

## 01. Задание

1. Встроить в Вашу информационную систему подсистему БСП "Оценка производительности", если ее нет
2. Составить список из 15 ключевых операций Вашей информационной системы

Список обязательно должен содержать операции проведения документов и формирования отчетов. Возможны другие операции, имеющие ценность для бизнеса

Предварительный список операций составить самостоятельно, окончательный список получить после согласования с бизнесом

3. Настроить сбор Апдекс для всех согласованных операций
4. Получить целевое время ключевых операций

Предварительное целевое время получить самостоятельно, окончательное целевое время согласовать с бизнесом

Для грубой оценки предварительного целевого времени можно использовать методику "Обратный Апдекс"

5. Настроить сбор основных счетчиков производительности на сервере 1С, СУБД, терминальном сервере

Собрать как счетчики ОС, так и счетчики MS SQL

6. Прислать преподавателю результат выполнения отчета (обработки) "Оценка производительности"

В отчете должен быть согласованный список КО и согласованное целевое время

7. Прислать преподавателю логи Perfmon за 1 день с сервера 1С и СУБД.

Прислать собственные выводы о том, соответствуют ли значения счетчиков рекомендуемым

8. Установить Линукс любой версии и проверить работу предложенных в курсе команд по оценке загруженности оборудования

## 02. Задание

**1. Выяснить, какие задержки есть на вашем рабочем инстансе MS SQL Server.**

Игнорировать задержки, связанные с ожиданиями запуска служебных фоновых процессов.

Объяснить значение TOP 5 задержек.

Являются ли они проблемой? Как можно их уменьшить?

Начиная с какого момента были накоплены эти задержки?

Как очистить накопленную статистику?

**2. Выяснить, какую часть нагрузки создают запросы по всем базам с точки зрения объема чтения и записи.**

Выяснить, какая из рабочих баз наиболее нагруженная.

Выяснить, какую часть нагрузки создает база tempdb

**3. Выяснить средний отклик при чтении и записи ко всем файлам всех баз.**

В пределах ли нормы этот отклик?

**4. Выяснить, какие индексы не используются в системе.**

Начиная с какого момента накапливалась эта информация?

**5. В самой нагруженной рабочей базе, которую определили в п.2, найти TOP 5 таблиц по количеству записей.**

Выяснить, какие индексы предлагает дополнительно создать MS SQL Server для них.

Можно ли такие или похожие индексы создать штатными средствами?

Начиная с какого момента накапливалась информация о недостающих индексах?

**6. С помощью DMV найти TOP 5 запросов, которые потребляют больше всего CPU на СУБД.**

Нормализовать результат DMV с помощью функций fn\_GetSQLHash\_3.sql, ProcessTrace\_PerformHASH.sql

С какого момента накапливалась статистика по найденным запросам?

Какие индексы предлагает дополнительно создать MS SQL Server для таблиц, которые используются в найденных TOP 5 запросах.

Можно ли такие или похожие индексы создать штатными средствами?

Начиная с какого момента накапливалась информация о недостающих индексах?

### 03. Задание

1. Для самой нагруженной базы собрать ТЖ за 4 часа в рабочее время с настройками, см ниже.

Рекомендуется собирать ТЖ в период, когда есть снижение производительности или нагрузка наиболее высокая.

Необходимо следить за свободным местом на диске, где будут собираться логи ТЖ!

Если место слишком быстро закончится, увеличить параметр Duration или собирать ТЖ на диске с большим объемом свободного места.

```
<?xml version="1.0"?>
<config xmlns="http://v8.1c.ru/v8/tech-log">

  <log location="c:\1c_logs\queries_total" history="4">

    <event>
      <eq property="name" value="DBMSSQL"/>
      <eq property="p:processName" value="ИмяВашейБазы"/>
      <ge property="Duration" value="10000"/>
    </event>

    <property name="Sql"/>
    <property name="Context"/>

  </log>

</config>
```

2. Проанализировать собранный ТЖ скриптом "Скрипт анализа ТЖ с запросами" и выяснить, какие запросы выполняются суммарно больше всего.
3. Модифицировать скрипт "Скрипт анализа ТЖ с запросами" так, чтобы он группировал запросы только по контекстам, а тексты запросов удалял.

Полученным скриптом выяснить, какие контексты выполняются суммарно больше всего.

4. Оптимизировать TOP 5 найденных запросов или предложить варианты их оптимизации.
5. Для 5 ключевых операций, которые имеют наибольший приоритет, собрать ТЖ с фильтром по контексту этих операций.

Например ТЖ с настройками ниже. Следить за свободным местом на диске с логами!

```
<?xml version="1.0"?>
<config xmlns="http://v8.1c.ru/v8/tech-log">

  <log location="C:\1c_logs\РеализацияТоваровУслуг" history="8">

    <event>
      <eq property="name" value="DBMSSQL"/>
      <eq property="p:processName" value="ИмяВашейБазы"/>
      <like property="Context" value="%Документ.РеализацияТоваровУслуг%"/>
    </event>

    <property name="Usr"/>
    <property name="Sql"/>
    <property name="Context"/>

  </log>

</config>
```

```

</log>
<log location="C:\1c_logs\ПоступлениеТоваровУслуг" history="8">

    <event>
        <eq property="name" value="DBMSSQL"/>
        <eq property="p:processName" value="ИмяВашейБазы"/>
        <like property="Context" value="%Документ. ПоступлениеТоваровУслуг %"/>
    </event>

    <property name="Usr"/>
    <property name="Sql"/>
    <property name="Context"/>

</log>

```

... ниже может быть еще несколько настроек

```
</config>
```

6. **Модифицировать скрипт "Скрипт анализа ТЖ с запросами" так, чтобы он вычислял среднюю длительность запросов и упорядочивал по средней длительности.**

Среднюю длительность необходимо отформатировать, для этого нужно использовать printf  
Пример использования printf:

```
printf "%d=Это целое число, %.3f=Это дробное число с тремя знаками, %s=Это строка\n", 123, 456.789,
Вася
```

Проанализировать запросы для TOP 5 ключевых операций, группировав по средней длительности, затем суммарной длительности.

7. **Оптимизировать TOP 5 запросов для ключевых операций или предложить варианты их оптимизации.**

8. **Собрать для самой нагруженной базы информацию о том, какие серверные вызовы потребляют больше всего памяти на сервере приложений:**

- Суммарно
- В среднем

Как можно уменьшить потребляемую память этих вызовов?

9. **Снять трассировку с помощью Extended Events на продуктиве за 4 часа в рабочее время.**

Рекомендуется снимать трассировку в период, когда есть снижение производительности или нагрузка наиболее высокая.

Использовать события rpc\_completed, sql\_batch\_completed, отбор по ID самой нагруженной базы, отбор по длительности >= 0.5 сек

Следить за размером файлов трассировки!

Полученную трассировку загрузить в таблицу базы данных и нормализовать скриптами  
fn\_GetSQLHash\_3.sql, ProcessTrace\_PerformHASH.sql

10. **Найти в трассировке TOP 5 запросов, которые потребляют CPU на сервере СУБД.**

Найти эти запросы в рабочей базе.

Оптимизировать найденные запросы, либо предложить варианты оптимизации.

11. **Найти в трассировке TOP 5 запросов, которые пишут в tempdb.**

Рекомендуется мониторить в период, когда отклик к tempdb низкий.

Найти эти запросы в рабочей базе.

Оптимизировать найденные запросы, либо предложить варианты оптимизации.

## 04. Задание

### 1. Воспроизвести в тестовой базе утечку памяти из за циклической ссылки на клиенте и на сервере

Расследовать эту утечку всеми известными Вам способами.  
Описать эти способы.

Дополнительно:

Проанализировать на рабочей базе с помощью ТЖ и события SCRIPTCIRCREFS, есть ли циклические ссылки на продуктиве.

Устранить найденные утечки.

### 2. Воспроизвести на тестовой базе TrProf83 таймауты на блокировках:

- при одновременной записи констант
- при обмене данными: при выгрузке изменений и удалении регистрации изменений
- при сдвиге границы последовательности
- при отсутствии разделения итогов
- на пустой таблице в автоматическом режиме блокировок

Воспроизвести проблемы как в управляемом (кроме пустой таблицы), так и в автоматическом режиме.

Расследовать блокировки с помощью ЦУП, ТЖ, Extended events/Profiler

### 3. Воспроизвести на тестовой базе TrProf83 дедлоки:

- из за разного порядка записи наборов записей регистров накопления
- из за разного порядка записи элементов справочников или документов
- из за повышения блокировки при чтении и записи набора записей регистра
- в автоматическом режиме из за повышения блокировки при чтении запросом и последующей записи прочитанных данных

Первые три проблемы воспроизвести как в управляемом, так и в автоматическом режиме.

Почему последний дедлок не воспроизводится в управляемом режиме?

Расследовать полученные дедлоки с помощью ТЖ (управляемые блокировки) , ЦУП и Extended events/Profiler

В случае дедлока на СУБД получить deadlock graph в трассировке.

### 4. Воспроизвести эскалацию блокировки на СУБД (MS SQL Server) и на управляемых блокировках

Каким образом можно избавиться от проблемы эскалации?

Перечислите все способы, как избавиться от эскалации на MS SQL Server и управляемых блокировках

### 5. Найти на продуктиве или в тестовой базе с помощью ТЖ TOP 5 самых долгих транзакций

Где в коде 1С эти транзакции были инициированы?

### 6. Найти на продуктиве все таймауты и дедлоки на блокировках 1С и СУБД с помощью ТЖ и Extended events

С какими событиями в этом случае целесообразно собирать ТЖ?

С какими событиями целесообразно собирать трассировку Extended events?

## 05. Задание

Устранить найденные проблемы блокировок, либо предложить решение.

1. **Внедрить «1С Тест центр» в одну из копий продуктивных баз**
2. **Написать простой нагрузочный тест для 3 самых активно используемых документов и 2 активно используемых отчетов**

Можно взять первые 3 вида документа в списке Apdex и первые 2 вида отчета в списке Apdex

В тесте должно быть минимум 3 роли и минимум 3 ВРМ

Каждая операция выполняется минимум 10 раз

В тесте выполняем следующее:

- Открываем формы списка
- Выполняем поиск/отбор в списке
- Открываем формы документов
- Проводим документы из формы
- Формируем отчеты

Необходимо предусмотреть:

1. Все пользователи проводят разные документы и отчеты с разными настройками
  2. Тест должен быть масштабируемым, т.е. с помощью некоторого коэффициента вы задаете пропорциональное увеличение количества всех операций для всех ролей
  3. Между операциями в тесте должны быть разумные интервалы времени  
Т.е. пользователи не должны выполнять операции непрерывно, однако интервалы подбираются так, чтобы сделать имитацию многопользовательской работы
  4. Тест должен быть повторяемым, т.е. несколько запусков теста приводили бы к проведению одних и тех же документов, формированию отчетов с одними настройками и в одинаковой последовательности
  5. Контролем остатков при проведении документов пренебрегать крайне не желательно, даже если фактическая запись в регистры будет осуществляться независимо от наличия остатков. Необходимо заранее предусмотреть такое количество остатков по всем регистрам, чтобы списания в минус не было.
  6. После выполнения теста предусмотреть удаление созданных в тесте документов. Эта опция должна быть отключаемой.
3. **Предусмотреть удаление созданных в тесте документов в нескольких фоновых заданиях**  
  
При этом количество одновременно выполняющихся фоновых заданий настраивается и может быть меньше числа ВРМ теста
  4. **Прислать преподавателю:**
    - Разработанную тестовую обработку/обработки
    - Скриншоты из конфигуратора для добавленных и измененных в конфигурации объектов, если такие изменения потребовались