

# Яндекс

## Основы MySQL

**Павел Пушкарев**  
paulus@yandex-team.ru

19 октября 2011 года

# О чём всё это?

« DB — это слишком приземленно по сравнению с ML и AI. В таких областях мне всегда было лучше книжку прочитать, чем на лекции ходить »

# О чём всё это?

## СУБД

- Что такое «база данных», и зачем она вообще нужна?

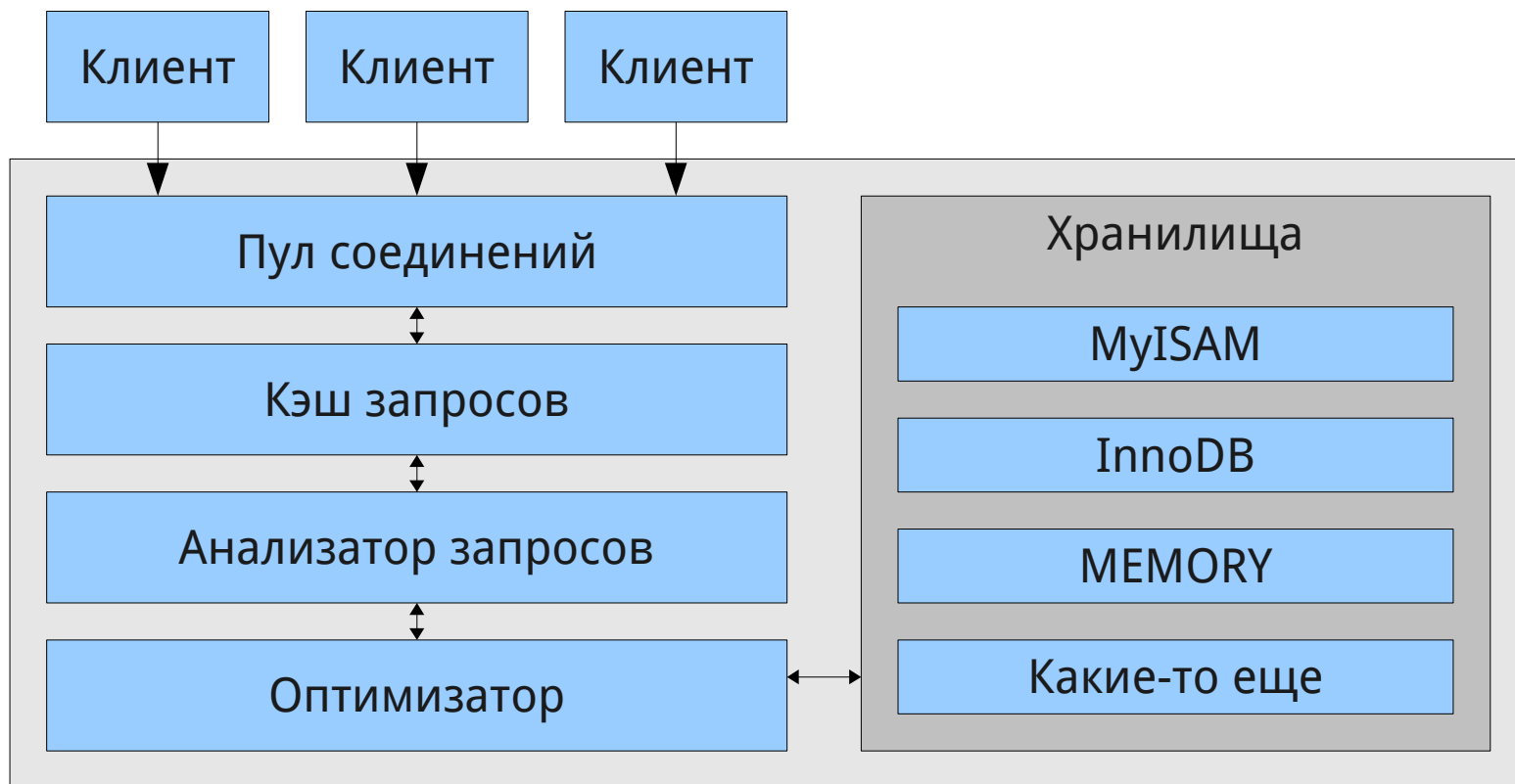
# Общие сведения MySQL как демон

- Клиент-серверная модель с использованием языка SQL
- Может использовать:
  - TCP/IP (по умолчанию — порт 3306)
  - Unix socket
  - Shared Memory
  - Pipes
- Огромное количество настроек

# Общие сведения Журналы работы

- Журнал ошибок
  - По умолчанию — `stdout`
- Журнал запросов
  - Раньше был `query`, теперь `general`
- Двоичный (или бинарный) журнал
  - `row`, `statement`, `mixed`
- Журнал медленных запросов

# Общие сведения Обработка запросов



# Пользователи MySQL

# Пользователи MySQL

## Идентификаторы

- Идентификатор пользователя состоит не только из имени, но и из имени обращающейся машинки
  - Например, `root@localhost`
  - Можно указывать IP или даже сети, например, `root@'127.0.0.1'`
  - `Unix socket == localhost`



# Пользователи MySQL

## Права доступа

- Права бывают глобальными, на базы, на таблицы, на столбцы, etc.
- Хранятся в системной базе mysql
- `GRANT ALL ON dbname.* TO user@localhost IDENTIFIED BY 'password'`

# Внезапно, фотография

- Кто это?



# Внезапно, фотография

- Майкл как бы намекает, что неплохо было бы сделать перерыв



# Хранилища

# Хранилища MyISAM

- Один из самых старых движков MySQL
- Быстрый, хорошо работает в случаях «много читаем, мало пишем»
- Поддерживает полнотекстовые индексы, геоинформацию
- Индекс — B-Tree
- Поддерживает дописывание в конец
- Блокировки на уровне таблиц

# Хранилища MyISAM на диске

- Каждая база — это каталог
- Файлы `frm` описывают структуру таблицы
- `MYD/MYI` содержат данные (и индексы) таблиц MyISAM

# Хранилища MyISAM в памяти

- Глобальные буферы
  - key buffer
  - table cache
- В памяти держатся только индексы
- Для доступа к данным полагается на file cache

# Хранилища InnoDB

- Поддержка ACID
  - Atomic
  - Consistent
  - Isolated
  - Durable
- Построчные блокировки
- Поддержка всех уровней изоляции
- Поддержка внешних ключей



# Хранилища InnoDB на диске

- Все данные хранятся в общем пространстве данных
- Можно настроить так, что данные, относящиеся к одной таблице, хранились в отдельных файлах `ibd`

# Хранилища InnoDB в памяти

- Использует собственные умные способы выделения памяти. Общий буфер — `innodb buffer pool`
- Хранит в памяти как индексы, так и сами данные
- Последнее обусловлено наличием кластерного ключа

# Блокировки

# Блокировки

## Принципы работы блокировок

- Блокировки нужны для многопользовательской работы
- MyISAM использует табличные блокировки
- InnoDB — построчные
- BDB использовал постраничные блокировки
- Блокировать можно на чтение и на запись

# Блокировки

## Блокировки в явном виде

- LOCK TABLES
  - READ [LOCAL]
  - WRITE
- SELECT ... LOCK IN SHARE MODE
  - FOR UPDATE

# Блокировки

## Уровни изоляции

- Путают с блокировками, это совсем не то же самое
- Бывают четыре уровня изоляции транзакций:
  - READ UNCOMMITTED
  - READ COMMITTED
  - REPEATABLE READ
  - SERIALIZABLE

# Резервное копирование

# Резервное копирование Двоичные бэкапы

- Хорошо работают для MyISAM
- Нужно в явном виде блокировать и сбрасывать таблички на диск
- Для InnoDB нужно использовать специальные утилиты — innobackupex или xtrabackup
- В случае InnoDB, копируется весь tablespace



# Резервное копирование Текстовые бэкапы

- `mysqldump` создает обычный SQL, который можно восстановить даже в другой СУБД
- Время восстановления сильно больше, чем у двоичного бэкапа
- Можно записывать положения в двоичном журнале для наливки реплик
- Особенно удобно в случае InnoDB с `--single-transaction`



## **Павел Пушкарев**

119021, Россия, Москва,  
ул. Льва Толстого, д. 16

+7 (495) 739-00-00

+7 (495) 739-70-70 — факс

[paulus@yandex-team.ru](mailto:paulus@yandex-team.ru)