



Размер Имеет Значение:

10 способов уменьшить размер БД и улучшить производительность системы

Dmitri Korotkevitch
<http://aboutsqlserver.com>

Здравствуйте!

20+ лет в IT

15+ лет работы с SQL Server

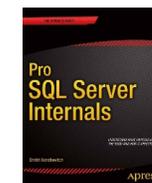
Microsoft SQL Server MVP

Microsoft Certified Master (SQL Server 2008)

Автор “Pro SQL Server Internals”

Blog: <http://aboutsqlserver.com>

Email: dk@aboutsqlserver.com



Зачем Уменьшать Размер БД?



Увеличение
производительности

Ускорение восстановления после
сбоев

Упрощение задач администрирования

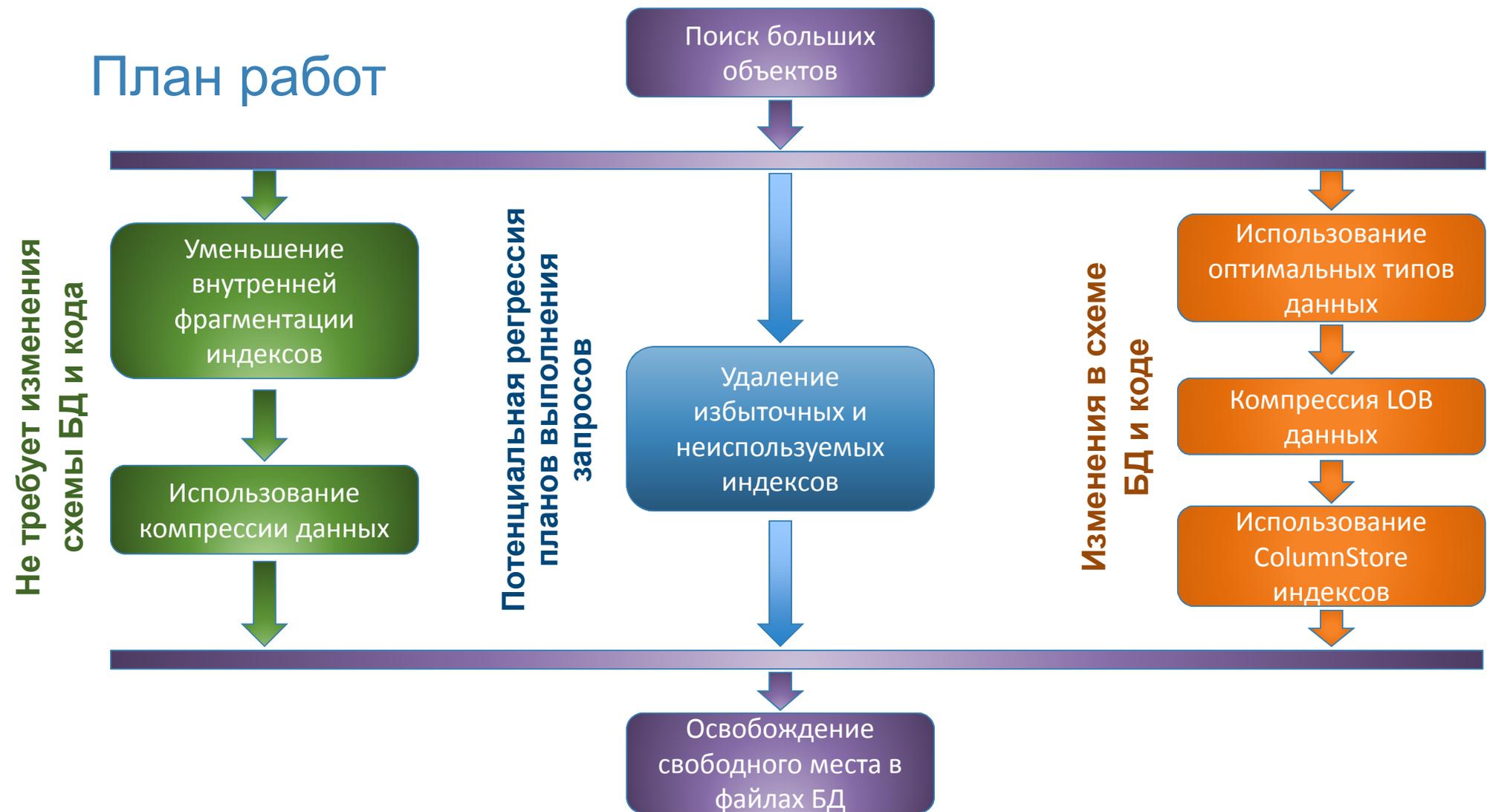
Уменьшение стоимости железа

Картинка с: <https://openclipart.org/detail/172012/award-by-gnokii-172012>



Демо:
DBCC SMART_SHRINK()

План работ



Рекомендованные Практики:

Auto Shrink: **OFF**

- Увеличивает фрагментацию индексов
- Увеличивает нагрузку на систему (I/O, CPU, T-Log)
- Абсолютно бесполезен

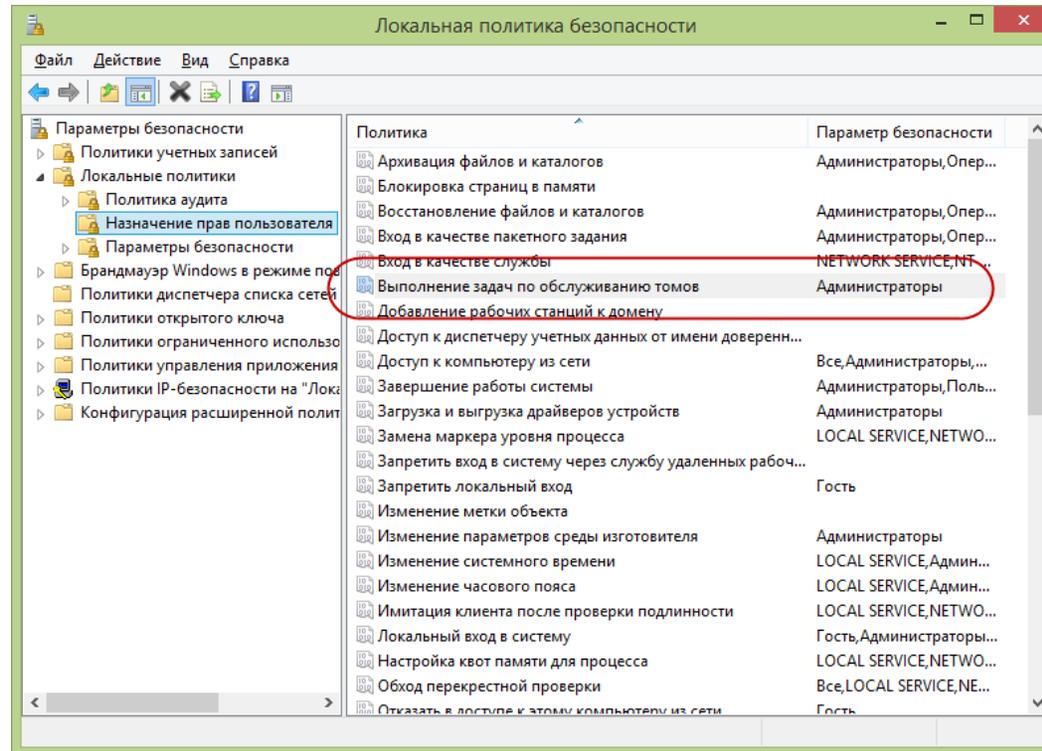
Instant File Initialization: **ON**

- Предотвращает заполнение файла данных нулями (zeroing out)
- Уменьшает время создания и роста файлов данных
- Непременимо к журналу транзакций

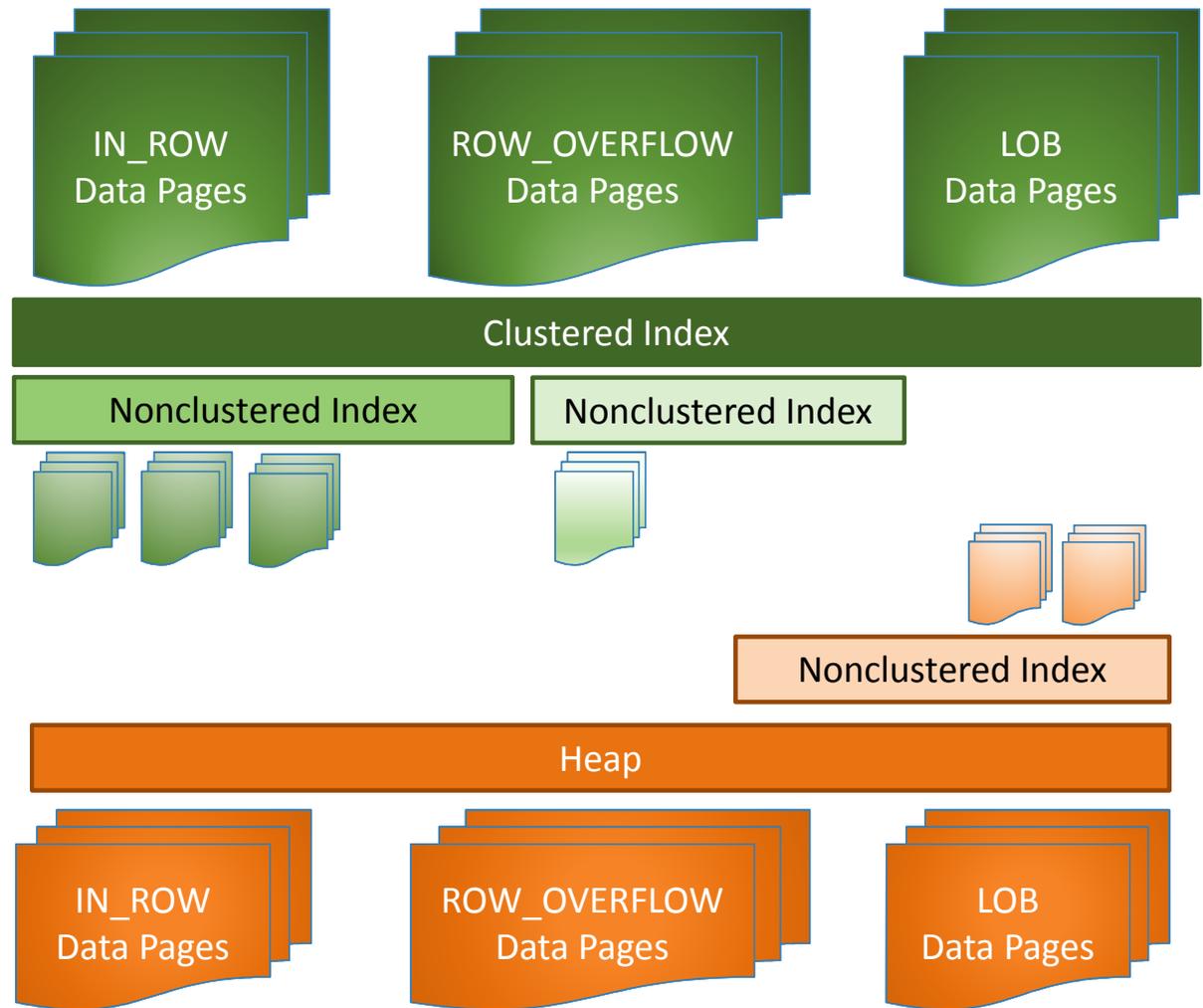
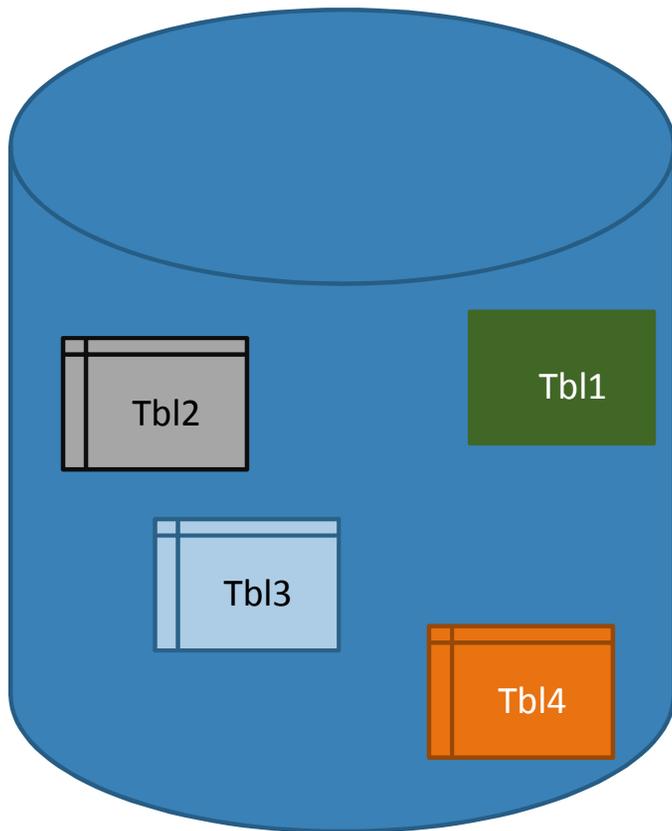


Демо: Instant File Initialization

Instant File Initialization



Хранение Данных



```

create table T
(
  ID int,
  DateCol datetime,
  VarCol1 varchar(5000),
  VarCol2 varchar(5000),
  VarCol3 varchar(max)
);

```

```

insert into T values
(
  1

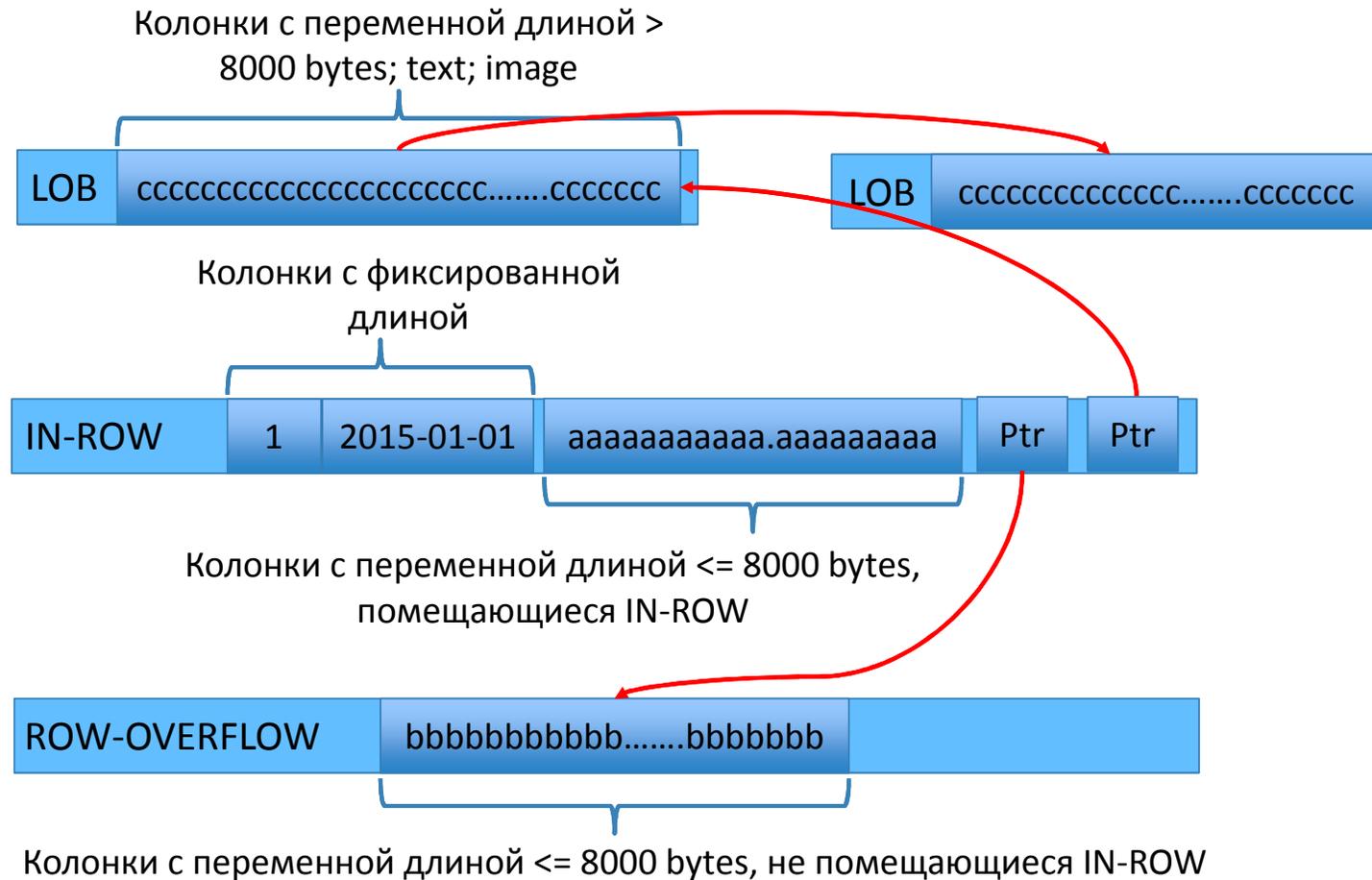
```

```

, '2015-01-01'
, REPLICATE('a', 5000)
, REPLICATE('b', 5000)
, REPLICATE('c', 32000)
);

```

Хранение Данных



План работ

Не требует изменения
схемы БД и кода



Потенциальная регрессия
планов выполнения
запросов



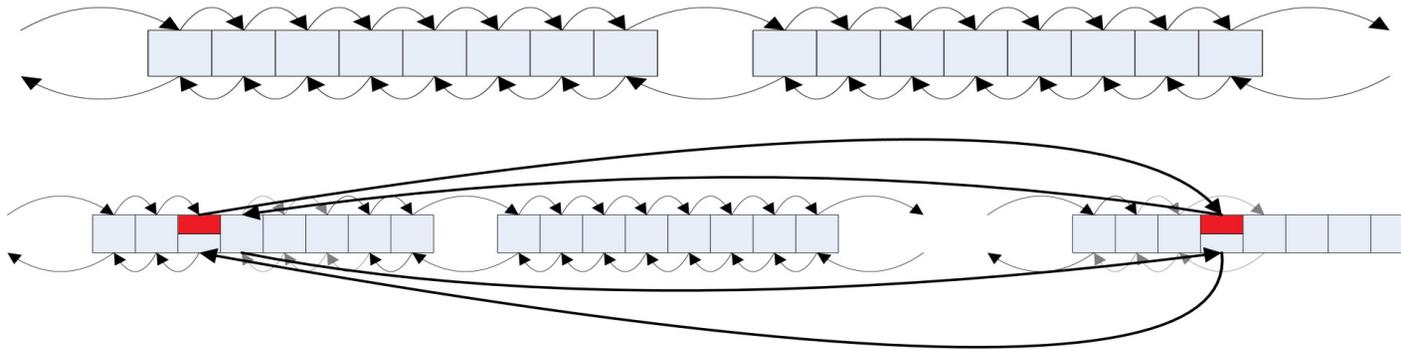
Изменения в схеме
БД и коде





Демо: Поиск Больших
Объектов

Фрагментация





Демо: Фрагментация

Компрессия Данных (Enterprise Edition)

Обычная запись

- Колонки с фиксированной длиной: Размер зависит от типа данных
- Колонки с переменной длиной: Размер зависит от данных + 2 байта

ROW Compression

- Убирает избыточность хранения данных в колонках с фиксированной длиной

PAGE Compression = ROW Compression + Prefix + Dictionary Compression

- Компрессия работает в рамках страницы

Компрессия работает только с IN_ROW данными



Демо: Data Compression

ЛОВ Компрессия

```
[Microsoft.SqlServer.Server.SqlFunction
(IsDeterministic = true, IsPrecise = true,
 DataAccess = DataAccessKind.None)]
public static SqlBytes BinaryCompress(SqlBytes input)
{
    if (input.IsNull)
        return SqlBytes.Null;

    using (MemoryStream result = new MemoryStream())
    {
        using (DeflateStream deflateStream = new DeflateStream(result,
            CompressionMode.Compress, true))
        {
            deflateStream.Write(input.Buffer, 0, input.Buffer.Length);
            deflateStream.Flush();
            deflateStream.Close();
        }
        return new SqlBytes(result.ToArray());
    }
}
```

```
[Microsoft.SqlServer.Server.SqlFunction
(IsDeterministic = true, IsPrecise = true,
 DataAccess = DataAccessKind.None)]
public static SqlBytes BinaryDecompress(SqlBytes input)
{
    if (input.IsNull)
        return SqlBytes.Null;

    int batchSize = 32768;
    byte[] buf = new byte[batchSize];

    using (MemoryStream result = new MemoryStream())
    {
        using (DeflateStream deflateStream =
            new DeflateStream(input.Stream,
                CompressionMode.Decompress, true))
        {
            int bytesRead;
            while ((bytesRead = deflateStream.Read(buf, 0, batchSize)) > 0)
                result.Write(buf, 0, bytesRead);
        }
        return new SqlBytes(result.ToArray());
    }
}
```



Демо: Компрессия LOB Данных

Использование Корректных Типов Данных

Типичные Ошибки:

- Boolean -> tinyint, smallint, int
- (n)char(N)

Избыточная Точность

- DateTime -> SmallDateTime или DateTime2
- Float -> Decimal

Не используйте LOB столбцы если в них нет необходимости

Использование Корректных Типов Данных

1,000 rows - 15K
1,000,000 rows - 15M
365,000,000 rows - 5.4Gb

```
create table Positions
(
  ATime datetime not null, -- 8 bytes
  Latitude float not null, -- 8 bytes
  Longitude float not null, -- 8 bytes
  IsValid int not null, -- 4 bytes
  IsAssistanceUsed int not null, -- 4 bytes
  -- total: 32 bytes
)
```

```
create table Positions2
(
  ATime datetime2(0) not null, -- 6 bytes
  Latitude decimal(9,6) not null, -- 5 bytes
  Longitude decimal(9,6) not null, -- 5 bytes
  IsValid bit not null, -- 1 byte
  IsAssistanceUsed bit not null, -- 0 bytes
  -- total: 17 bytes
)
```

Думайте о будущем!

Пересоздавайте индексы после изменения структуры таблицы (ALTER TABLE)



Демо: Изменение Структуры
Таблицы и Размер Столбца

Удаление Неиспользуемых Индексов

sys.dm_db_index_usage_stats

- Сколько раз индекс появлялся в планах выполнения запросов

sys.dm_db_index_operational_stats

- Сколько раз индекс использовался
- Включает I/O статистику и информацию о блокировках

Статистика очищается после ALTER INDEX..REBUILD (SQL Server 2012+)



Демо: Анализ Использования Индексов

Избыточные Индексы

IDX1(A, B) & IDX2(A) -> IDX2 избыточен

IDX3(A) INCLUDE (B) & IDX4(A) INCLUDE (C) -> IDX5(A) INCLUDE(B,C)

IDX6(A,B) & IDX7(A,C) ->

- IDX8(A,B) INCLUDE (C)
- IDX9(A,C) INCLUDE (B)
- Или оставить оба индекса: IDX6 & IDX7



Демо: Избыточные Индексы

ColumnStore Индексы

Колоночный формат хранения данных с хорошей компрессией

Оптимизация Data Warehouse запросов. Неоптимальны для OLTP

HEAP таблица без компрессии	HEAP таблица с PAGE компрессией	COLUMNSTORE компрессия	COLUMNSTORE_ARCHIVE компрессия
10,504MB	2,440MB	831MB	362MB

Таблица с 60M записей, сгенерированная из AdventureWorks2012DW..FactSalesBig



Демо: ColumnStore Индексы

Освобождение Пустого Места из Файла Данных

DBCC SHRINKFILE

- Ведет к фрагментации (возможно использование совместно с ALTER INDEX REORG)
- Потенциальные проблемы в случае нескольких файлов в файловой группе

INSERT INTO NewTable SELECT FROM OldTable

- Проблематично в случае если таблица должна быть доступна пользователям

CREATE INDEX WITH (DROP_EXISTING=ON) ON [NewFileGroup]

- Потенциальные проблемы с LOB-столбцами



Демо: Перенос Данных Между Файловыми Группами

Мы обсудили

Поиск больших объектов

Уменьшение
внутренней
фрагментации
индексов

Использование
компрессии данных

Потенциальная регрессия
планов выполнения
запросов

Удаление
избыточных и
неиспользуемых
индексов

Изменения в схеме
БД и коде

Использование
оптимальных типов
данных

Компрессия LOB
данных

Использование
ColumnStore
индексов

Освобождение
свободного места в
файлах БД

Q&A

Спасибо за внимание!

Блог: <http://aboutsqlserver.com>

- Скрипты: <http://aboutsqlserver.com/presentations>
- Пост на тему: “Size Does Matter..”: <http://goo.gl/J644Zz>

E-Mail: dk@aboutsqlserver.com



Вопросы?



Спасибо за участие!