

Разработка в
HighLoad.
Нужен ли
эксперт?

Здравствуйте! я Сергей Носков

Руководитель отдела производительности систем
в компании **BIA Technologies**



Задачи отдела

- ☐ Аудит.
- ☐ Оптимизация.
- ☐ Нагрузочное тестирование.

5'000+

Одновременно работающих пользователей

18'000 Gb

Суммарный объем данных

1'500'000

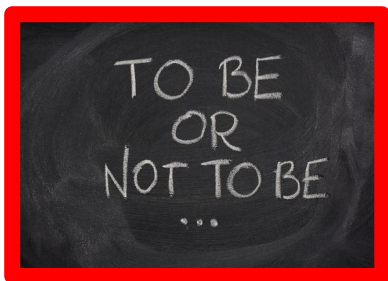
*Количество изменений документов за **один** рабочий день*

55'000

***Среднее** число запросов в секунду*

27'000

***Среднее** число транзакций в секунду*



APDEX

*36.6°C не 100% показатель
здоровья, но это первое, что
мы проверяем.*



- ❑ Определи важные бизнес-процессы.
- ❑ Рассчитай ключевое время.
- ❑ Учитывай количество пользователей, меняй границы.
- ❑ Нетерпеливые пользователи?
Измени 4Т на 3Т...

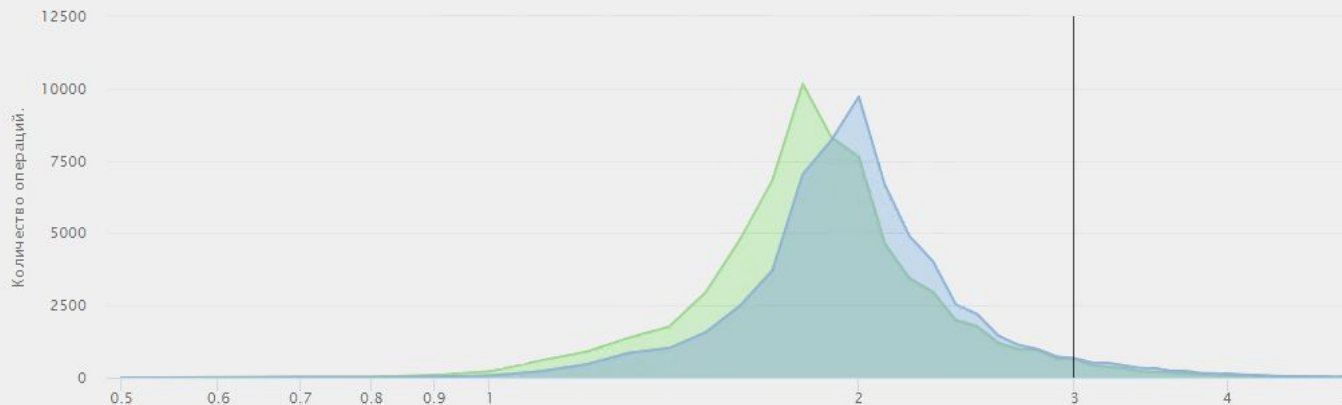
И да, **apdex** требует внедрения, как любой другой бизнес-процесс.

Индекс это
отражение
процента
довольных

	Apdex
<i>Отлично</i>	0,94 – 1,00
<i>Хорошо</i>	0,85 - 0,93
<i>Удовлетворительно</i>	0,70 - 0,84
<i>Плохо</i>	0,50 - 0,69
<i>Неприемлемо</i>	0,00 - 0,49

Действуй на опережение

- ☐ Контролируй отклонения от обычного уровня.
- ☐ Заметил изменение - **действуй сразу**. Не жди.



Разработка Hi-Load

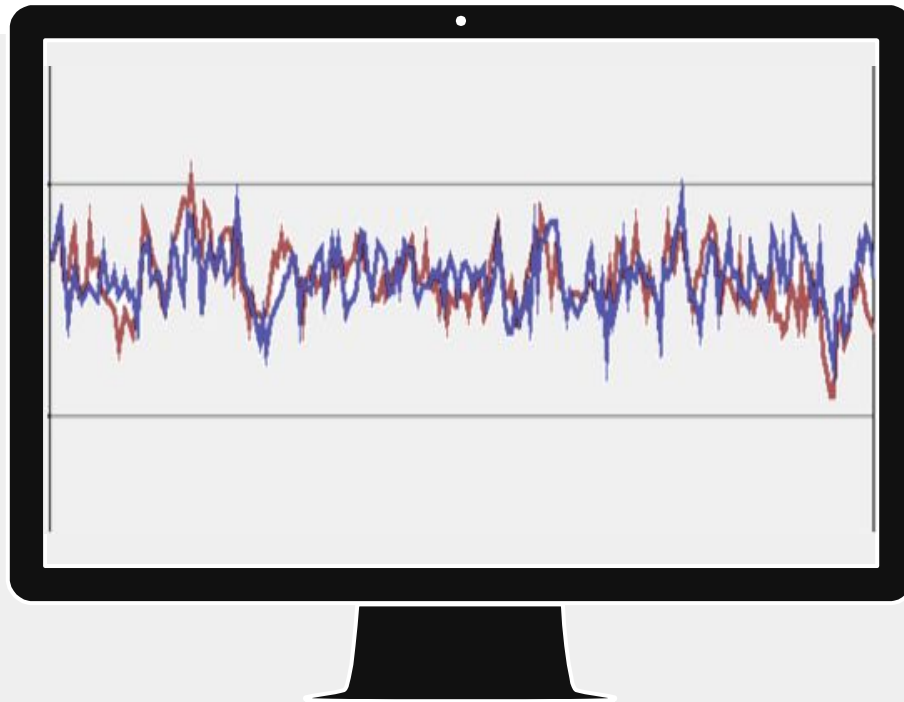
*Для того, что бы база работала быстро,
достаточно всего лишь сделать это*



Это
ВСЕГДА
база данных

- ❑ Ты разрабатываешь **базу данных**.
- ❑ Научись читать план запроса.
- ❑ Изучи как работает **индекс**.
- ❑ Не кидайся пилить запросы - **думай** над архитектурой.
- ❑ Ошибки неизбежны. Просто имей это ввиду.

Нагрузку на
сервер СУБД
создают запросы



Корреляция нагрузки на
процессор и количества запросов.

1. Изучи как «работает» индекс

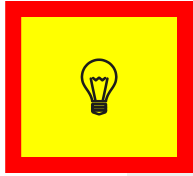
YouTube канал: *Russian VC*

Докладчик: Дмитрий Короткевич

Доклад: «Индексы и всё, всё, всё»

2. Индексы платформы

Запомни **принцип** создания индексов
платформой 1С.



Правильность архитектуры хранения данных определяется тем, как будут использоваться эти данные.

3. Правильно оценивай качество запроса

Всего 2 параметра определяют
оптимальность запроса:

- ❑ Объем прочитанных данных.
- ❑ Затраченное процессорное время.

4. Исследуй «физику» процесса

Разберись сам, почему стандарты именно такие, исследуя «как это работает»

- ☐ Как мой запрос «выглядит» в СУБД?
- ☐ В цикле точно дольше/хуже?
- ☐ А временная таблица тут нужна?.. а её индекс?
- ☐ Где в СУБД виртуальные таблицы?
- ☐ ...



ошибки

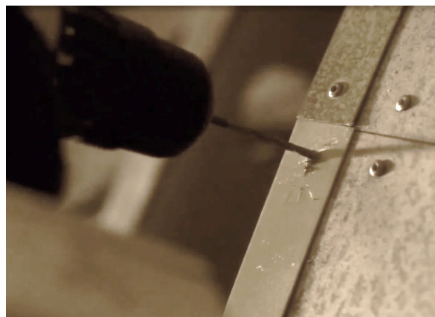
*Примеры из сферы, где само их
наличие вызывает «шок и трепет».*

История одного костыля

*Задача: установить титановую
пластину.*

*Проблема: отверстия на корпусе и
пластине не совпадают.*

Механик
авиакомпания
Continental Airlines



История одного костыля



Через 16 часов
пластина
отваливается при
взлете



История одного костыля



Финал истории

Кусок покрышки весом 4-5 кг врезается в крыло и вызывает **гидроудар** в топливном баке.



Почему это важно

Извлекаем уроки.

- ☐ Все косячат. Иногда самая простая доработка вызывает самые большие проблемы.
- ☐ Расследуй каждый сбой, каждую аварию.
- ☐ Организуй «черные ящики» - собирай логи ТЖ и нагрузку на железо.
- ☐ Распиши, кто и что должен делать в момент сбоя.

Архитектура бизнес-процесса без регистров.

Когда

1. Простой бизнес-процесс.
2. Фильтр по **одному** реквизиту документа возвращает очень мало строк.

Риски

1. «Сюрпризы» при доработках.
2. Хуже «оптимальность» запросов.

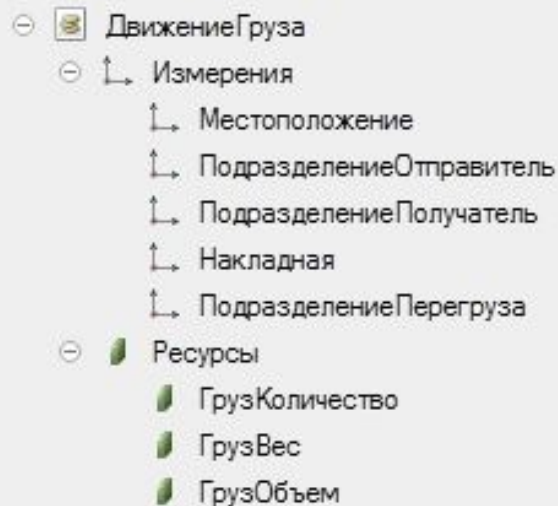
Одна таблицы для N-учетов.

Когда

1. Малый объем данных.
2. «Молодой» бизнес-процесс.

Риски

1. Регламенты на большой таблице приводят к бо́льшим проблемам.
2. Переделка не окупается.



Консоль запросов «в массы».

Когда

1. Нет программистов.
2. «SLA??!
Не, не слышали..»

Риски

1. Потеря контроля.
2. Нет развития архитектуры.
3. Не отобразить.

Спасибо!

Вопросы?

Sergey.Noskov@bia-tech.ru

✈ @SergeyANoskov

?