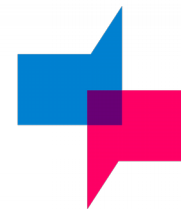


# Пятая конференция разработчиков ПО «DevParty»

25 марта 2017 года, Вологда



**DEV PARTY**  
конференция разработчиков

## Percona Server для MongoDB Обзор возможностей

Денис Противенский



PERCONA



# Что есть MongoDB?

---

NoSQL-база данных — Not SQL изначально, Not Only SQL впоследствии: отсутствие схемы данных

```
> var coll = db.coll
> coll.insert({ name : "John",
                title : "Developer" })
> coll.insert({ name : "Mike",
                age : 20 })
```



# Что есть MongoDB?

---

Удобна для быстрой разработки, когда основные запросы — это CRUD-операции

```
> coll.findOne({ age : { $gte : 18 } })  
{  
  "_id" :  
    ObjectId("58c25a0204f55ccc3c25c6c2"),  
  "name" : "John",  
  "age" : 20  
}
```



# Что есть MongoDB?

---

- Проста в конфигурировании, впоследствии можно без труда настроить репликацию и шардирование, прозрачные для клиентов
- JavaScript как основной язык доступа, удобен для веб-разработчиков
- MongoDB Community Edition — продукт с открытым исходным кодом и бесплатным распространением



# Чем отличается версия от Persona?

---

- Полная совместимость с MongoDB Community Edition 3.4 (\* версия для Linux)
- Два новых движка хранения: MongoRocks и Persona Memory Engine
- Поддержка аудита операций
- Внешняя аутентификация через LDAP и AD
- «Горячее» резервное копирование для WiredTiger и MongoRocks
- Ограничитель активности профилировщика



# MongoRocks

---

**mongod --storageEngine=rocksdb**

- Основан на библиотеке RocksDB  
(<https://github.com/facebook/rocksdb>)
- Оптимизирован на запись
- Хорошо подходит для единичных выборок
- Немного хуже для диапазонных выборок



# Percona Memory Engine

---

**mongod --storageEngine=InMemory**

- Совместим с движком InMemory в поставке MongoDB Enterprise
- Основан на WiredTiger
- Без персистентности
- Значительное ускорение записи
- Меньше джиттера
- Может использоваться как кеш приложения; для управления сессиями, сохранения временного состояния и т.д.



# Аудит

---

**mongod --auditDestination=file --auditFormat=JSON**

- Совместим с плагином аудита из поставки MongoDB Enterprise
- Позволяет отслеживать и сохранять активность пользователя:
  - Аутентификация (user- and role-mgmt, login, logout, shutdown)
  - Конфигурирование реплик/шардов
  - Неудачи регулярных операций (чтение, запись, курсор, схема)
- Несколько направлений вывода: файл, консоль, syslog
- Несколько форматов вывода: JSON, BSON





# Внешняя аутентификация

---

1. Конфигурация библиотеки SASL для доступа к saslauthd
2. Конфигурация saslauthd для доступа к серверу LDAP

```
> db.getSiblingDB('$external').auth(  
  { mechanism: 'PLAIN', user: 'user', pwd:  
    'secret', digestPassword: false } )
```

- Совместим с плагином внешней аутн. из поставки MongoDB Enterprise
- Прозрачный для клиента (используя метод PLAIN и базу \$external)
- Может работать через LDAP и AD
- Может авторизовывать, используя роли из базы \$external



# “Горячее” резервное копирование

---

```
> adminDB.runCommand( { createBackup: 1,  
backupDir: '/my/backup/data/path' } )
```

- Поддерживает WiredTiger и MongoRocks
- Работает без остановки основных активностей
- Перспективы:
  - Инкрементальное резервное копирование
  - Регулирование пропускной способности
  - Копирование по сети



# Ограничитель профилировщика

---

```
> db.runCommand( { profile: 2, slowms: 150,  
ratelimit: 3 } )
```

- Позволяет ограничивать количество быстрых запросов, которые сохраняются в коллекцию `system.profile` при профилировании
- Удобен для восстановления и анализа характера запросов пользователя без внесения существенных задержек в режим функционирования



**DEV PARTY**  
конференция разработчиков

---

# Вопросы?