телеметрия Server'a 2003:
или метрология счетчиков производительности

крис касперски, ака мыщъх, a.k.a. nezumi, a.k.a. souriz, a.k.a. elraton, no-email

начиная с самых ранних версий в NT был заложен мощный механизм мониторинга, предназначенный для сбора и анализа информации о "жизнедеятельности" системы в реальном времени, именуемый счетчиками производительности (performance counters). это словно приборная панель со множеством датчиков в автомобиле или даже авианосце. вот только "датчиков" у NT на порядок больше и далеко не каждый знает для чего они предназначены и как их можно использовать для выявления "узких" мест в системе и настройки сервера на максимальную производительность

введение

Вполне типичная ситуация — еще вчера сервер работал нормально, а сегодня он тормозит не по детски, дрыгает жестким диском, поглощает огромное количество памяти, гоняя по сетке совершенно левый мусор, занимающий практически всю пропускную способность локальной сети. Возможно, на сервере завелся злобный вирус или хакеры решили устроить DDoS атаку, или какой-то процесс поехал крышей. Но как выяснить — какой из них?!

Опытному администратору достаточно лишь бросить взгляд на счетчики производительности, чтобы проанализировать оперативную обстановку и сделать оргвыводы, убив дурной процесс или даже написав несложный скрпип, делающий это автоматически (что особенно полезно в тех случаях, когда пользователям разрешается запускать на сервере свои программы).

Счетчики производительности так же полезны при планировании аппаратной конфигурации и апгрейде сервера: достаточно ли оперативной памяти, мощности процессора, быстродействия дисковой подсистемы, ect. Вслепую не так-то просто определить где происходит затык и какой именно компонент в наибольшей степени ответственен за производительность. А вот собрав данные со счетчиков производительности за некоторое время (например, за день или неделю) мы сможем с абсолютной точностью установить, что происходит внутри сервера. О том, как именно это сделать, мы сейчас и поговорим.

Говорить мы будем об английской версии Server 2003 (другой в распоряжении мыщъх'а просто нет), приводя "родные" названия счетчиков. Пользователям русской версии Server 2003 предлагается сыграть в увлекательную игру "как, черт возьми, это называлось в оригинале?!", но читатели у нас умные и к таким передрягам им не привыкать, тем более, что большинство названий переведены правильно и обратный перевод большой проблемы не представит.

счетчики производительности в диспетчере задач

Диспетчер Задач, вызываемый по <ALT-CTRL-DEL> или <CTRL-SHIFT-ESC>, позволяет отображать пару десятков важнейших счетчиков производительности для каждого из процессов (вкладка "Processes", см. рис. 1). Добавление/удаление счетчиков осуществляется через "View → Select Columns". К наиболее полезным счетчикам относятся загрузка ЦП ("CPU Usage") и объем виртуальной памяти, потребляемой процессом ("Memory Usage"), что позволяет быстро находить процессы, жрущие ЦП и память.

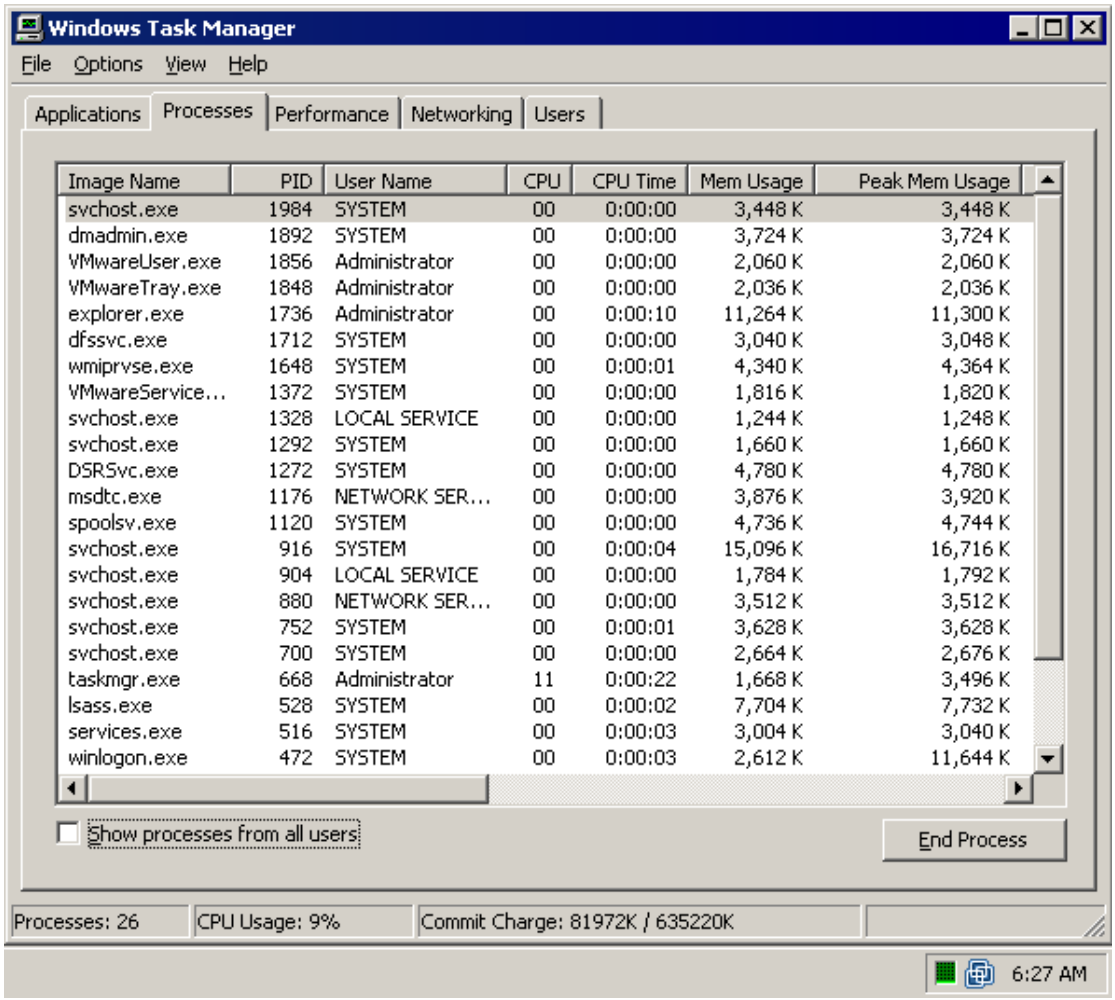

Рисунок 1 Диспетчер Задач, вкладка Processes

Вкладка "Performance" (см. рис. 2) рисует красивые графики загрузки ЦП и общего количества выделенной памяти, что вместе с другой полезной информацией позволяет оценить среднюю/мгновенную нагрузку на сервер.

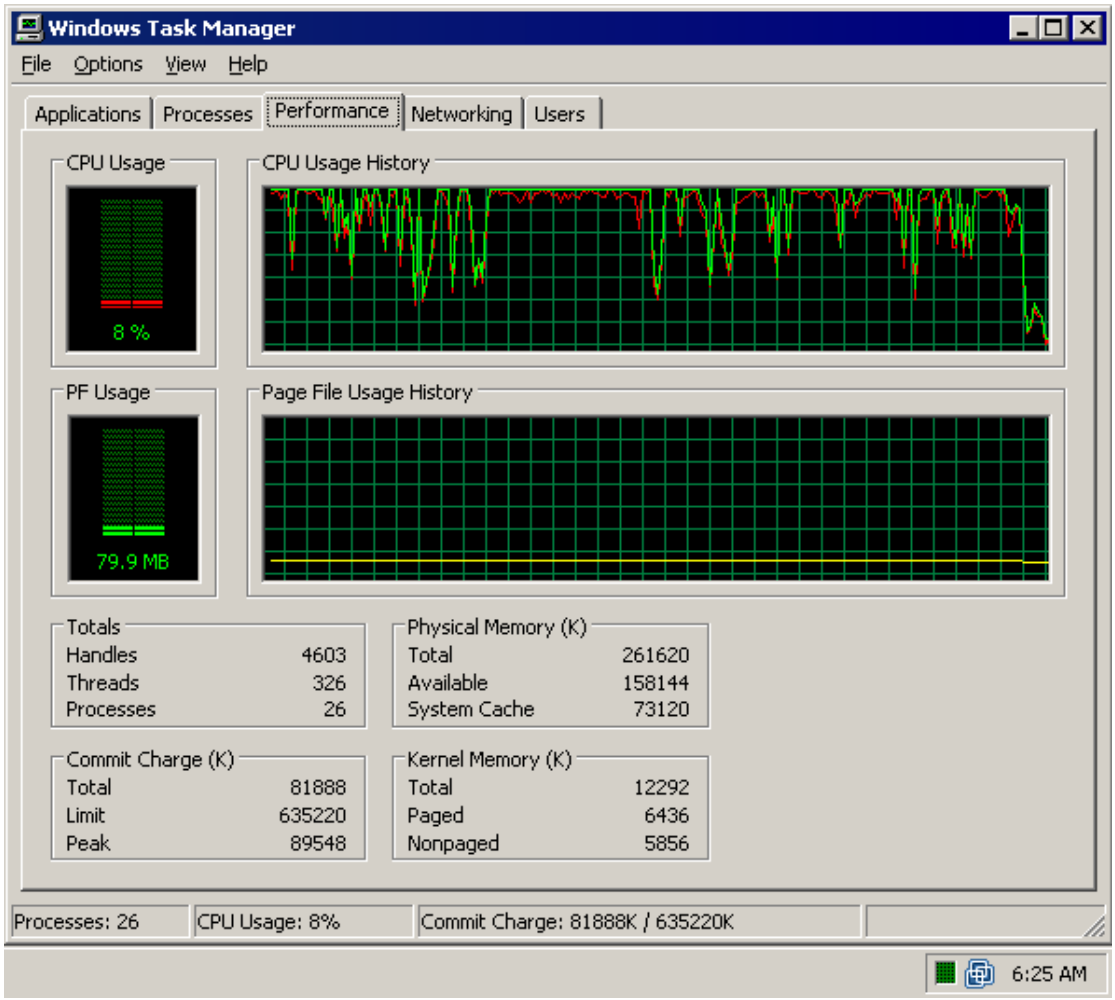

Рисунок 2 Диспетчер Задач, вкладка "Performance"

Соответственно, на вкладке "Networking" можно найти график общей загруженности сети по всем интерфейсам (см. рис. 3).

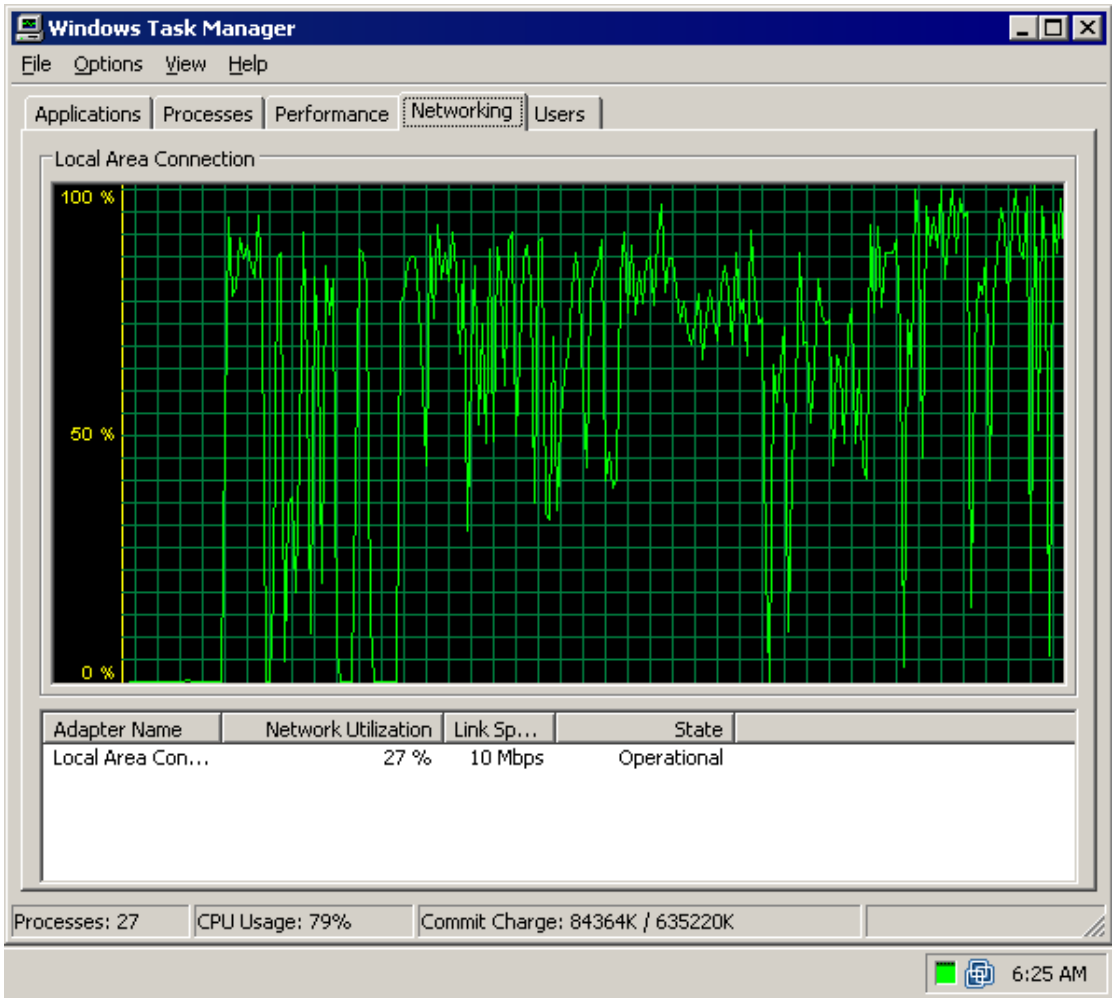

Рисунок 3 Диспетчер Задач, вкладка "Networking"

счетчики производительности в консоли управления

"Монитор Производительности", вызываемый по "start PERFMON.MSC" из командой строки или через "Start → Administrative Tools → Performance", представляет собой основное средство для работы со счетчиками производительности, отображая их значение виде графика (см. рис. 4), диаграммы (см. рис. 5) или таблицы (см. рис. 6).

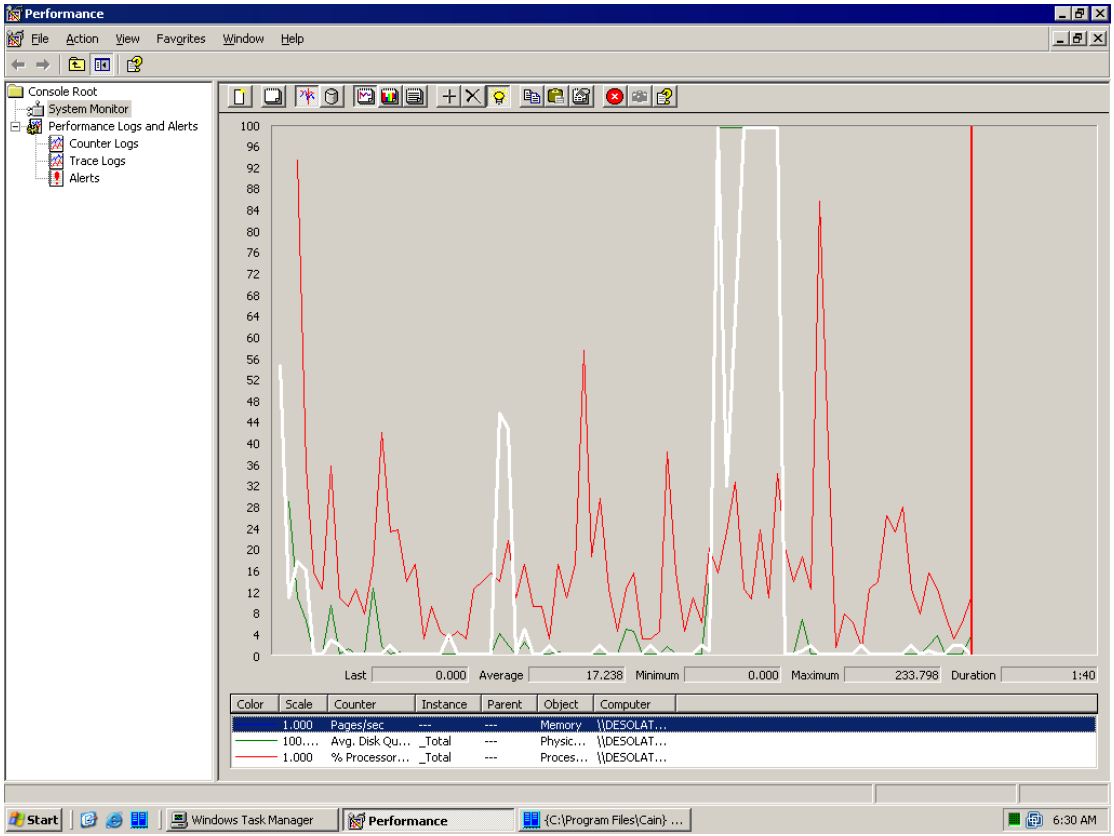

Рисунок 4 Монитор Производительности, отображение счетчиков в виде графика

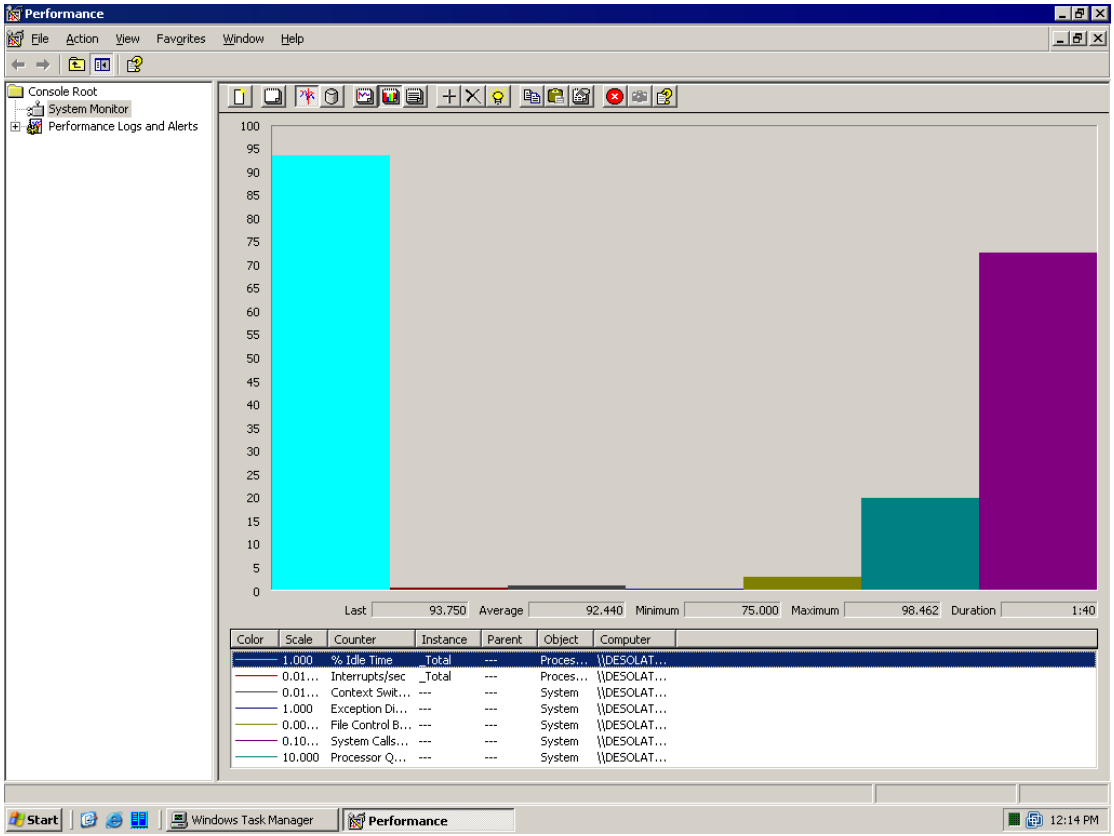

Рисунок 5 Монитор Производительности, отображение счетчиков в виде диаграммы

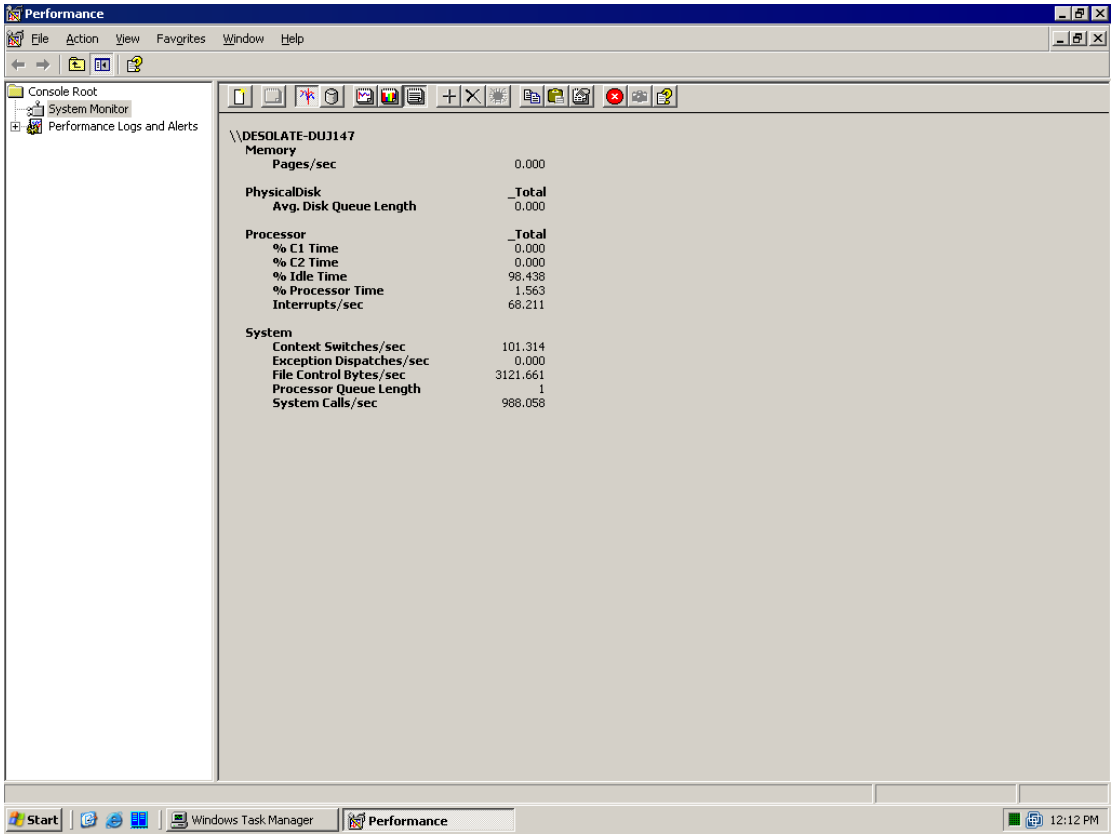

Рисунок 6 Монитор Производительности, отображение счетчиков в виде таблицы

Так же имеется возможность записи истории значений в лог-файл, создания истории событий или определенной реакции на достижения счетчиком порогового значения (например, запуска программы или отправке уведомления администратору). Все эти действия осуществляются через ветвь "Performance Logs-n-Alerts", расположенную в левом окне. Интуитивно понятный интерфейс не создает никаких проблем и "Монитор Производительности" осваивается за несколько минут даже без чтения документации.

обзор важнейших счетчиков производительности

Счетчиков производительности — тысячи и рассказать о каждом из них нет никакой возможности. Некоторую информацию можно нарыть в базе знаний и MSDN, однако, Microsoft дает лишь формальное описание, не вдаваясь подробности. Между тем, интерпретация показаний — дело непростое и тут есть множество тонкостей, незнание которых приводит к грубым ошибкам.

Мыщъх отобрал с десяток важнейших счетчиков, растерзав их в пух и прах до самого основания.

Processor: % Processor Time:

Отображает загрузку процессора в процентах, но мало кто из администраторов задумается в процентах _от_ _чего_?! Объясняю. Каждому потоку отпускается определенный квант времени, по истечении которого, планировщик принудительно отберет у него управление, передавая его потоку, ближе всех стоящего в очереди (планировка очереди потоков — отдельная песня и в различных системах она реализована по разному). Однако, поток может вернуть неизрасходованный остаток кванта, обратившись к планировщику, например, через API-функцию Sleep(0).

Счетчик "Processor Time" на самом деле измеряет не загрузку процессора как таковую, а готовность планировщика предоставить управление потоку по первому требованию. Правильно спроектированная программа практически не загружает процессор, даже если занимается такими "тяжелыми" операциями как сжатие, криптография, etc. А кривая программа показывает 100% загрузку независимо от мощности процессора (для этого ей достаточно просто войти в длинный цикл и баста).

Таким образом, 100% загрузка ЦП указывает на наличие одной или нескольких кривых программ, тормозящих систему даже на очень мощном ЦП. Переход на более мощный ЦП не решает проблемы и тормоза остаются. В таком случае необходимо найти процесс, который грузит ЦП (удобнее всего это делать через "Диспетчер Задач" или систему Alert "Монитора Производительности) и убить его, после чего деинсталлировать соответствующее ему приложение и воспользоваться более корректно написанным программным пакетом. Если же это невозможно, остается либо мириться с тормозами или приобрести многопроцессорную машину, тогда загрузка со 100% сразу упадет до 50%.

Кстати говоря, на HT-процессорах, 50% загрузка равносильна 100%, поскольку, загрузка одного виртуального процессора, парализует работу другого со всеми вытекающими отсюда последствиями.

В нормальных же условиях, средняя загрузка процессора не должна превышать 85%, в противном случае, необходимо либо наращивать частоту процессора/системной шины/оперативной памяти или же увеличивать количество самих процессоров, естественно, убедившись, что мы имеем дело с реальной, а не липовой загрузкой ЦП, вызванной кривым ПО, а убедиться в этом очень просто — слегка тормозим процессор (большинство современных матерей позволяют это делать налету) и смотрим, если загрузка не изменилась, значит, виновато ПО и ЦП тут совсем не при чем.

System: Processor Queue Length

Длина очереди потоков, простаивающих в ожидании процессора. Естественно, чем длиннее очередь — тем ниже производительность. Если среднестатистическая длина очереди превышает 10 потоков, имеет смысл задуматься о добавлении новых процессоров в систему, чтобы увеличить производительность. Более дешевое решение — увеличить таковую частоту, чтобы очередь потоков продвигалась быстрее, а ее длина, соответственно, сокращалась. Однако, это работает только в тех случаях, когда все программы делятся неиспользованными квантами времени, в противном случае длина очереди останется прежней, поскольку длительность кванта не зависит от тактовой частоты.

Server Work Queues: Queue Length

Рабочая очередь сервера. Чем короче — тем лучше. Если длина очереди >4 запросов, сервер начинает реально тормозить, время отклика увеличивается и клиенты начинают чувствовать себя крайне некомфортно.

Узким местом может быть и производительность дисковой подсистемы, и недостаточная частота (или количество) процессоров, и недостаточное количество оперативной памяти, поэтому однозначных рекомендаций по устранению этой проблемы, увы, не существует.

Memory: Page Faults/Sec

Количество отказов страниц в секунду. "Отказом страницы" называется ситуация, когда процессор обращается к странице памяти, которая еще не выделена или вытеснена на диск. В некоторых руководствах встречается утверждение, что при избытке оперативной памяти отказов страниц вообще не возникает, но это не более чем расхожее заблуждение.

При запуске исполняемого файла (подключении динамической библиотеки) система грузит его в память не сразу, а по потребности. Частями. При первом обращении к странице (длина которой в большинстве случаев равна 4 Кбайтам), возникает fault и система считывает кусочек файла в память.

Память под стек так же выделяется не сразу. На вершине стека располагается специальная "сторожевая" страница, при обращении к которой генерируется "отказ", указывающий на необходимость выделения дополнительного стекового пространства.

Таким образом, наличие отказов страниц — не только нормальное, но и вообще _неизбежное_ явление, которое может быть вообще не связано с файлом подкачки, однако, если в секунду наблюдается пять или более отказов, то _скорее_ _всего_ мы имеем дело с хроническим недостатком оперативной памяти, увеличение объема которой позволит существенно увеличить производительность. Как вариант, можно оптимизировать файл подкачки, но об этом ниже.

Memory: Pages/Sec

Интенсивность обмена с файлом подкачки (страниц в секунду). Сюда же входят файлы, проецируемые в память (с которыми работают некоторые программы) и сами исполняемые файлы и динамические библиотеки, трактуемые как файл подкачки, доступный только на чтение. Следовательно, даже если отключить файл подкачки, установив его размер в ноль, данный счетчик все равно продолжит упорно работать ;) Так что трактовать его показания нужно с умом.

Если интенсивность обмена с файлом подкачки превышает 10 страниц в секунду, для увеличения производительности необходимо либо добавить оперативной памяти, либо (на худой конец) перенести файл подкачки на отдельный диск, желательно подключенный к другому IDE-контроллеру (по умолчанию файл подкачки располагается на системном диске, что не есть хорошо).

При интенсивности обмена в 60 и более страниц в секунду, рекомендуется использовать программный или аппаратный RAID уровня 0 (Server 2003 поддерживает программный RAID, только по маркетинговым соображениям именует его "динамическим диском" типа stripped). Чем больше дисков мы задействуем, тем быстрее будет происходить обмен данными. Как минимум необходимо выделить один диск для каждых 60 страниц. То есть, при интенсивности обмена в 180 страниц в секунду, нам необходимо как минимум три диска, на которых будет размещен _только_ файл подкачки и больше ничего. Однако, производительность все равно будет оставаться низкой до тех пор, пока мы не установим дополнительную оперативную память, так что RAID-массив можно рассматривать лишь как временное решение проблемы. Исключения составляют случаи, когда требуемое количество оперативной памяти просто не поддерживается железом (точнее ее поддержка обходится через чур дорого) и лучше мириться с низкой производительностью, чем вкладывать огромные средства в быстродействие.

Memory: Available Bytes

Количество доступной физической памяти в байтах. Во многих руководствах утверждается, что если физической памяти нет, значит, все уходит в своп и мы имеем тормоза, а если наличествует хотя бы несколько десятков мегабайт, значит, физическая память еще не исчерпана, подкачка не используется и сервер шурует с крейсерской скоростью. На самом деле это _очень_ большое упрощение.

Допустим, мы имеем 1 Гбайт RAM, а потребности сервера составляют 10 Гбайт. Вопрос: какое количество физической памяти покажет счетчик? Ответ: *возможно*, и ноль байт, но **крайне маловероятно**. Предположим, что процесс освободил 10 Мбайт используемой памяти (например, потому что от сервера отключился клиент). Если вся эта память размешалась в RAM, то количество свободной физической памяти увеличится на 10 Мбайт, при том, что ~9 Гбайт будут болтаться в файле подкачки. Если система интенсивно выделяет/освобождает большое количество памяти, то показания данного счетчика могут достигать 25% и более от общего объема физической памяти, но это еще не значит, что памяти достаточно и нужно смотреть на количество обращений к файлу подкачки, что описано выше.

Memory: Committed Bytes

Общее количество выделенной памяти в байтах. Если оно превышает объем физической памяти, часть страниц вытесняется в файл подкачки, что _обычно_ приводит к снижению производительности. Если количество выделенной памяти не превышает размер физической, то сервер летает на форсаже и никаких особых комментариев тут не требуется. Но это идеализированная ситуация и в реальной жизни памяти сплошь и рядом оказывается недостаточно, но сам по себе объем выделенной памяти ни о чем не говорит!!! Вытеснение редко используемого кода/данных в своп практически не снижает производительности и тут опять-таки нужно смотреть на интенсивность обмена с файлов подкачки.

PhysicalDisk: Current Disk Queue Length

Длинна очереди запросов на чтение/запись к физическому диску. Чем короче — тем лучше. Если в очереди постоянно находится два и более запросов, то это нехорошо и для увеличения производительности рекомендуется обзавестись программным или аппаратным RAID'ом или использовать более скоростные диски.

Как вариант, можно реорганизовать размещение программ и данных, распределив их по разным разделам или просто запустить дефрагментатор.

PhysicalDisk: % Disk Time

Время занятости диска, в течении которого он обрабатывал запросы на чтение/запись, в процентах. Если загруженность диска достигает 100%, то образуется конкретный затор, требующий перехода на RAID-массивы, использования более быстродействующих винчестеров или дефрагментации. Загрузка менее 80% считается вполне допустимой.

>>> вставка мониторинг локальных дисков в W2K

В Windows Server 2003/Server 2008 счетчик производительности "LogicalDisk" включен по умолчанию, однако в W2K и XP он по умолчанию включен и чтобы заставить его давать показания необходимо выполнить следующую команду "diskperf -yv" и перезагрузиться.

LogicalDisk: % Free Space

Объем свободного дискового пространства в процентах. Если диск заполнен на 80% и более, файловая система NTFS в силу своих конструктивных особенностей начинает конкретно тормозить, а если свободного пространства остается менее 10%, происходит необратимая фрагментация $MTF-файла, хранящего данные обо всех остальных файлах на диске. То есть, если диск хотя бы однажды окажется заполненным на >90%, то рекомендуется скопировать данные на другой носитель, отформатировать его и вернуть данные обратно. Или, как вариант, установить в "Мониторе Производительности" Alert на этот счетчик и при заполнении диска на 80% начать удалять временные файлы, кэш, или оправлять SMS администратору, чтобы тот разрулил ситуацию.

Network Interface: Bytes Total/sec

Загруженность сетевого интерфейса в байтах в секунду. Чем ближе она подбирается к его пропускной способности, тем это хуже для пользователей. К сожалению, в такой ситуации очень мало, что можно предпринять (переход со 100 Мбитного Ethernet'a на 1 Гбитный — не предлагать). Разве что пересмотреть политику документооборота, например, перенести часть файлов с сервера на рабочие станции или установить еще один сервер, но это уже требует серьезных вложений.

Network Interface: Output Queue Length

Длина очереди запросов к сетевому интерфейсу. В идеале, никакой очереди не должно быть, но 1-2 запроса считаются вполне приемлемыми, а вот дальнейший рост очереди вызывает ощутимое падение производительности. Причиной может быть как недостаточная пропускная способность сетевых каналов, так и медленная обработка запросов на сервере, обусловленная недостаточной мощностью процессора, нехваткой памяти, и т.д. Так что универсальных решений тут нет и нужно смотреть на остальные счетчики производительности, описанные выше.

заключение

Работа со счетчиками производительности требует глубоких знаний об устройстве операционной системы и слепое следование рекомендациям обычно приводит к неоправданному наращиванию аппаратных мощностей, а сервер все равно продолжает тормозить. Обидно? Обидно! Но что поделаешь… Интерпретация показаний счетчиков производительности редко бывает однозначна и прежде, чем принимать какое-то решение, рекомендуется проштудировать "Внутреннее устройство Windows" Руссиновича и "Современные операционные системы" Таненбаума.

>>> врезка полезные ссылки

- **Windows Server 2003 Performance Counters Reference**:

- "официальное" описание счетчиков производительности от Microsoft с методикой поиска "узких мест" в системе (на английском языке): http://technet2.microsoft.com/windowsserver/en/library/3fb01419-b1ab-4f52-a9f8-09d5ebeb9ef21033.mspx?mfr=true;

- **Monitoring Microsoft Windows Server 2003**:
  - неофициальная презентация в формате PowerPoint, посвященная мониторингу счетчиков производительности и тюнингу Server 2003 (на английском языке): http://cs.senecac.on.ca/~rhonda.murdoch/Chapter03.ppt;
- **Performance Counters**:
  - еще одно неофициальное, но довольно подробное описание основных счетчиков производительности Server 2003 (на английском языке): http://www.sysed.com/DnLoads/QualityContents/W2K3Server/99_W2K3_Server_AppE_Performance_Counters.pdf;
- **Счётчики производительности**:
  - статья в двух частях, рассказывающая как создавать свои собственные счетчики производительности и снимать с них показания (на русском языке): http://rsdn.ru/article/baseserv/perfcounters1.xml и http://rsdn.ru/article/baseserv/perfcounters2.xml;