



Администрирование баз данных



План

1. Общие понятия администрирования БД.
2. Режимы работы с БД.
3. Физическая организация данных.
4. Резервное копирование и восстановление БД.
5. Обеспечение целостности данных.

АДМИНИСТРИРОВАНИЕ БД

```
graph TD; A[АДМИНИСТРИРОВАНИЕ БД] --> B[Администрирование БД]; A --> C[Администрирование СУБД];
```

Администрирование БД:

- создание объектов БД;
- разработка структуры системы безопасности;
- реализация разработанной системы безопасности;
- контроль целостности и достоверности данных;
- проведение мониторинга выполнения запросов.

Администрирование СУБД:

- установка и обновление версий, исправлений;
- запуск и установка служб СУБД;
- конфигурирование и настройка СУБД;
- управление учетными записями пользователей;
- конфигурирование устройств резервного копирования;
- выполнение резервного копирования системных и пользовательских БД;
- создание и модифицирование БД;
- мониторинг СУБД.



Основные требования к БД:

- обработка транзакций;
- коллективный доступ к данным;
- разграничение прав доступа;
- распределенная обработка данных;
- реализация технологии хранилища данных на основе системы поддержки принятия решения;
- масштабируемость;
- снижение совокупной стоимости владения.

Режимы работы с БД

- Однопользовательский
- Многопользовательский
- последовательный
- параллельный
- с централизованной БД
- с распределенной БД

Распределенная БД

(Distributed DataBase - DDB)

включает фрагменты из нескольких БД,
которые располагаются на различных
узлах сети компьютеров, и, возможно
управляются различными СУБД.

12 свойств идеальной DDB (Дейт):

- 1) Локальная автономия
- 2) Независимость от центрального узла
- 3) Непрерывность доступа
- 4) Прозрачность расположения
- 5) Прозрачность фрагментации
- 6) Прозрачность тиражирования
 - n Обработка распределенных запросов
 - n Обработка распределенных транзакций
- 1) Независимость от оборудования
- 2) Независимость от операционных систем
- 3) Прозрачность сети
- 4) Независимость от СУБД

⇒ DDB - слабосвязанная сетевая структура, узлы которой представляют собой локальные базы данных

Физическая организация данных

Объекты :

- Файлы, объединенные в файловые группы;
- Журналы;
- Страницы, объединенные в экстенды;
- Записи (строки).

Файлы БД

- Файлы данных:
 - основной (Primary, *.mdf)**
 - вторичный (Secondary, *.ndf)**
- Один файл не может принадлежать нескольким БД одновременно
- Файл имеет два имени:
 - логическое**
 - физическое**



Журналы

- Хранится отдельно от данных
- Файл типа **Transaction Log**, *.ldf
- 25% от размера файлов БД
- БД имеет минимум 1 журнал



Файловые группы

- Файл может относиться только к одной группе
- В одной БД может быть до 256 файловых групп
- Не обязательно создание файловых групп (по умолчанию группа Primary)
- Параллельная запись во все файлы группы (пропорционально размерам файлов)



Страницы

- Идентифицируются парой параметров <FileID> <PageNO>
- Номера страниц уникальны только внутри одного файла
- Размер страницы 8 Кбайт
- Объем каждой страницы 8 192 байта (первые 96 байт – заголовок)



Экстенты

- **Uniform** (однородные) – все страницы экстента принадлежат одному объекту
- **Mixed** (смешанные) – страницы экстента принадлежат разным объектам



Типы страниц

- Data
- Text / Image
- Index
- Global Allocation Map (GAM)
и Secondary Global Allocation Map (SGAM)
- Page Free Space (PFS)
- Index Allocation Map (IAM)



Page Free Space (PFS)

1-50%

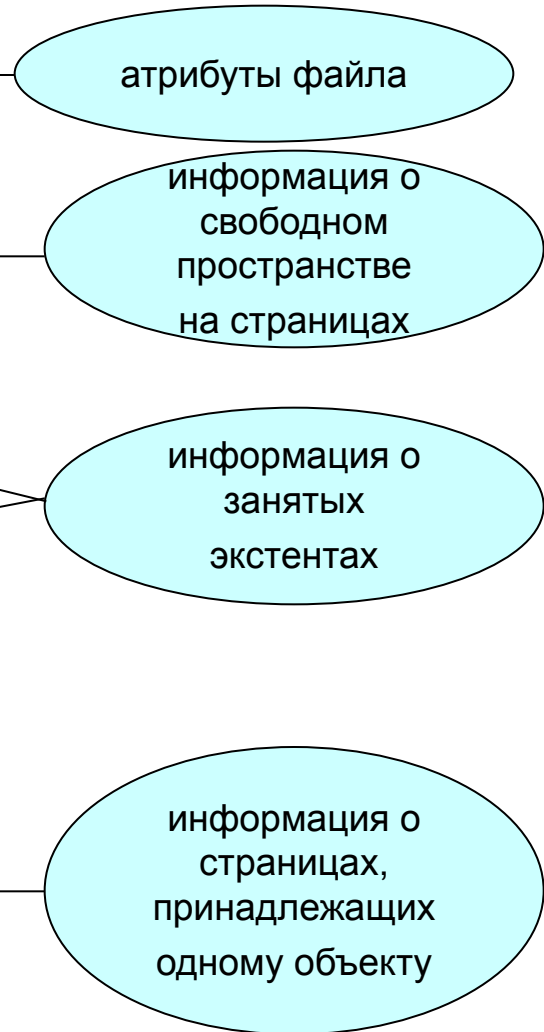
51-80%

81-95%

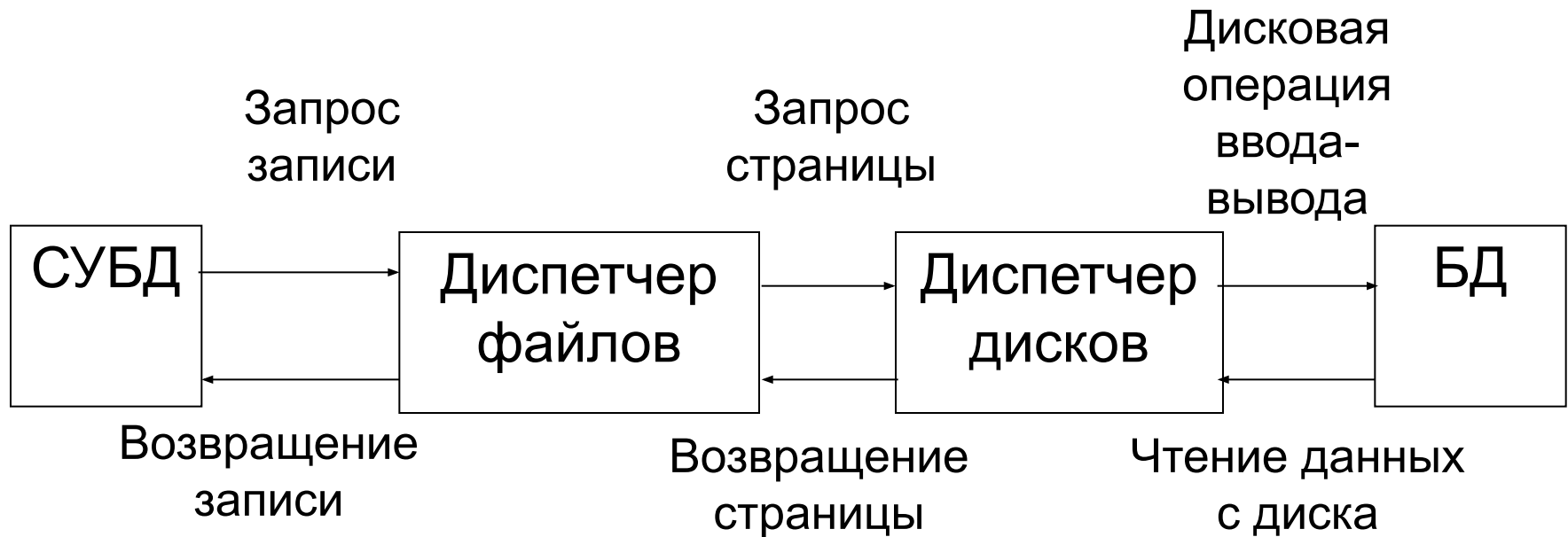
96-100%

свободна

Page 0 Атрибуты файла
Page 1 Page Free Space
Page 2 Global Allocation Map
Page 3 Secondary Global Allocation Map
Page 4 ... (X-1) Данные
Page X Index Allocation Map
Page (X+1) ... N Данные



Доступ к БД



Резервное копирование данных

- *полное копирование (Full);*
- *дифференциальное копирование (Differential);*
- *копирование журнала транзакций (Transaction Log).*

Восстановление базы данных

- *полное восстановление (Full);*
- *восстановление без массовых операций (Bulk-Logged);*
- *простое восстановление (Simple).*

Обеспечение целостности данных

Правила целостности:

- **Целостность объектов (сущностей):** в каждой таблице - первичный ключ \neq NULL.
- **Ссылочная целостность:** поддержание постоянной связи между первичным ключом и внешним ключом.
- **Целостность области значений (доменов):** определяет набор допустимых для поля значений (в том числе и NULL).

Обеспечение целостности:

- ***Декларативный способ.*** Критерии, которым должны удовлетворять данные, задаются при определении объекта и являются частью определения базы данных.
- ***Процедурный способ.*** Критерии описываются в пакетах операторов, выполнение которых и определяет целостность данных.


Декларативные способы обеспечения целостности:

Ограничение – это формулировка требований к данным на уровне поля или таблицы.

- **Первичный ключ (PRIMARY KEY)** - правило целостности объектов.
- **Внешний ключ (FOREIGN KEY)** - правило ссылочной целостности.

ON UPDATE { NO ACTION | CASCADE }

ON DELETE { NO ACTION | CASCADE }

- 
- **Ограничения уникальности значений (UNIQUE)**
 - **Ограничения значений по умолчанию (DEFAULT)**
 - **Ограничения проверки значений (CHECK)**
Like ' [0-9] [0-9] - [0-9] [0-9] -[0-9] [0-9] '

Процедурные способы обеспечения целостности:

Создание **отдельных объектов**, которые затем можно применять.

■ **Значения по умолчанию**

CREATE DEFAULT имя AS константа

Пример:

```
CREATE DEFAULT phone_default AS '00-00-00'
```


■ Правила

Определяют допустимый диапазон значения поля.

CREATE RULE имя_правила AS условие

Пример:


```
CREATE RULE R_rule AS @r > 0
```

■ Триггеры

Тип хранимой процедуры, автоматически выполняемый при определенных событиях.

Типы триггеров:

- триггеры INSERT,
- триггеры DELETE,
- триггеры UPDATE,
- триггеры, создаваемые с учетом одновременного возникновения и совпадения событий.



Назначение триггера – предотвращение вставки, изменения или удаления записей, отвечающим некоторым условиям.

Применение триггеров:

- обращение к данным в еще одной, связанной таблице;
- проверка дельты обновления;
- формирование сообщений об ошибках.

Примеры:

- ✓ триггер, не допускающий создание записи в таблице **Клиенты-Секции** для клиентов, чей абонемент просрочен.
- ✓ триггер, запрещающий вставлять в таблицу **Клиенты_Секции** записи, если количество человек, записанных в эту секцию, достигло 15.
- ✓ триггер, запрещающий удалять из таблицы **Клиенты** записи о клиентах, имеющих абонемент типа «Золотая карта». Вывод сообщения «Клиент с золотой картой!» при попытке удалить такую запись.

Выбор метода обеспечения целостности данных

Метод	Описание	Функциональность	Нагрузка	Выполнение до или после транзакции
Ограничения	Определяются при создании таблицы и проверяют данные перед началом транзакции, что повышает производительность	Средняя	Низкая	До
Значения по умолчанию и правила	Реализуются в виде отдельных объектов, которые могут быть привязаны к нескольким различным полям различных таблиц	Низкая	Средняя	До
Триггеры	Обеспечивают дополнительную функциональность, позволяют реализовать каскадные операции и сложную прикладную логику. При нарушении ограничения операции в БД отменяются.	Высокая	Высокая	После