

# Поддержка идентичности структуры БД при командной разработке

Михаил Кечинов

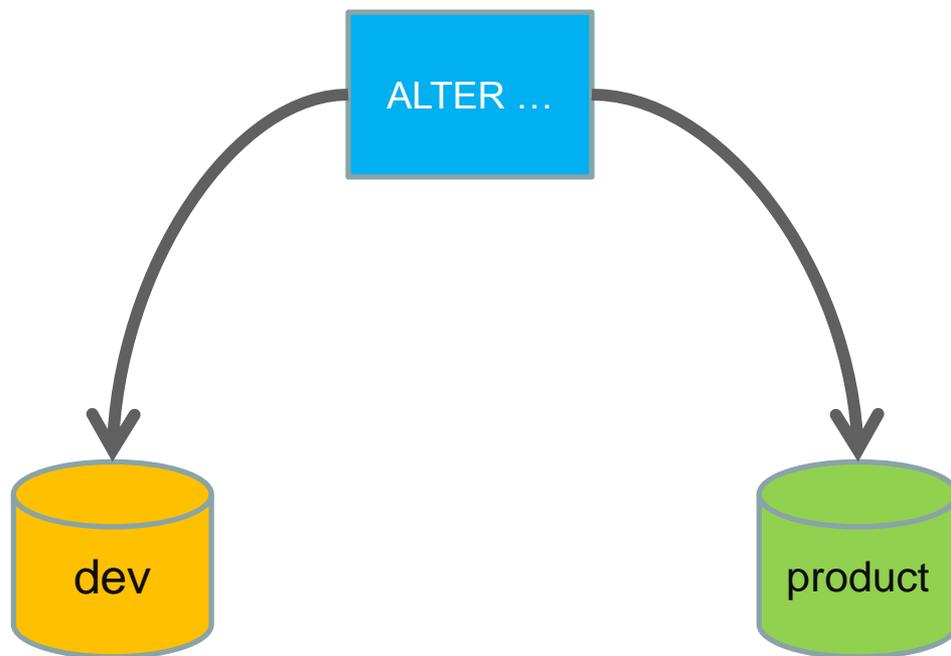
Студия веб-разработок Михаила Кечинова



<http://www.devconf.ru>

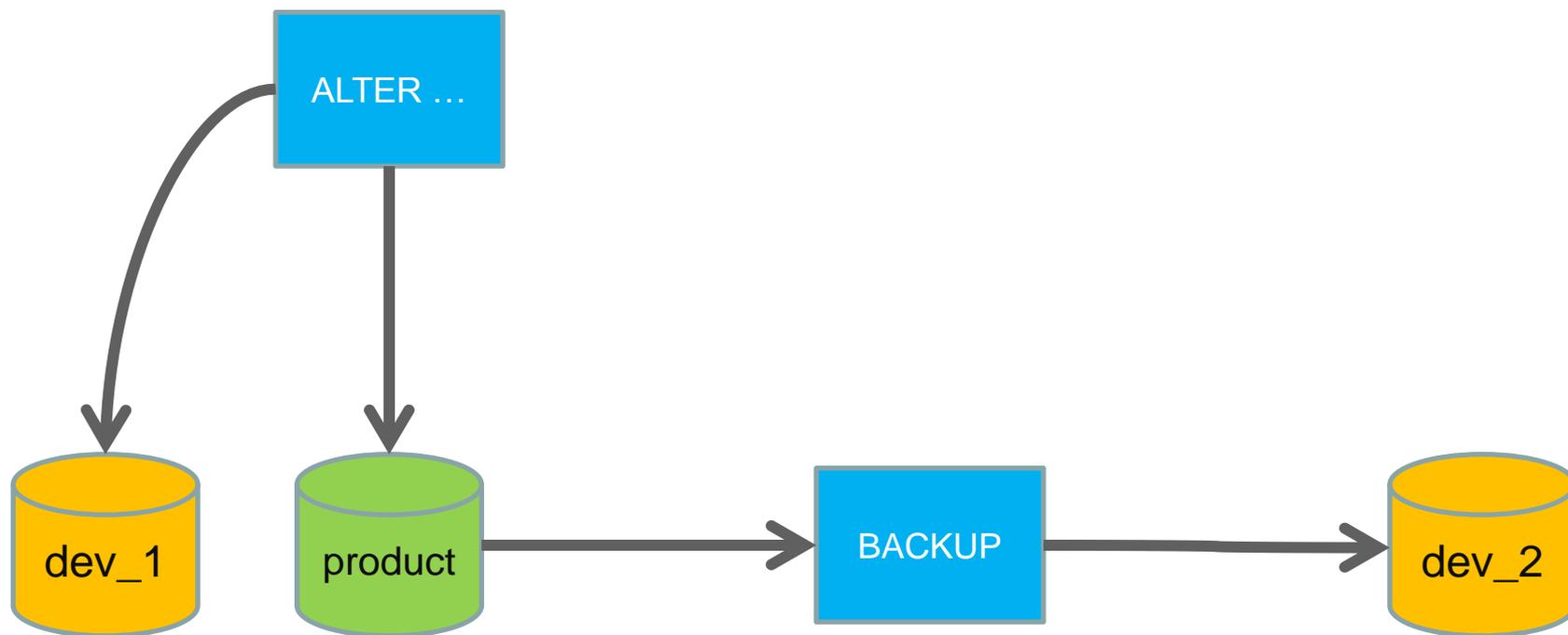


## Один разработчик, две базы



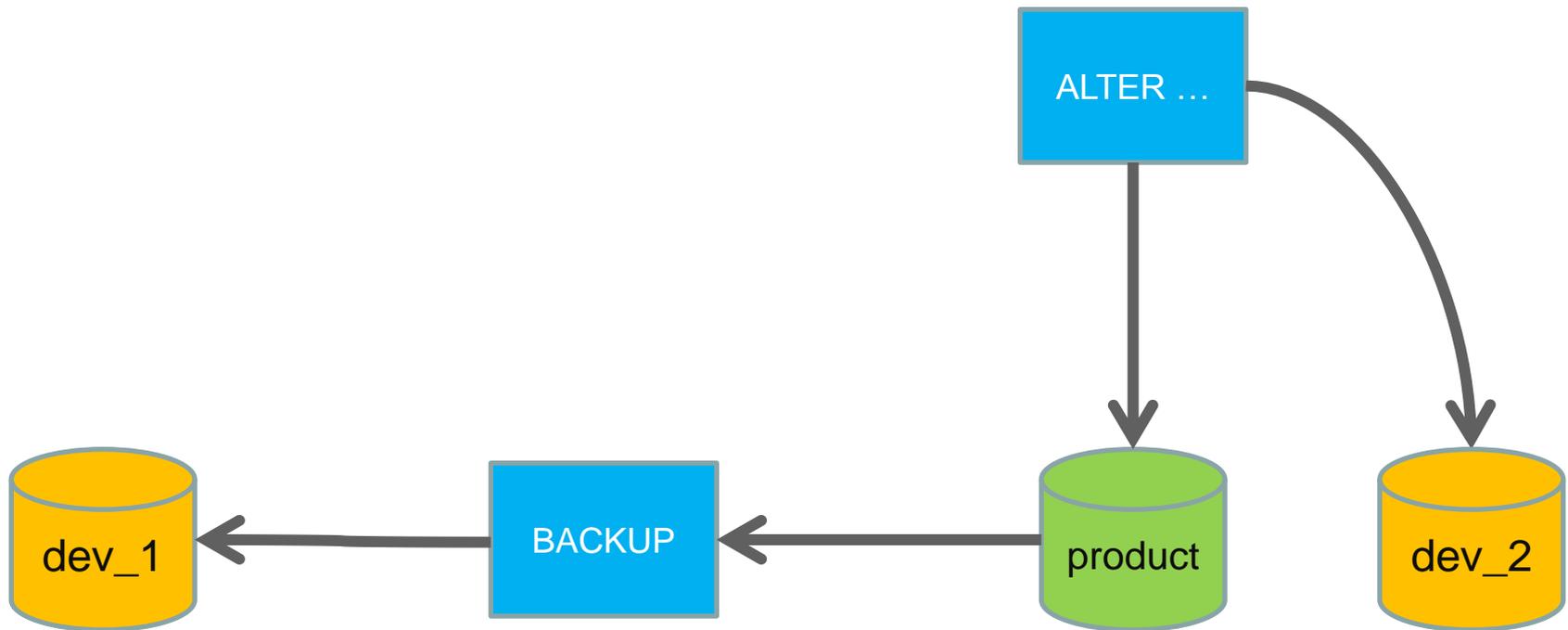
- Каждый запрос заносится в dev- и в production-базу.
- Для небольших проектов и одного человека.

## Один разработчик, три базы



- SQL-запрос на DEV1 и PRODUCTION
- Бэкап PRODUCTION и восстановление его на DEV2

## Один разработчик, три базы



- SQL-запрос на DEV2 и PRODUCTION
- Бэкап PRODUCTION и восстановление его на DEV1

## Два разработчика, много баз



```
<?php
define ("DB_REVISION", 34);

$scripts = array();

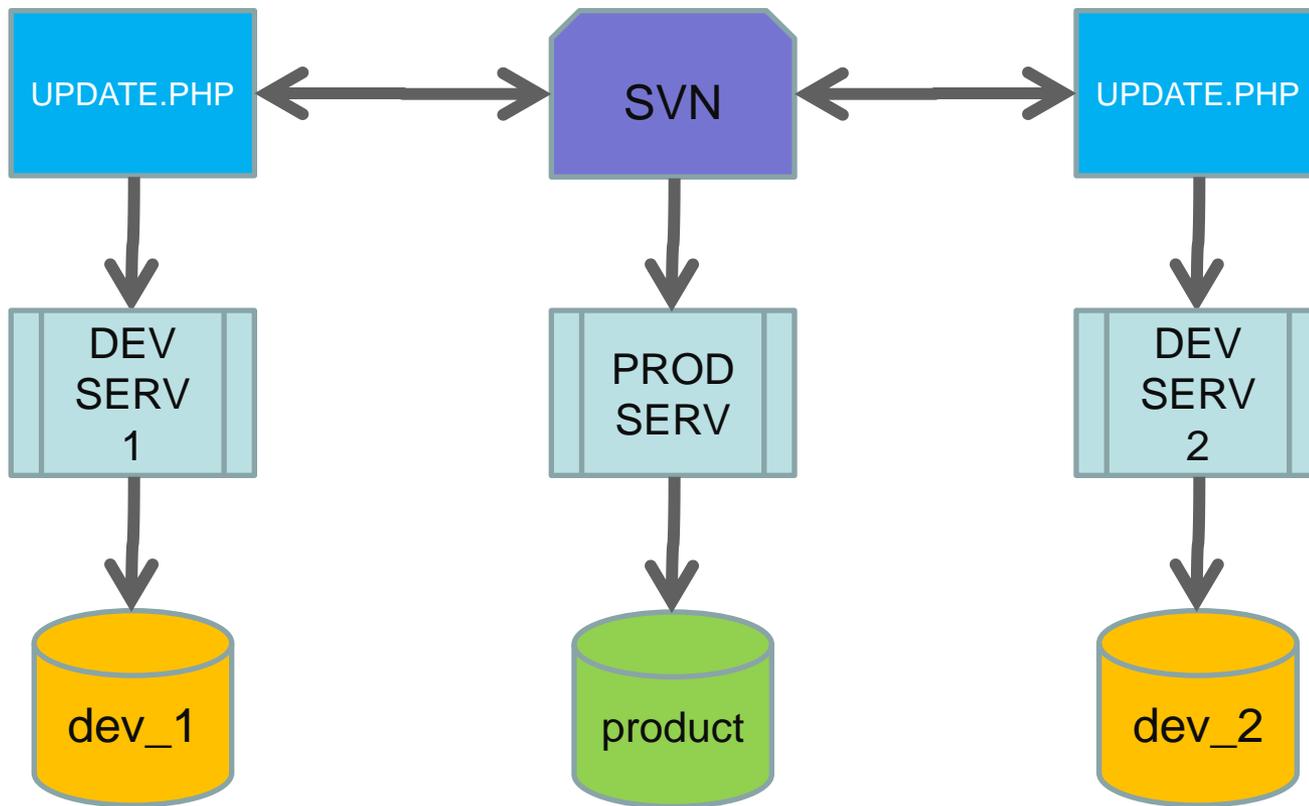
$scripts[1] = array(
    "CREATE TABLE `user` ...",
    "CREATE TABLE `post` ..."
);

...

$scripts[34] = array(
    "ALTER TABLE `user` ADD COLUMN ..."
);
```

- Скрипт с SQL-патчами.
- В БД хранится номер последнего патча.
- Выполняем новые патчи.
- Меняем версию БД.

## Два разработчика, много баз



## Два разработчика, много баз

- SVN-конфликты при одновременном создании одинаковых ревизий.



## Два разработчика, много баз II



### 1.SQL

```
CREATE TABLE `user` ...;  
CREATE TABLE `post` ...;
```

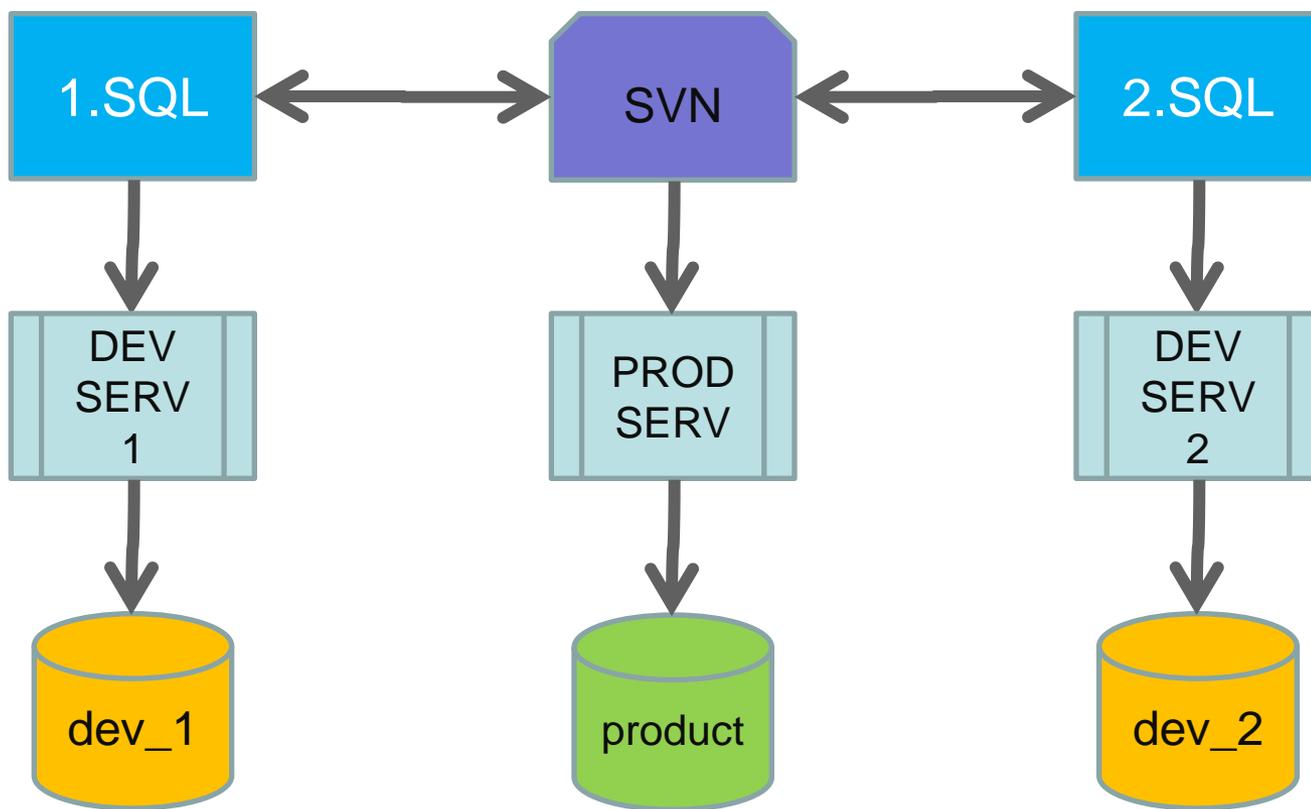
...

### 34.SQL

```
ALTER TABLE `user` ADD COLUMN ...;
```

- Отдельные пронумерованные SQL-файлы.
- В БД хранится номер последнего патча.
- Выполняем новые патчи.
- Меняем версию БД.

## Два разработчика, много баз II



## Два разработчика, много баз II

- SVN-конфликты в дереве при одновременном создании одинаковых файлов разрешаются переименованием файлов.
- Конфликты порядка изменений имеют место.



## Два разработчика, много баз III

### 20110511231531121.SQL

```
CREATE TABLE `user` ...;  
CREATE TABLE `post` ...;
```

...

### 20110514230417965.SQL

```
ALTER TABLE `user` ADD COLUMN ...;
```

- Отдельные пронумерованные SQL-файлы по TIMESTAMP.
- В БД хранятся номера обработанных файлов.
- Выполняем необработанные файлы.
- Помещаем в БД номера обработанных файлов.



## Два разработчика, много баз III

- Конфликты порядка изменений имеют место.



## Нерешенная проблема

Как избавиться от конфликта  
порядка изменений?

Кроме регулярного обновления

## Резюме

```
<?php
```

```
$filename = date("YmdHisu");  
$content = "-- " . date("d.m.Y H:i") . PHP_EOL;  
$content .= "-- @author " . PHP_EOL;  
$content .= "-- @comment " . PHP_EOL;  
$content .= PHP_EOL;  
file_put_contents($filename . ".sql", $content);  
echo "Revision file " . $filename . ".sql created" . PHP_EOL;
```

Решение под PHP

Ruckusing

<http://code.google.com/p/ruckusing/>

## Ruckusing howto

```
> php generate.php create_user_table
```

```
db/migrate/20110602193549_CreateUserTable.php
```

## db/migrate/20110602193549\_CreateUserTable.php

```
<?php
class CreateUserTable extends Ruckusing_BaseMigration {
    public function up() {
        $t = $this->create_table("users");
        $t->column("first_name", "string");
        $t->column("last_name", "string");
        $t->column("email", "string", array('limit' => 128));
        $t->finish();
        $this->add_index("users", "email", array('unique' => true));
    }
    public function down() {
        $this->drop_table("users");
    }
}
?>
```

## Process

> php main.php db:migrate

## Контактные данные

Михаил КечинОВ  
sam@mkechinov.ru  
mkechinov.ru  
+7 950 0099233