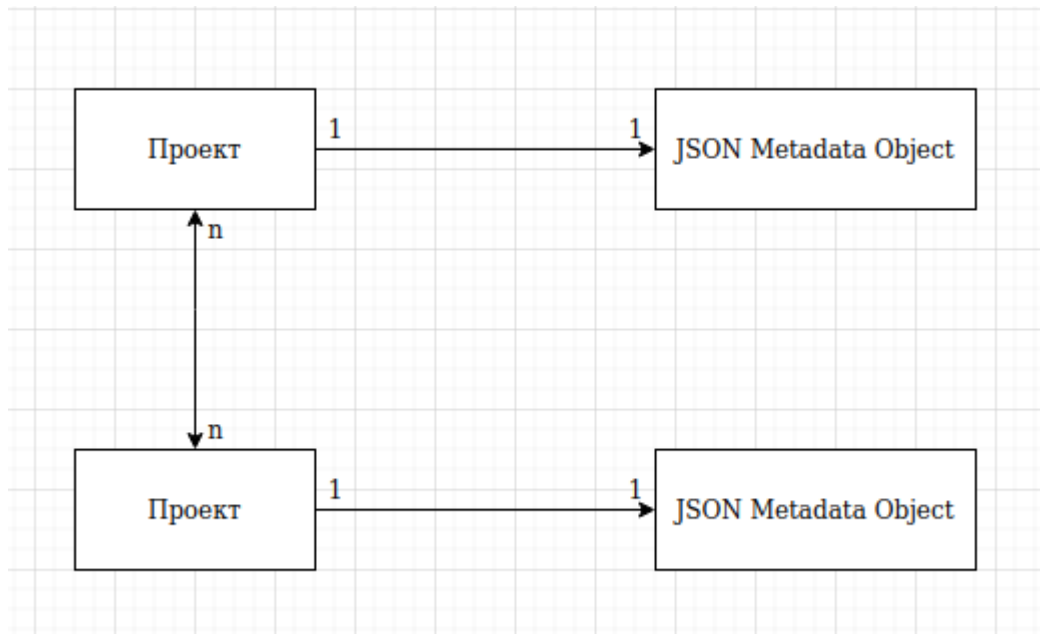


4. МОДЕЛЬ ДАННЫХ

Нереляционная модель

Графическое представление



Описание сущностей и типов данных

Сущность Проект. Атрибуты:

- id String 32 - ID проекта
- name String 20 - Название проекта
- description String 200 - Описание проекта
- images_count Integer 1 - Количество изображений
- metadata_size Double 1 - Размер мета-данных

Сущность JSON Metadata Object. Набор атрибутов сущности имеет динамическую природу.

Избыточность модели

Пусть N, M это количество записей для Проект и JMO соответственно. Пусть id будет весить 4 байта, тогда получим:

$$(4 + 20 + 200 + 1 + 1)N + (4 + \sim 1000)M = 226N + 1004M$$

Пусть $M = N$; Получим: $1230N$

Чистые данные не содержат id, тогда: $307N$

Вычислим отношение "фактического" и чистого объемов данных:

1230 / 307 = 4

Направление роста

По проектам: 30b

По json metadata object: 30b

Запросы

Просмотр всех проектов:

```
db.projects.find()
```

Поиск по проектам:

```
db.projects.find({"name": /.*{{project_name}}.*/})
```

Поиск по метаданным:

создание текстового индекса на все существующие поля

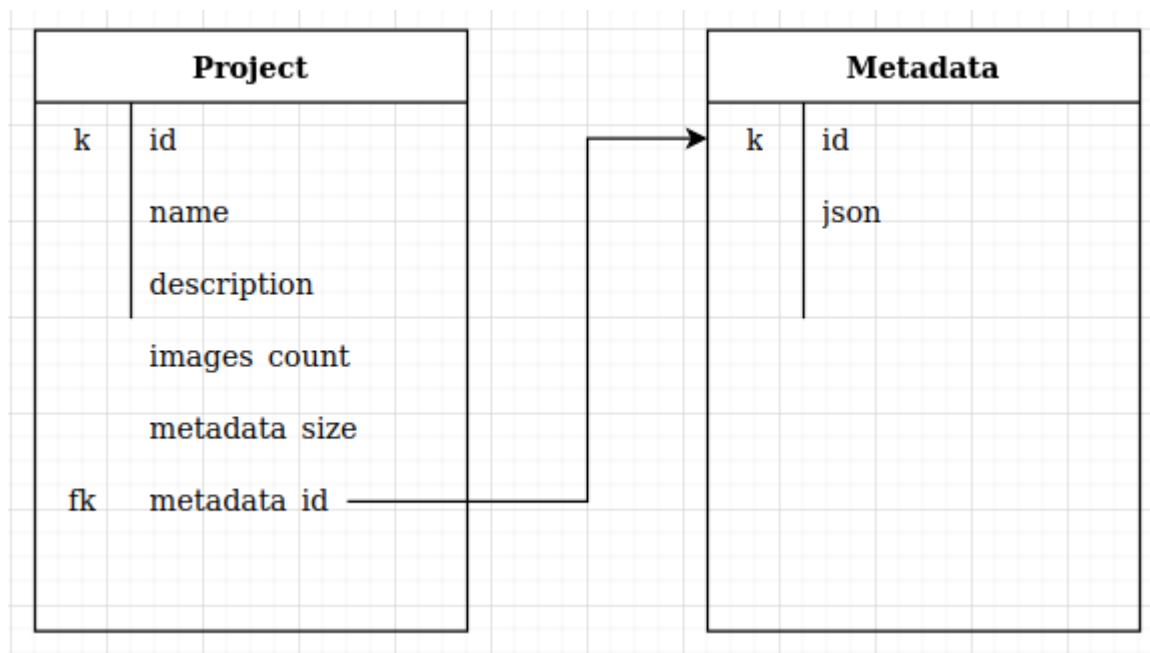
```
db.metadata.createIndex({ "$**": "text" })
```

поиск по всем представленным полям

```
db.metadata.find({ $text: { $search: "{{metadata_value}}" } }));
```

Реляционная модель

Схема данных



Описание таблиц и типов данных

Таблица Проект. Поля:

- name - строка, содержащая в себе название проекта, тип SHORT TEXT. Максимальное кол-во символов - 20.
- description - строка, содержащая в себе описание проекта, тип SHORT TEXT. Максимальное кол-во символов - 200.
- images_count - вещественное число, содержащее количество изображений, тип NUMBER.
- metadata_size - вещественное число, обозначающее размер мета данных, тип NUMBER.
- metadata_id - внешний ключ, создающий связь между проектом и его метаданными, тип SHORT TEXT

—

Таблица Метаданные. Поля:

- id - строка, содержащая в себе ID метаданных(сгенерированное в коде), тип - SHORT TEXT.

- json - метаданные проекта, тип - JSON

Избыточность модели

Пусть L, K - кол-во записей для таблиц Проект и Метаданные, соответственно. Тогда общий вес: $(4 + 20 + 200 + 1 + 1 + 4)K + (4 + \sim 1000)L = 230K + 1004L$

Пусть $L = K$, тогда общий вес будет равен $1234K$

Так как поле id и metadata_id дублируется в обеих таблицах (для обеспечения связи между ними) одно из этих полей является излишним, тогда чистый вес: $1230K$

Отношение фактического и чистого объема данных: $1234/1230 = 1.003$

Направление роста

По проектам: 30b

По json metadata object: 30b

Запросы

Просмотр проектов:

```
SELECT * FROM projects;
```

Поиск по проектам:

```
SELECT * FROM project WHERE project.name ILIKE  
'%{{project_name}}%';
```

Поиск по метаданным:

```
SELECT * FROM metadata WHERE metadata.json::text ILIKE  
'%{{metadata_value}}%';
```

Сравнение моделей

Можно заметить, что модели хранения SQL и NOSQL схожи. Только в модели SQL необходимо дублирование данных.