

# 1 Производительность

Когда и в какой ситуации что выбирать?

Случаи в которых возникают проблемы с производительностью:

- Соединение таблиц
- Поиск данных, операции со строками

## 1.1 Механизм индексов

ISU	ФИО	Группа	Группа	ОП
-----	-----	--------	--------	----

*Индекс* – метод, который позволяет получить структуру данных, предназначенную для оптимизации поиска

Существует вне хранимого отношения, как отдельная структура данных

key	value - address of tuple
-----	--------------------------

### 1.1.1 Виды индексов

- *Первичный индекс* – файл упорядочен по первичному ключу и по нему же построен индекс, так что гарантируется уникальность каждого кортежа

$\text{map}\langle \text{int} \rangle \rightarrow T$

- *Кластерный индекс* – файл упорядочен по ключевому или по неключевому атрибуту, по которому построен индекс, при этом несколько кортежей (кластер) соответствуют одному значению индекса

$\text{map}\langle \text{int} \rangle \rightarrow \text{vector}\langle T \rangle$

- *Вторичный индекс* – построен по атрибуту отличного от того, по которому осуществлено упорядочивание

Может быть любое число вторичных индексов, но только один первичный и кластерный

### 1.1.2 Проблемы индексов

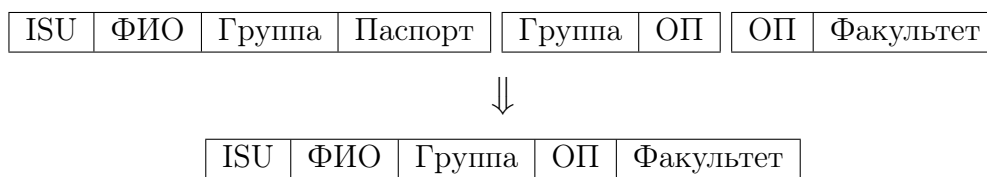
- Перестройка индексов
- Рост индексов

### 1.1.3 Виды индексов

- *Плотный индекс* – индекс, охватывающий все записи
- *Разреженный индекс* – индекс, охватывающий только определенные блоки

## 1.2 Механизм представлений

*Представление* – динамически сформированный результат одной или нескольких реляционных операций, выполненных над отношениями с целью получения нового отношения



### 1.2.1 Виды представлений

- *Материализованное представление* – непосредственно в памяти храним дубликаты данных, полученные в результате выполнения некоторого запроса, его актуально поддерживается за счет специальной процедуры
- *Представление замены* – храним структуру подзапроса

Представление замены решает больше задачу безопасности нежели производительности

### 1.2.2 Цели построения представлений

- Независимость от данных - внешние приложения будут удачно работать при редактировании мной исходной базы данных
- Повышение защищенности данных
- Снижение сложности запросов

### 1.2.3 Недостатки построения представлений

- Ограниченные возможности обновлений  
*Обновляемое представление* – могу внести изменение в само представление и существует надежный способ, позволяющий отобразить эти изменения на таблице
- Структурные ограничения - если я что-то меняю в моих таблицах нужно не забыть внести изменения в представления - проблемы с целостностью данных

- Снижение производительности