удобства администраторов, недавно работающих в QNX;
- **sin** – информативная QNX-утилита мониторинга процессов. С помощью **sin** можно, задав соответствующую опцию, получить информацию о процессах на другом узле сети Qnet. По умолчанию **sin** выдает для каждого процесса PID, размер кода, размер стека и использование процессора.

- `args` – показать аргументы процессов;
- `cpu` – показать использование ЦПУ;
- `env` – показать переменные окружения процессов;
- `fds` – показать открытые файловые дескрипторы;
- `flags` – показать флаги процессов;
- `info` – показать общую информацию о системе;
- `memory` – показать память, используемую процессами;
- `net` – показать информацию об узлах сети;
- `registers` – показать состояние регистров;
- `signals` – показать сигнальные маски;
- `threads` – показать информацию по потокам;
- `timers` – показать таймеры, установленные процессами;
- `users` – показать реальные и эффективные идентификаторы владельцев и групп процессов.

Окно утилиты psin

The screenshot shows the System Process Inspector window. The top pane displays a list of processes, with 'psin' selected. The bottom pane shows the memory dump for the selected process, including file names, addresses, offsets, and permissions.

Name	Pid	Code	Data	Stack	Vstack	CPU
procnto	1	0	0	0	0	1202135
io-graphics	745502	84K	948K	28K	780K	4343
pwm	393238	76K	188K	8K	516K	1016
shelf	438298	52K	748K	28K	648K	2207
pterm	761869	48K	268K	12K	516K	610
sh	761877	144K	72K	8K	516K	8
psin	782360	28K	208K	12K	516K	32
bkgdmgr	475163	12K	132K	12K	516K	7692

Name	Vaddr	Offset	Code	Map	Data	Map
782360 psin						
/dev/zero	08045000	00000000			12K	_Rw__
usr/photn/bin/psin	08048000	00000000	28K	SR_X_	12K	PRW__
/dev/zero	08052000	00000000			96K	_Rw__
ldqnx.so.2	B0300000	00448000	312K	SR_X_	16K	PRWX_
libphexlib.so.2	B8200000	00000000	60K	SR_X_	4K	PRW__
libAp.so.2	B8210000	00000000	56K	SR_X_	8K	PRW__

localhost 95M 1 2189Mhz Intel 686 F6M15S11

QNX System Information

The screenshot shows the QNX System Information window in QNX Momentics Integrated Development Environment. The window is titled "QNX System Information - QNX Momentics Integrated Development E" and has a menu bar with File, Edit, Perspective, Project, Window, and Help. The interface is divided into two main panes. The left pane is the Target Navigator, showing a tree view of the system components. The right pane is the System Summary, displaying various system details and a table of processes.

Target Navigator

- local (daddy)
 - procnto (1)
 - tinit (2)
 - slogger (3)
 - mqueue (12292)
 - pci-bios (5)
 - devb-eide (6)
 - devc-con (7)
 - fs-pkg (8)
 - pipe (4105)
 - io-net (200714)
 - devc-pty (45067)
 - devc-par (77836)
 - eclipse (1499149)
 - spooler (77838)
 - sh (1499151)
 - devb-fdc (147472)
 - random (155665)
 - devc-ser8250 (1106)
 - dumper (167955)
 - Photon (327700)
 - io-audio (110613)
 - font sleuth (405526)
 - pwm (1380375)
 - io-graphics (372760)
 - devi-hirun (401433)
 - shelf (1425434)
 - bkgdmgr (1462299)

System Summary

Hostname: daddy Board: x86pc
OS Version: 6.2.0 (2002/05/17-13:47:15edt)
Boot Date: Thu Jan 01 00:00:00 GMT 1970

1 x86 cpu
x86 @ 939Mhz

System Memory: 101M/511M

Processes (37)

Applications				Servers			
Name	Heap	CPU	Start	Name	Heap	CPU	Start
pwm	80K	111ms	Tue Sep 16 ...	procnto	1M	12mi...	Tue Sep 1...
sh	0	3ms	Tue Sep 16 ...	tinit	16K	1ms	Tue Sep 1...
sh	0	4ms	Tue Sep 16 ...	slogger	64K	5ms	Tue Sep 1...
ped	0	27ms	Tue Sep 16 ...	mqueue	16K	999us	Tue Sep 1...
				pci-bios	16K	13ms	Tue Sep 1...
				devb-eide	12M	57se...	Tue Sep 1...
				devc-con	48K	174ms	Tue Sep 1...
				fs-pkg	1M	13se...	Tue Sep 1...
				pipe	80K	101ms	Tue Sep 1...
				io-net	724K	1sec...	Tue Sep 1...
				devc-pty	112K	101ms	Tue Sep 1...
				devc-par	32K	8ms	Tue Sep 1...
				eclipse	16K	2ms	Tue Sep 1...
				spooler	16K	26ms	Tue Sep 1...
				sh	32K	3ms	Tue Sep 1...
				devb-fdc	200K	5ms	Tue Sep 1...
				random	336K	8ms	Tue Sep 1...

Target Navigator Navigator System Summary Process Information Thread Information Memory Information Malloc Information

Диагностика процессов при сбое

