MicroSoft-Windows

Практическая работа №2:
Концепции и инструменты ОС Microsoft Windows.

Цель: изучить основные термины, концепции и ключевые механизмы ОС; получить представление о полезных инструментах, позволяющих изучать внутренние структуры данных Windows.

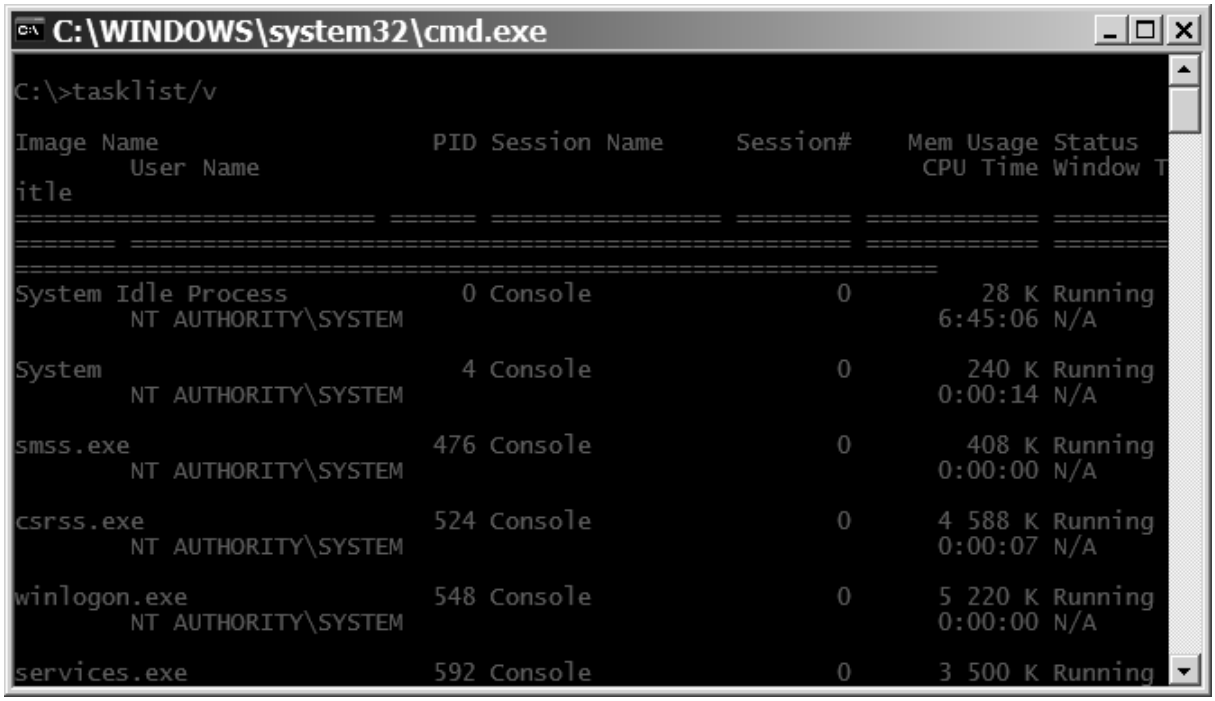
ВОТ КАК-ТО ТАК ВСЕ ЭТО И РАБОТАЕТ

Windows Support Tools включают около **40 утилит**, полезных в администрировании систем на базе **Windows** и устранении неполадок в них.

Вы можете установить **Support Tools**, запустив **Setup.exe** из папки **\Support\Tools** в дистрибутиве любого издания **Windows**.

ЭКСПЕРИМЕНТ: просмотр дерева процессов, 7

Большинство утилит не отображает такой уникальный атрибут, как идентификатор родительского процесса. Значение этого атрибута можно получить программно или с помощью оснастки **Performance**, запросив значение счетчика **Creating Process ID** (**Код создавшего процесса**). Дерево процессов показывается утилитой **tlist.exe** (из **Windows Debugging Tools**), если вы указываете ключ **/t**. Также можно использовать команду **tasklist /v**. Вот образец вывода этой команды:



ЭКСПЕРИМЕНТ: просмотр информации о процессах через диспетчер задач, 9

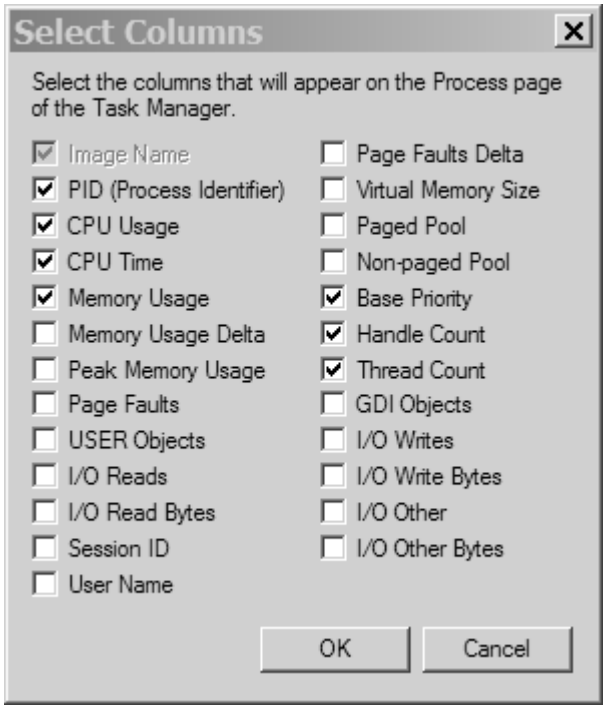
Диспетчер задач отображает список выполняемых в системе процессов. Его можно запустить тремя способами:

1) нажав клавиши **Ctrl+Shift+Esc**;

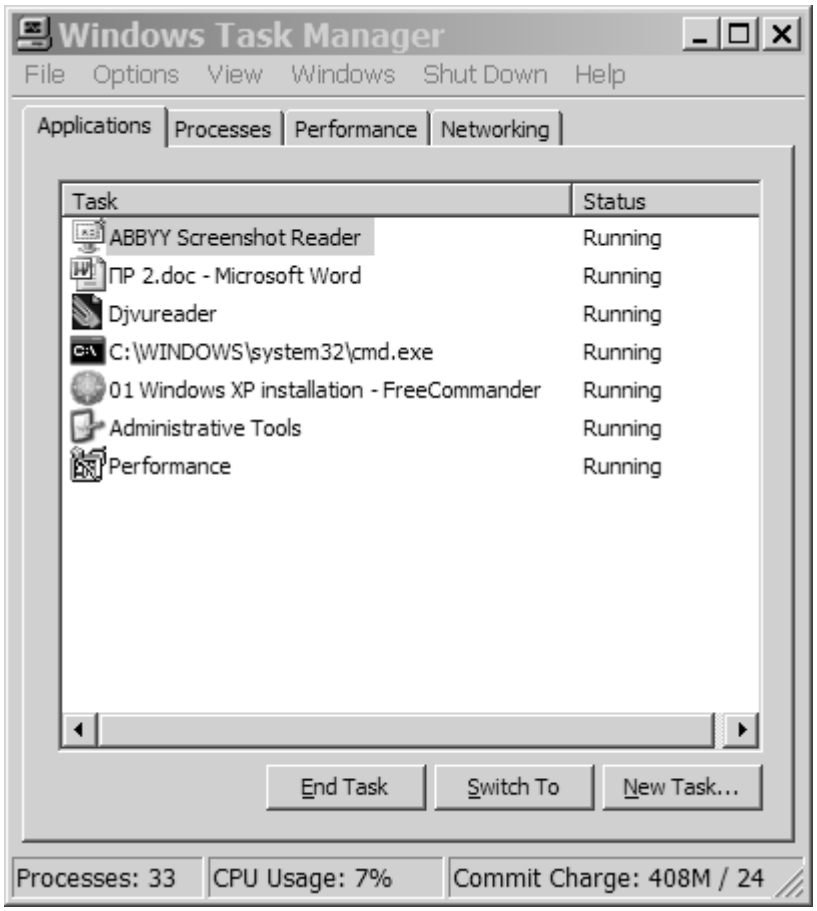
2) щелкнув панель задач правой кнопкой мыши и выбрав команду **Task Manager** (**Диспетчер задач**);

3) нажав клавиши **Ctrl+Alt+Del**.

После запуска диспетчера задач откройте вкладку **Processes** (**Процессы**). Заметьте, что процессы идентифицируются по имени образа, экземплярами которого они являются. В отличие от некоторых объектов процессам нельзя присваивать глобальные имена. Для просмотра более подробных сведений выберите из меню **View** (**Вид**) команду **Select Columns** (**Выбрать столбцы**) и укажите, какая дополнительная информация вас интересует.



Если вкладка **Processes** окна диспетчера задач со всей очевидностью показывает список процессов, то содержимое вкладки **Applications** (**Приложения**) нуждается в пояснениях. На ней отображается список видимых окон верхнего уровня всех объектов «**рабочий стол**» интерактивною объекта **WindowStation** (по умолчанию существуют два объекта «рабочий стол», но вы можете создать дополнительные рабочие столы через Windows-функцию **CreateDesktop**). Колонка **Status** (**Состояние**) даст представление о том, находится ли поток – владелец окна в состоянии ожидания Windows-сообщения. «**Running**» (**Выполняется**) означает, что поток ожидает ввода в окно, а «**Not Responding**» (**Не отвечает**) – что не ожидает (т.е. занят либо ждет завершения операции ввода-вывода или освобождения какого-либо синхронизирующего объекта).

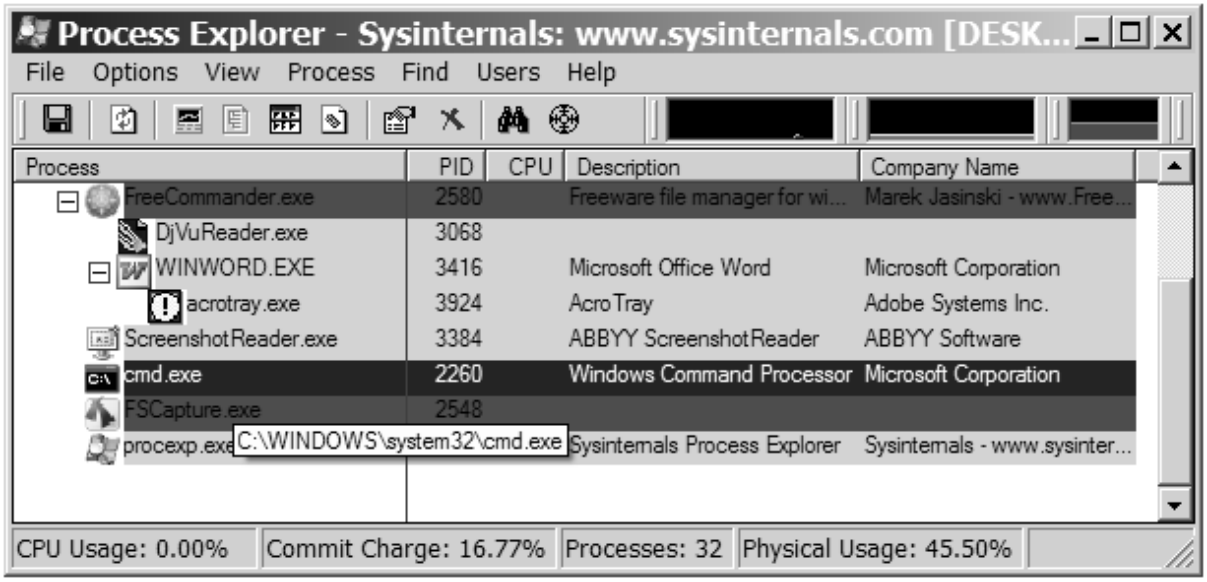


Вкладка **Applications** позволяет идентифицировать процесс, которому принадлежит поток, владеющий каким-либо окном задачи. Для этого щелкните правой кнопкой мыши имя задачи и выберите команду **Go To Process** (**Перейти к процессам**).

ЭКСПЕРИМЕНТ: просмотр детальных сведений о процессах с помощью Process Explorer, 11

Скачайте **Process Explorer** с сайта **www.sysinternals.com** и запустите программу. При первом запуске вы увидите сообщение о том, что на данный момент символы не сконфигурированы. Когда они корректно сконфигурированы **Process Explorer** может обращаться к символьной информации для отображения символьного имени стартовой функции потока и функций в его стеке вызовов (для этого нужно дважды щелкнуть процесс и выбрать вкладку **Threads**). Эта информация полезна для идентификации того, что именно делают потоки внутри процесса. Для доступа к символам вы должны установить **Debugging Tools**. Потом щелкнуть **Options**, выбрать **Configure Symbols** и набрать подходящий путь **Symbols**.

При запуске **Process Explorer** по умолчанию выводит список процессов в верхней половине окна, а список открытых описателей для выбранного на данный момент процесса в нижней половине. Если вы задержите курсор мыши над именем процесса, программа также показывает описание образа, название компании и полный путь.



Вот как использовать базовые возможности **Process Explorer**:

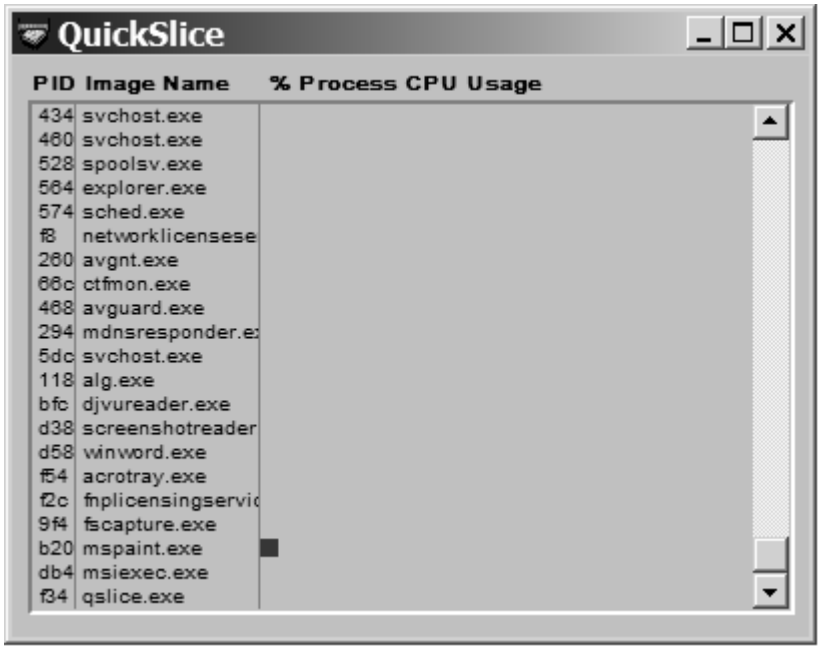
1. Отключите нижнюю секцию, сбросив **View**, **Show Lower Pane**. (Нижняя секция может отображать открытые описатели или проецируемые **DLL** и файлы).
2. Обратите внимание на то, что процессы, являющиеся хостами сервисов, по умолчанию выделяются розовым цветом. Ваши собственные процессы выделяются синим. (Эти цвета можно настроить).
3. Задержите курсор мыши над именем образа и обратите внимание на то, что в подсказке отображается полный путь.
4. Щелкните **View**, **Select Columns** и добавьте путь образа.
5. Отсортируйте по колонке процессов и вы увидите, что представление в виде дерева исчезло. (Вы можете либо вывести представление в виде дерева, либо сортировать по любой из отображаемых колонок.) Снова щелкните для сортировки по алфавиту в обратном порядке (от Z к A). После этого очередной щелчок вернет представление в виде дерева.
6. Сбросьте **View**, **Show Processes From All Users** для отображения только ваших процессов.
7. Перейдите в **Options**, **Difference Highlight Duration** и смените значение на 5 секунд. Потом запустите новый процесс и обратите внимание на то,

что этот процесс выделяется зеленым в течение 5 секунд. Закройте новый процесс и заметьте, что этот процесс выделяется красным в течение 5 секунд, прежде чем исчезнуть из древовидного списка. Эта функция может понадобится для обнаружения создаваемых и завершаемых процессов в системе.

8. Наконец, дважды щелкните какой-нибудь процесс и изучите вкладки, доступные в окне свойств процесса. Эти вкладки понадобятся в дальнейших экспериментах.

ЭКСПЕРИМЕНТ: наблюдение за активностью потоков с помощью QuickSlice, 20

QuickSlice позволяет в динамике наблюдать за соотношением времени, проведенного каждым процессом и **режиме ядра** и в **пользовательском режиме**. На диаграмме красная часть столбца отражает количество процессорного времени в режиме ядра, а синяя в пользовательском режиме. Сумма всех показателей, отображаемых столбцами в окне должна соответствовать **100 % процессорного времени**. Запустите **QiuckSlice**. Например, попробуйте запустить, такое интенсивно использующее графику приложение, как **Paint**. Откройте программу расположив его окно рядом с окном **Paint**, и нарисуйте в **Paint** несколько кривых. В это время вы сможете наблюдать за выполнением Mspaint.exe в окне **QuickSlice** как показано ниже.



Чтобы получить дополнительную информацию о **потоках** процесса, дважды щелкните имя нужного процесса или соответствующий цветной столбик на диаграмме. Вы увидите список потоков этот процесса и относительное процессорное время, используемое каждым потоком (в рамках процесса, а не всей системы).

ЭКСПЕРИМЕНТ: режим ядра и пользовательский режим, 21

С помощью оснастки **Performance** вы можете выяснить, сколько времени ваша система работает в режиме ядра и в пользовательском режиме.

Запустите оснастку **Performance** (**Производительность**), открыв меню **Start** (**Пуск**) и последовательно выбран команды **Programs** (**Программы**), **Administrative Tools** (**Администрирование**), **Performance** (**Производительность**).

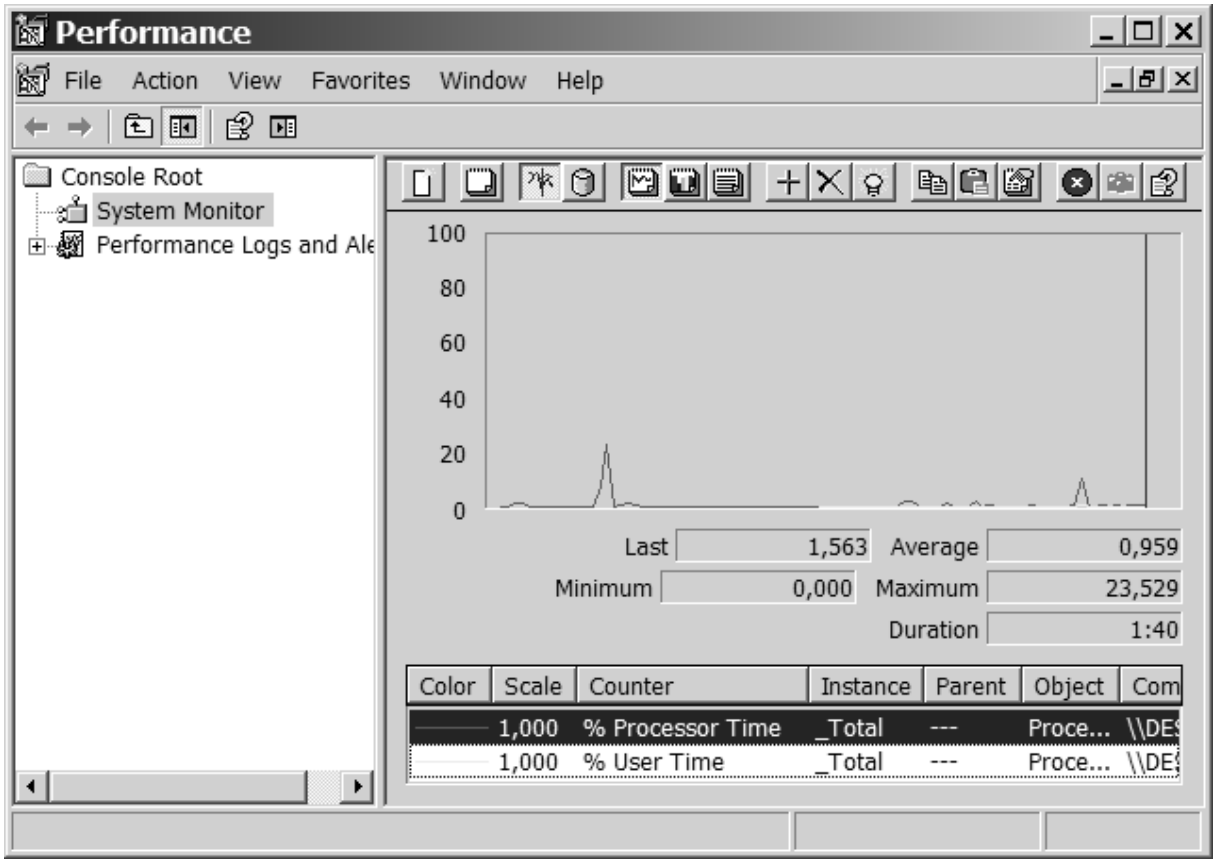
Щелкните на панели инструментов кнопку **Add** (**Добавить**).

Выберите в списке объект **Processor** (**Процессор**), щелкните счетчик **% Privileged Time** (**% работы в привилегированном режиме**) и, удерживай клавишу **Ctrl** в нажатом состоянии, щелкните счетчик **% User Time** (**% работы в пользовательском режиме**).

Щелкните кнопку **Add** (**Добавить**), а затем **Close** (**Закрыть**).

Быстро подвигайте мышью. При этом вы должны заметить всплеск на линии **% Privileged Time**, который отражает время, затраченное на обслуживание прерываний от мыши, и время, понадобившееся подсистеме поддержки окон на отрисовку графики (эта подсистема работает преимущественно как драйвер устройства в режиме ядра).

Закончив, щелкните на панели инструментов кнопку **New Counter Set** (**Новый набор счетчиков**) (или просто закройте оснастку).

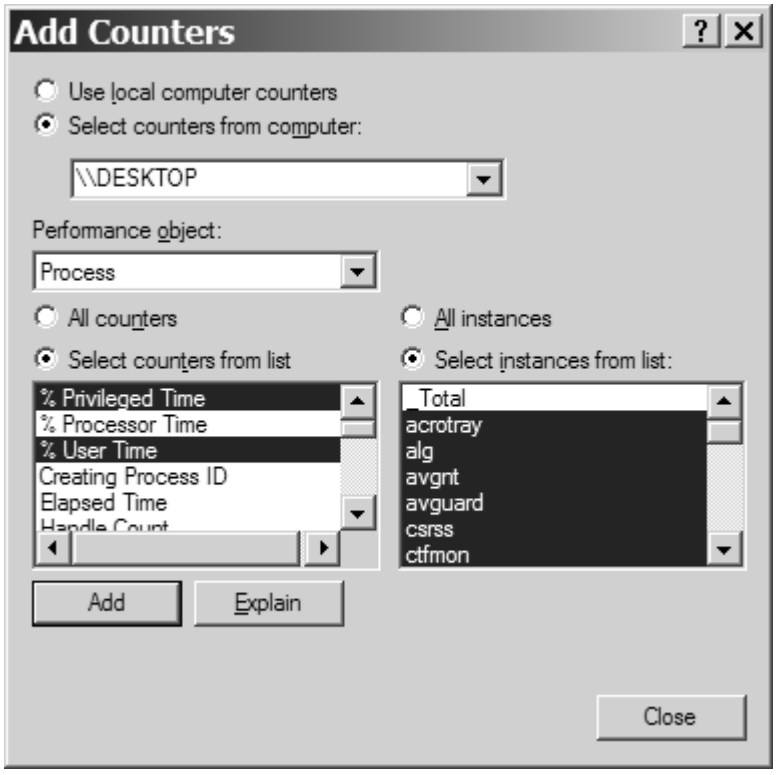


*Оснастка **Производительность**, показывающая, как распределяется время работы процессора между двумя режимами – пользовательским и ядра*

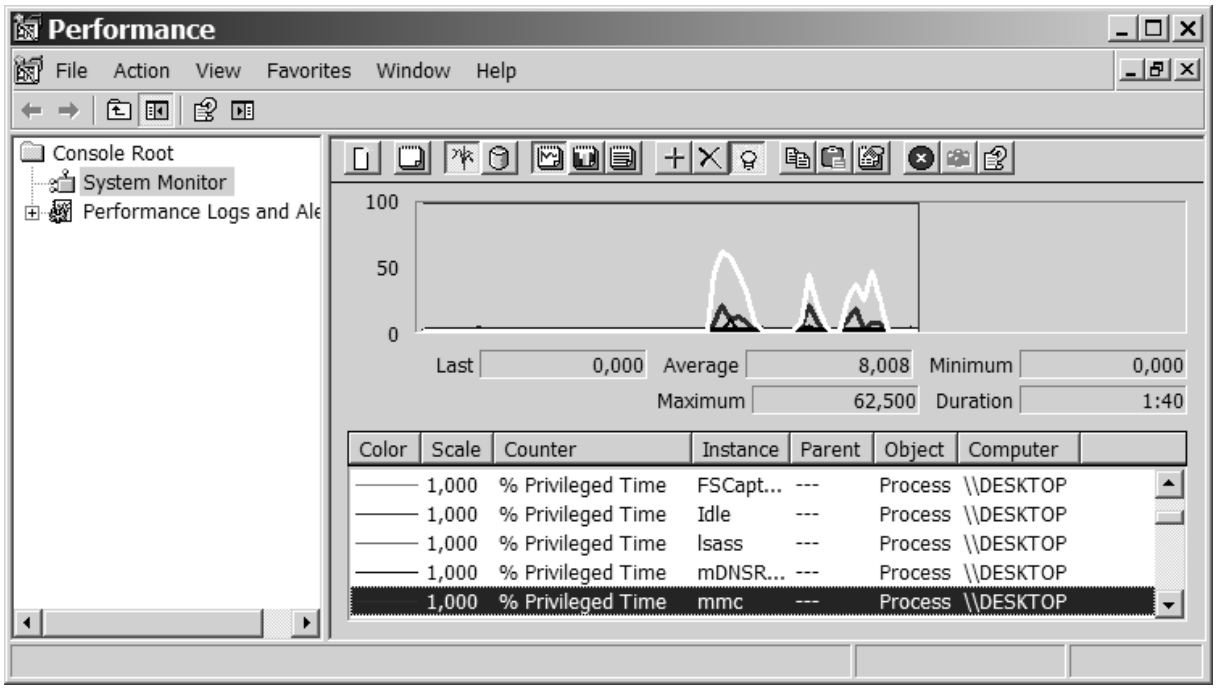
За той же активностью можно понаблюдать через **Task Manager** (**Диспетчер задач**). Просто перейдите в нем на вкладку **Performance** (**Быстродействие**), а затем выберите из меню **View** (Вид) команду **Show Kernel Times** (**Вывод времени ядра**). Процент загруженности процессора отражается зеленым цветом, а процент времени работы в режиме ядра – красным.

Чтобы увидеть, как сама оснастка **Performance** использует время в двух режимах, запустите ее снова, но добавьте те же счетчики для объекта **Process** (**Процесс**).

1. Если вы закрыли оснастку **Performance**, снова запустите ее. (Если она уже работает, откройте новый экран, щелкнув на панели инструментов кнопку **New Counter Set**).
2. Щелкните кнопку **Add** на панели инструментов.
3. Выберите в списке объект **Process**.
4. Выберите счетчики **% Privileged Time** и **% User Time**.
5. В списке экземпляров объекта выберите все процессы (кроме процесса **_Total**).
6. Щелкните кнопку **Add**, а затем **Close**.
7. Быстро подвигайте мышью.
8. Нажмите комбинацию клавиш **Ctrl+H** для активизации режима выделения – текущий выбранный счетчик будет выделен цветом.
9. Прокрутите список всех счетчиков в нижней части окна оснастки, чтобы определить процессы, потоки которых выполнялись при перемещении мыши, и обратите внимание на то, в каком режиме они выполнились – пользовательском или ядра.



Вы должны заметить, как значения счетчиков для процесса оснастки **Performance** – ищите **mmc** в колонке **Instance** (**Экземпляр**) – резко увеличивается при перемещении мыши, поскольку код приложения выполняется в пользовательском режиме, а вызываемые им Windows-функции – в режиме ядра. Вы также заметите, что при перемещении мыши увеличивается активность работы в режиме ядра потока процесса **csrss**. Он представляет поток необработанного ввода (**raw input thread**) подсистемы Windows, принимающий ввод от клавиатуры и мыши и передающий его процессу, к которому он подключен. Наконец, процесс с именем **Idle**, потоки которого, как вы убедитесь, тратят почти **100 %** своего времени в режиме ядра, на самом деле не является процессом. Это **лжепроцесс**, используемый для учета тактов процессора в состоянии простоя. Таким образом, когда Windows нечего делать, она предастся этому занятию в режиме ядра.


Performance
File Action View Favorites Window Help
Console Root
System Monitor
Performance Logs and Ale
100
50
0
Last 0,000 Average 8,008 Minimum 0,000
Maximum 62,500 Duration 1:40
Color Scale Counter Instance Parent Object Computer
1,000 % Privileged Time FSCapt... --- Process \\DESKTOP
1,000 % Privileged Time Idle --- Process \\DESKTOP
1,000 % Privileged Time lsass --- Process \\DESKTOP
1,000 % Privileged Time mDNSR... --- Process \\DESKTOP
1,000 % Privileged Time mmc --- Process \\DESKTOP

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое процесс?
2. Какие состояния может принимать процесс?
3. Что такое родительский процесс?
4. Что означает аббревиатура PPID? Где ее можно встретить?
5. Для чего предназначена программа ProcessExplorer?
6. В чем отличие режима ядра от пользовательского режима?
7. Для чего предназначена программа QuickSlice?
8. Что означают счетчики % Privileged Time и % User Time в оснастке Performance?