Защита от утечек в Exchange 2013

Windows IT Pro/RE

№5 МАЙ 2017 | WWW.WINDOWSITPRO.RU | ИНФО ДЛЯ ИТ-ПРО

Мобильная версия

App Store

Google play

Анализ ввода-вывода

Анализ системы ввода-вывода SQL Server

Идентификация имен входа при подключении к базе данных по умолчанию в SQL Server

Проверка имен входа

Определяем потерянные учетные данные

ISSN 1563-101X

17005

9 771563 101008

Windows IT Pro/RE
№5 май 2017
12+

Издание для специалистов, интересующихся технологиями компании Microsoft.

Главный редактор: **Д. Торопов** (toropovd@osp.ru)
Ответственный редактор: **Е. Петровичева**
Корректор: **Л. Теременко**
Верстка и дизайн: **О. Шуранова**
Номер также готовили: **Е. Овсянников**
Т. Евдокимова, А. Китаев, А. Федотов,
Н. Басалова, Ю. Власов, Д. Щепкин, А. Адзиев

Адрес для писем: 123056, Москва, а/я 82
Телефоны: (495) 725-4780/83, (499) 703-1854
Факс: (495) 725-4783
E-mail: windowsitpro@osp.ru

© 1999-2017 Издательство «Открытые системы»
© 1999-2017 Penton Media, Inc.

Свидетельство о регистрации средства массовой информации ПИ №ФС 77-63737 от 16 ноября 2015 г.
Выдано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзором).

Цена свободная. Выходит 12 раз в год.

ОТКРЫТЫЕ СИСТЕМЫ
Open Systems Publications

Учредитель и издатель:
ООО «Издательство «Открытые системы»
127254, Москва, пр-д Добролюбова, д. 3, стр. 3, каб. 13.
Президент **М. Е. Борисов**
Генеральный директор **Г. А. Герасина**
Директор ИТ-направления **П. В. Христов**
Коммерческий директор **Т. Н. Филина**
Подписные индексы:
Объединенный каталог «Пресса России» — 38185,
«Каталог российской прессы» — 99483,
ФГУП «Почта России» — П2337
Отпечатано в ООО «Богородский полиграфический комбинат»,
142400, Московская обл., г. Ногинск, ул. Индустриальная, д. 40б
Тираж: 6900 экз. — печатная версия,
3280 экз. — PDF-версия

Редакция не несет ответственности за содержание рекламных материалов. Все права защищены. Полное или частичное воспроизведение или размножение каким бы то ни было способом материалов, опубликованных в настоящем издании, допускается только с письменного разрешения ООО «Издательство «Открытые системы».
Windows®, Windows Vista® и Windows Server® — зарегистрированные торговые марки корпорации Microsoft. Название Windows IT Pro используется Penton Media, Inc. в соответствии с соглашением с владельцем торговой марки. Название Windows IT Pro/RE используется ООО «Издательство «Открытые системы» по лицензионному соглашению с Penton Media, Inc. Windows IT Pro/RE — независимое от корпорации Microsoft издание. Корпорация Microsoft не несет ответственности за редакционную политику и содержание журнала.
Редакция оставляет за собой право не вступать в переписку.
Отобранные для публикации письма редактируются в соответствии с терминологическими нормами, принятыми в издательстве.
Названия продуктов и компаний, упомянутых в журнале, могут быть товарными знаками их владельцев.

PENTON
Penton Media, Inc.

Содержание

ТЕМА НОМЕРА

2 Анализ системы ввода-вывода SQL Server. Часть 1
ТИМ ФОРД

8 Идентификация имен входа при подключении к базе данных по умолчанию в SQL Server
ТИМ ФОРД

12 Проверка имен входа
ТИМ ФОРД

15 Определяем потерянные учетные данные
ТИМ ФОРД

SQL SERVER

16 Обнаружение пересечений в интервалах
ИЦИК БЕН-ГАН

23 Динамические объекты управления групп доступности в SQL Server 2016
ТИМ ФОРД

28 Вычисление различий даты и времени по частям
ИЦИК БЕН-ГАН

EXCHANGE & OUTLOOK

31 Защита от утечек в Exchange 2013
АЛЕКСАНДР КУЗНЕЦОВ, ЕКАТЕРИНА ДАНИЛОВА

OFFICE SYSTEM

36 SharePoint и машинное обучение
ЛИАМ КЛИРИ

40 Совместная работа в «облаке» в Office 365
ЛИАМ КЛИРИ

ВВОДНЫЙ КУРС

46 Управление дополнительными параметрами доставки Центра обновления для Windows 10
РИЧАРД ХЭЙ

48 Автоматическая очистка диска в обновлении Windows 10 Creators Update
РИЧАРД ХЭЙ

52 Выключаем подсказки в Windows 10
РИЧАРД ХЭЙ

53 Пакетное переименование файлов в приложении «Проводник» в Windows 10
РИЧАРД ХЭЙ

55 Средства устранения неполадок в обновлении Windows 10 Creators Update
РИЧАРД ХЭЙ

56 Смена мобильных экосистем
ДЕРЕК УОЛТЕР

ИНТЕРНЕТ И ВЕБ-СЛУЖБЫ

59 Office 365 и CDN
ЛИАМ КЛИРИ

СПРОСИ ЭКСПЕРТА

62 На вопросы читателей отвечает Джон Сэвилл

ИЛЛЮСТРАЦИЯ НА ОБЛОЖКЕ ROLFFIMAGES© (FOTOLIA.COM)

Часть 1

Анализ системы ввода-вывода SQL Server

Тим Форд

Microsoft SQL Server располагает системными объектами Dynamic Management Objects, которые предоставляют исчерпывающие метаданные, относящиеся к характеристикам производительности экземпляров SQL Server. В частности, использование одного из них, sys.dm_io_virtual_file_stats, позволяет специалисту в области данных оценить влияние ввода-вывода хранилища данных, структуры файлов и даже плохо спроектированных запросов и гиперактивных конечных пользователей. В предлагаемой статье мы рассмотрим, как определить самые высокие задержки чтения и записи данных и файлов журналов соответственно.

Вечные проблемы

В 2010 году я написал книгу о динамических объектах управления в соавторстве с Луисом Дэвидсоном, обладателем сертификата SQL Server MVP. Теперь, по прошествии нескольких лет, я еще выше ценю возможности, которые открывают эти конструкции как перед администраторами баз данных, так и перед разработчиками при идентификации проблем с производительностью. Данная статья стала результатом споров с аналитиком приложений относительно причин низкой производительности приложений, в которой обычно винят «медленное хранилище данных».

Когда я спросил, на чем основаны предположения моего оппонента, он упомянул как системный монитор, так и средство Performance Analysis of Logs (http://pal.codeplex.com/), более известное как PAL и доступное на сайте CodePlex. PAL — средство, которое читает журналы системного монитора и предоставляет отчеты о производительности на основе этих журналов. Оба инструмента предоставляются компанией Microsoft, и в основе их предупреждений лежат иные представления о порогах производительности ввода-вывода, нежели готова признать Microsoft, или мягкие стандарты «реального мира». PAL выдает предупреждение об операциях чтения длительностью более 20 миллисекунд (мс) и записи более 5 мс. Всеми признано, что приемлемо время менее 100 мс для чтения и 20 мс для записи. Однако кому не захочется обвинить «медленный ввод-вывод», увидев, как PAL выдает показатели ввода-вывода, едва превышающие пороговые значения (см. экран 1).

Среда созрела для повышения производительности как внутренних, так и подготовленных внешним поставщиком запросов, но аналитики хотели только одного: чтобы инженеры, отвечающие за инфраструктуру (сервер, хранилище данных, администраторы баз данных), добавляли оборудование, вместо того чтобы внедрять оптимальные стандарты проектирования и кодирования

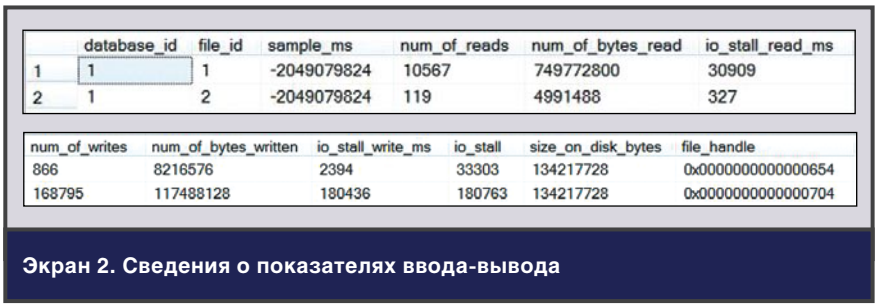
Этот пример для меня не нов. В течение длительного времени данная проблема докучала нашей группе администраторов баз данных. Мы уже выполнили миграцию с хранилища VNX «уровня 2» на платформу хранения данных «уровня 1» (HP 9500 SAN.) Мне также было известно, что приложение CA Service Desk имеет существенные изъяны, и в дополнение к ним два человека в нашей компании направляли в базу данных чрезвычайно неэффективные запросы и никак не реагировали на рекомендации по настройке производительности от администраторов баз данных. Среда созрела для повышения производительности как внутренних, так и подготовленных внешним поставщиком запросов, но аналитики хотели только одного: чтобы инженеры, отвечающие за инфраструктуру (сервер, хранилище данных, администраторы баз данных), добавляли оборудование, вместо того чтобы внедрять оптимальные стандарты проектирования и кодирования.

Медлительность целиком объяснялась тем обстоятельством, что в отчетах PAL задержка ввода-вывода для чтения и записи измеряется миллисекундами. Другими словами, реальные задержки ввода-вывода приемлемы. Но поскольку PAL выделила эти элементы как проблемные, не было смысла обращаться к аналитикам. Я воспользовался своим любимым средством, динамическими объектами управления, чтобы решить задачу с учетом фактов и метаданных, применяя знание математики и личный опыт.

Далее в статье мы рассмотрим способы определения нагрузки ввода-вывода на нескольких экземплярах в средах SQL Server и сопоставления результатов с задержкой ввода-вывода, что позволит сделать заключение о влиянии соотношения задержки и нагрузки на производительность ввода-вывода. Это поможет заняться плохо настроенными запросами, вместо того чтобы терять драгоценное время в погоне за призраками на уровне хранилища. Однако мы будем не спеша приближаться к цели и сначала посмотрим, как измерить нагрузку и задержку ввода-вывода на уровне экземпляра и как использовать некоторые широко распространенные

Condition
Greater than 25 ms physical disk READ response times
Greater than 25 ms physical disk WRITE response times

Экран 1. Показатели ввода-вывода утилиты PAL



	database_id	file_id	sample_ms	num_of_reads	num_of_bytes_read	io_stall_read_ms
1	1	1	-2049079824	10567	749772800	30909
2	1	2	-2049079824	119	4991488	327

num_of_writes	num_of_bytes_written	io_stall_write_ms	io_stall	size_on_disk_bytes	file_handle
866	8216576	2394	33303	134217728	0x0000000000000654
168795	117488128	180436	180763	134217728	0x0000000000000704

Экран 2. Сведения о показателях ввода-вывода

инструменты анализа и подготовки отчетов для представления данных администраторам баз данных и аналитикам приложений. Надеюсь, с помощью этой статьи вы освоите применение различных динамических объектов управления, а также рассматриваемых здесь средств Excel, Tableau и Microsoft Power Tools и статистических функций mean, median и mode.

Базовая функция динамического управления

В центре всей серии цикла материалов, который мы начинаем этой статьей, находится функция динамического управления. Эта функция принимает два параметра:

- database_id;
- file_id

Поскольку мы обычно не запоминаем наше значение database_ids, оно часто преобразуется из имени с использованием системной функции db_id (). Синтаксис довольно очевиден, как и результаты. Например, приведенный ниже

Листинг 1. Сохранение метаданных DMO на диске

```
 CREATE TABLE dbo.database_file_io
(
[server] sysname NOT NULL
, database_name sysname NOT NULL
, logical_name sysname NOT NULL
, physical_name NVARCHAR(260) NOT NULL
, file_size_mb BIGINT NOT NULL
, type_desc NVARCHAR(60) NOT NULL
, num_of_reads BIGINT NOT NULL
, num_kb_read BIGINT NOT NULL
, avg_daily_reads BIGINT NOT NULL
, avg_daily_kb_read BIGINT NOT NULL
, io_stall_read_ms BIGINT NOT NULL
, avg_io_stall_read_ms BIGINT NOT NULL
, num_of_writes BIGINT NOT NULL
, num_kb_written BIGINT NOT NULL
, avg_daily_writes BIGINT NOT NULL
, avg_daily_kb_written BIGINT NOT NULL
, io_stall_write_ms BIGINT NOT NULL
, avg_io_stall_write_ms BIGINT NOT NULL
, total_io_stall BIGINT NOT NULL
, avg_io_stall_ms BIGINT NOT NULL
, pct_read DECIMAL(4,1) NOT NULL
, pct_write DECIMAL(4,1) NOT NULL
, date_stamp DATETIME NOT NULL
)

ALTER TABLE dbo.database_file_io ADD CONSTRAINT
PK_database_file_io PRIMARY KEY CLUSTERED
(
logical_name,
server,
database_name,
date_stamp
)
WITH
(
STATISTICS_NORECOMPUTE = OFF
, IGNORE_DUP_KEY = OFF
, ALLOW_ROW_LOCKS = ON
, ALLOW_PAGE_LOCKS = ON
, FILLFACTOR = 70
) ON [PRIMARY];
GO

--====================================================
 CREATE TABLE dbo.database_file_io_history
(
[server] sysname NOT NULL
, database_name sysname NOT NULL
, logical_name sysname NOT NULL
, physical_name NVARCHAR(260) NOT NULL
, file_size_mb BIGINT NOT NULL
, type_desc NVARCHAR(60) NOT NULL
, num_of_reads BIGINT NOT NULL
, num_kb_read BIGINT NOT NULL
, avg_daily_reads BIGINT NOT NULL
, avg_daily_kb_read BIGINT NOT NULL
, io_stall_read_ms BIGINT NOT NULL
, avg_io_stall_read_ms BIGINT NOT NULL
, num_of_writes BIGINT NOT NULL
, num_kb_written BIGINT NOT NULL
, avg_daily_writes BIGINT NOT NULL
, avg_daily_kb_written BIGINT NOT NULL
, io_stall_write_ms BIGINT NOT NULL
, avg_io_stall_write_ms BIGINT NOT NULL
, total_io_stall BIGINT NOT NULL
, avg_io_stall_ms BIGINT NOT NULL
, pct_read DECIMAL(4,1) NOT NULL
, pct_write DECIMAL(4,1) NOT NULL
, date_stamp DATETIME NOT NULL
);
ALTER TABLE dbo.database_file_io_history ADD CONSTRAINT
PK_database_file_io_history PRIMARY KEY CLUSTERED
(
logical_name,
server,
database_name,
date_stamp
)
WITH
(
STATISTICS_NORECOMPUTE = OFF
, IGNORE_DUP_KEY = OFF
, ALLOW_ROW_LOCKS = ON
, ALLOW_PAGE_LOCKS = ON
, FILLFACTOR = 70
) ON [PRIMARY];
GO
```

Листинг 2. Сценарий сбора данных из sys.dm_io_virtual_file_stats

```
DECLARE @days INT
SELECT @days = DATEDIFF(d, create_date, GETDATE()) FROM sys.
databases WHERE name = 'tempdb'
IF @days = 0
BEGIN
SELECT @days = 1
END
INSERT INTO [dbo].[database_file_io]
(
server
, database_name
, logical_name
, physical_name
, file_size_mb
, type_desc
, num_of_reads
, num_kb_read
, avg_daily_reads
, avg_daily_kb_read
, io_stall_read_ms
, avg_io_stall_read_ms
, num_of_writes
, num_kb_written
, avg_daily_writes
, avg_daily_kb_written
, io_stall_write_ms
, avg_io_stall_write_ms
, total_io_stall
, avg_io_stall_ms
, pct_read
, pct_write
, date_stamp
)
SELECT @@SERVERNAME AS [server]
, DB_NAME(FS.database_id) AS database_name
, DB.[name] AS logical_name
, DB.physical_name
, FS.size_on_disk_bytes/1024/1024 AS file_size_mb
, DB.type_desc
, FS.num_of_reads
, FS.num_of_bytes_read / 8192 AS num_kb_read
, FS.num_of_reads / @days AS avg_daily_reads
, FS.num_of_bytes_read / 8192 / @days AS avg_daily_kb_read
, FS.io_stall_read_ms
, CASE
WHEN FS.num_of_reads = 0 THEN FS.io_stall_read_ms
ELSE FS.io_stall_read_ms / FS.num_of_reads
END AS avg_io_stall_read_ms
, FS.num_of_writes
, FS.num_of_bytes_written / 8192 AS num_of_kb_written
, FS.num_of_writes / @days AS avg_daily_writes
, FS.num_of_bytes_written / 8192 / @days AS avg_daily_kb_written
, FS.io_stall_write_ms
, CASE
WHEN FS.num_of_writes = 0 THEN FS.io_stall_write_ms
ELSE FS.io_stall_write_ms / FS.num_of_writes
END AS avg_io_stall_write_ms
, FS.io_stall AS total_io_stall
, CASE
WHEN (FS.num_of_reads + FS.num_of_writes) = 0 THEN FS.io_stall
ELSE FS.io_stall / (FS.num_of_reads + FS.num_of_writes)
END AS avg_io_stall_ms
, CAST(100. * FS.num_of_reads / (FS.num_of_reads + FS.num_of_writes)
AS decimal(4,1)) AS pct_read
, 100.0 - (CAST(100. * FS.num_of_reads / (FS.num_of_reads + FS.num_
of_writes) AS decimal(4,1))) AS pct_write
, GETDATE() AS date_stamp
FROM sys.dm_io_virtual_file_stats(NULL, NULL) FS
INNER JOIN master.sys.master_files DB
ON FS.database_id = DB.database_id
AND FS.file_id = DB.file_id
ORDER BY 3
, 1
, 2;
```

программный код возвращает информацию обо всех файлах, связанных с базой данных master:

```
SELECT *
FROM sys.dm_io_virtual_file_stats
  (DB_ID ('master'), NULL);
```

Возвращенные столбцы содержат сведения, относящиеся к файлу и базе данных (database_id и file_id), количество времени в образце в миллисекундах (sample_ms), а затем показатели ввода-вывода для операций обоих типов, количество прочитанных байтов и замедление ввода-вывода в миллисекундах как для чтения, так и для записи (см. экран 2). Динамическое административное представление (DMV) также предоставляет общее замедление ввода-вывода для операций чтения и записи, текущий размер файла и file_handle, уникальный идентификатор для файла в экземпляре SQL. Обратите внимание, что sample_ms в этих выходных данных является отрицательным числом. Это значение нельзя использовать как показатель непрерывной работы, так как оно возвращается к исходному после достижения предела для этого типа данных.

	server	database_name	physical_name	avg_daily_reads	avg_io_stall_read_ms
1	[redacted]	WebAnalyticsSer...	D:\MSSQL\Data\...	55468	5609
2	[redacted]	WebAnalyticsSer...	D:\MSSQL\Data\...	51620	5321
3	[redacted]	WebAnalyticsSer...	D:\MSSQL\Data\...	48974	5278
4	[redacted]	WebAnalyticsSer...	D:\MSSQL\Data\...	54320	5239
5	[redacted]	WebAnalyticsSer...	D:\MSSQL\Data\...	51329	5064

Экран 3. Три файла с худшими показателями задержки

В процессе подготовки этого материала я заинтересовался идентификаторами файлов database_id и file_id, а также столбцами нагрузки ввода-вывода, в частности num_of_reads, io_stall_read_ms, num_of_writes и io_stall_write_ms. С помощью этих столбцов можно определить среднюю задержку ввода-вывода для операций чтения и записи и назначить ее любому файлу в любой базе данных.

Необходимо помнить, что эта информация изменчивая и накопительная. Метаданные DMO не сохраняются между перезапусками службы. В случае с этой функцией DMF метаданные также

Листинг 3. Запрос на определение файлов с самой высокой средней задержкой чтения при вводе-выводе

```
SELECT TOP 5 FIO.server, FIO.database_name, FIO.physical_name, FIO.avg_daily_reads, avg_io_
stall_read_ms
 FROM dbo.database_file_io FIO
 WHERE FIO.type_desc = 'rows' AND FIO.database_name  NOT IN ('tempdb', 'model', 'msdb',
'master', 'iDBA')
 ORDER BY avg_io_stall_read_ms DESC;
```

Листинг 4. Пять файлов журналов с самой высокой средней задержкой записи при вводе-выводе

```
SELECT TOP 5 FIO.server, FIO.database_name, FIO.physical_name, FIO.avg_daily_writes, avg_io_
stall_write_ms
  FROM dbo.database_file_io FIO
 WHERE FIO.type_desc = 'log' AND FIO.database_name NOT IN ('tempdb', 'model', 'msdb', 'master',
'iDBA')
 ORDER BY avg_io_stall_write_ms DESC;
```

Листинг 5. Получение показателей чтения для искомой базы данных

```
WITH Results AS (SELECT ROW_NUMBER()
 OVER (ORDER BY (FIO.avg_io_stall_read_ms) DESC) AS latency_rank, ROW_NUMBER()
 OVER (ORDER BY (FIO.avg_daily_reads) DESC) AS io_load_rank, FIO.server, FIO.database_name,
FIO.physical_name, FIO.avg_daily_reads, FIO.avg_io_stall_read_ms
 FROM dbo.database_file_io FIO
 WHERE FIO.type_desc = 'rows' AND FIO.database_name NOT IN ('tempdb', 'model', 'msdb',
'master', 'iDBA'))
SELECT Results.latency_rank, Results.io_load_rank, Results.physical_name, Results.avg_io_stall_
read_ms, Results.avg_daily_reads
 FROM Results
 WHERE Results.server = '<server_name,,>' AND Results.database_name = '<db_name,,>'
```

со временем накапливаются. Если вы хотите получить эталонный показатель нагрузки ввода-вывода между двумя точками во времени или получить информацию о трендах для базовых уровней, то необходимо сохранить информацию на диске. Именно это и будет показано далее.

Сохранение метаданных DMO на диске

Процесс сохранения чрезвычайно прост: создайте таблицу с похожей схемой для собираемых данных, добавьте столбец date_stamp и настройте рутинный сбор информации, как показано в листинге 1.

Я подготовил две таблицы, так как намеревался установить 90-дневный базовый уровень. Я собирался ежедневно перемещать данные из базовой таблицы в таблицу журнала, а затем усекать ежедневную таблицу и удалять из таблицы журнала все данные старше 90 дней.

Я немного расширил коллекцию, включив вычисленные столбцы для расчета дневных значений ввода-вывода, а также изменения единиц измерения от байтов. Я добавил эти столбцы, поскольку в дальнейшем планировал расширить оценку ввода-вывода и при сохранении данных от наших 150+ серверов SQL Server всего лишь за 90 дней не беспокоился о пространстве на диске.

Где были созданы эти таблицы? Вы можете сделать это локально на каждом SQL Server, но я покажу, как собрать информацию для всех баз данных вашего домена для сравнения в масштабах среды. Если вы хотите отыскать шумные серверы и базы данных — скрипучие колеса, сделать это можно следующим образом. У меня есть центральная база данных, именуемая «спасательной лодкой» (lifeboat), и это действительно спасательная лодка в случаях, когда требуется быстро найти ответы на разнообразные вопросы.

Теперь, имея целевые таблицы, настало время представить сценарий сбора, который будет получать данные из sys.dm_io_virtual_file_stats (и некоторых других системных объектов) для заполнения этой базовой таблицы ежедневного сбора данных (листинг 2).

Заметьте, что я расширил некоторые описательные столбцы для базы данных и файла присоединением к системному представлению sys.master_files. Я также рассчитал количество дней, в течение которых ведется сбор данных из DMF с помощью create_date для tempdb. В конечном итоге мы объединим данные из этого экземпляра SQL со всеми остальными данными в нашей среде, которые будут иметь разное время работы, поэтому единственный способ корректного сравнения — на дневной основе.

В завершение теории: простой запрос

Мы углубимся в аналитику чуть позже. А пока попытаемся ответить на три вопроса:

1. Какие пять файлов данных имеют самую высокую среднюю задержку чтения при вводе-выводе?
2. Какие пять файлов журналов имеют самую высокую среднюю задержку записи при вводе-выводе?
3. К какому типу по уровню задержки ввода-вывода относится база данных, которая стала первопричиной всей этой работы?

Пять файлов данных с самой высокой средней задержкой чтения при вводе-выводе определяются с помощью кода листинга 3.

Неудивительно, что три файла с худшими показателями задержки принадлежат к нашей реализации Sharepoint (экран 3). Однако 5 секунд для каждого — это неожиданно и требует разъяснения. Вы заметите, что и здесь я включил ежедневную нагрузку чтения. Высокая задержка для файла с низкой активностью не повод для беспокойства. Однако именно