

CRUD Operations



Maximilian Schwarzmüller

AUTHOR

@maxedapps www.mschwarzmueller.com



Overview



Quickly setup database with Migrations

Dive into Eloquent to model relations

Use Eloquent to work with data

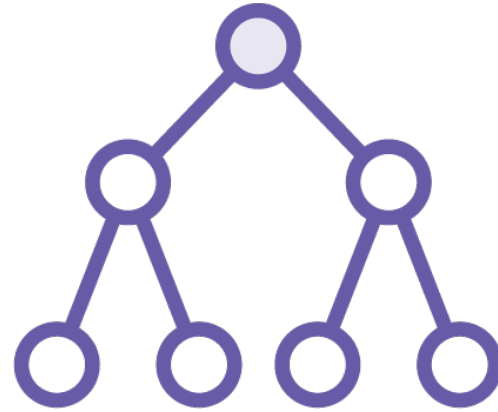


Laravel (MySQL) Database Support



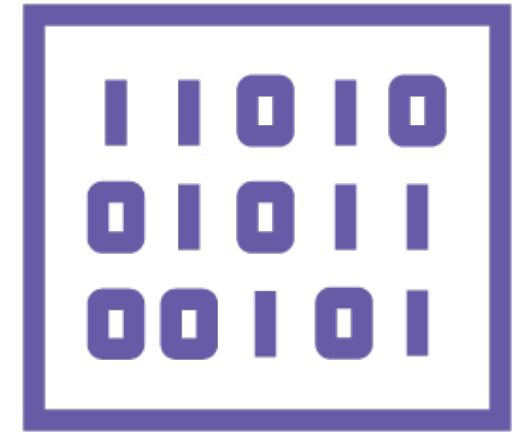
Create Tables

Create and configure tables in PHP files, with migrations



Configure Relations

Define relations between PHP models

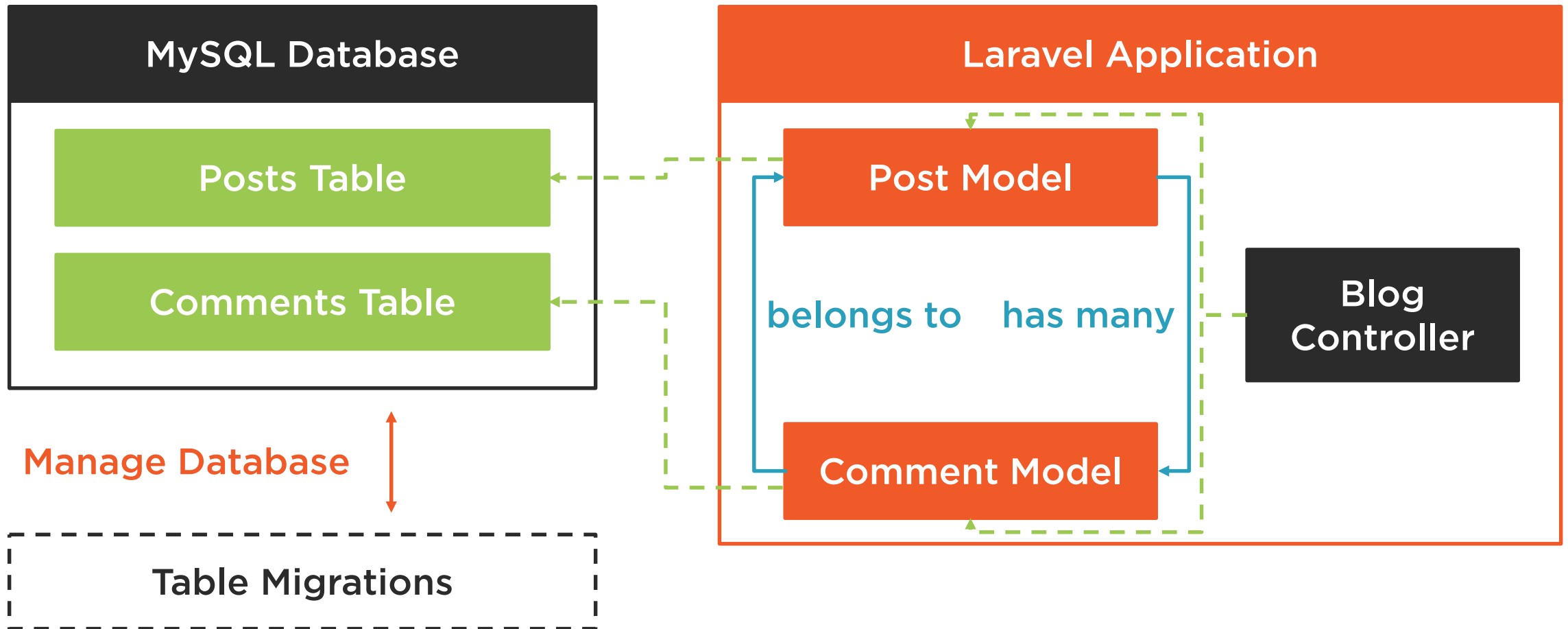


Work with Data

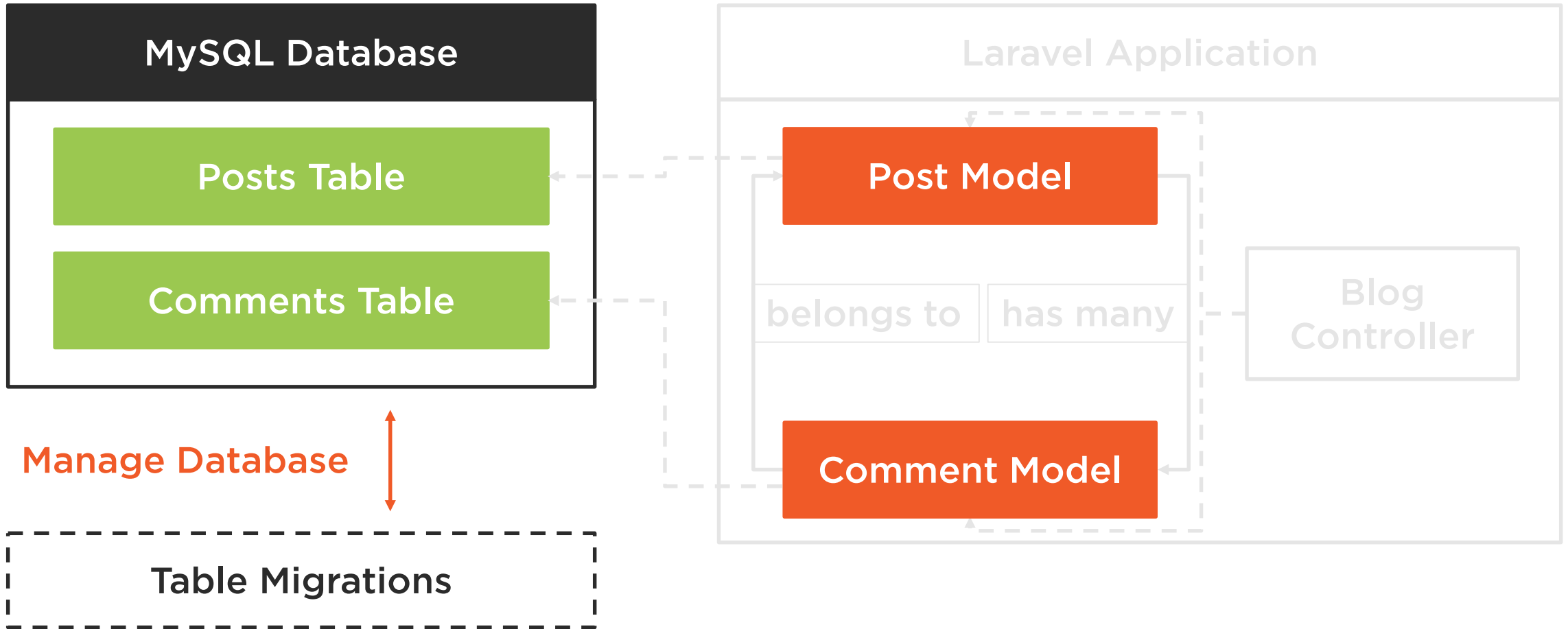
Store, fetch and manipulate data through PHP models



Using Databases – Eloquent & Migrations



Database Structure - Migrations



```
php artisan make:model Post -m
```

```
php artisan make:migration PostTableMigration
```

Create Models & Migrations

Creates model file (PHP file) in the app folder

Creates migration file to configure the database table in the database folder

Alternative: Only create migration file to configure a database table



Configure Database Table

```
public function up()  
{  
    Schema::create('posts', function (Blueprint $table) {  
        $table->increments('id');  
        $table->timestamps();  
        $table->string('title');  
        $table->text('content');  
    });  
}
```



Manage Migrations

```
php artisan migrate // Create database table
```

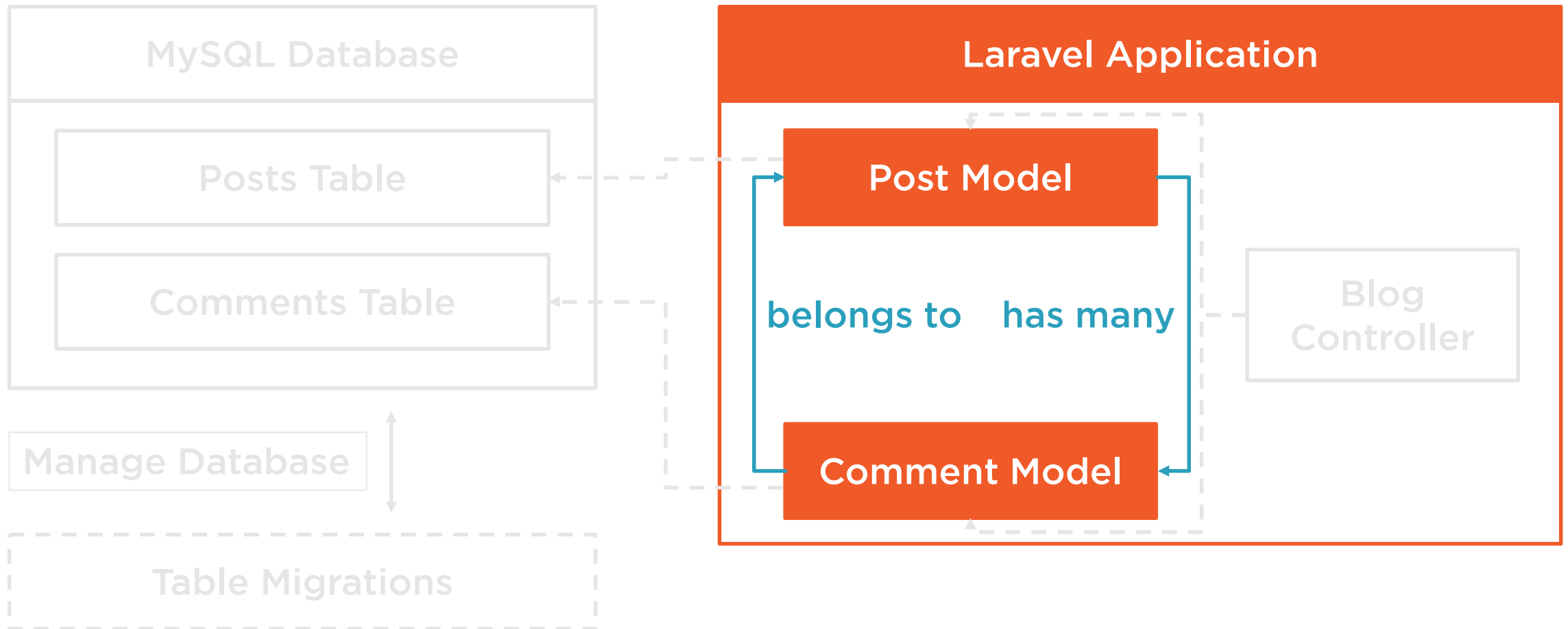
```
php artisan migrate:rollback // Rollback latest migrations
```

```
php artisan migrate:reset // Rollback all migrations
```

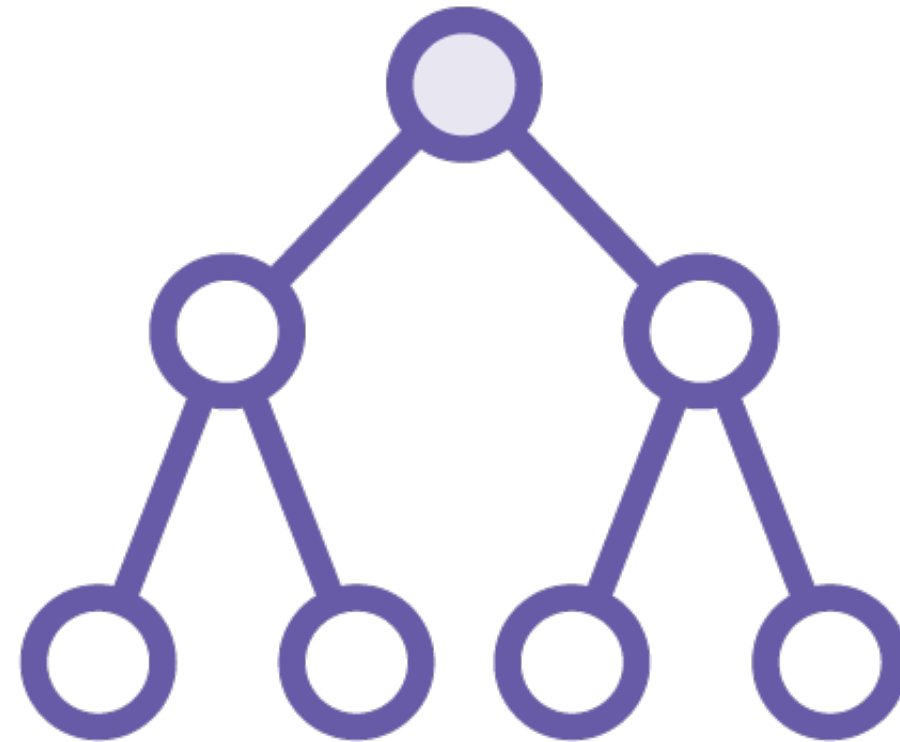
```
php artisan migrate:refresh // Rollback & migrate again
```



Using Databases – Relations



One-to-one
One-to-many
Many-to-many
...



```
public function comments()  
{  
    return $this->hasMany( 'App\Comment' );  
}
```

Defining Relations

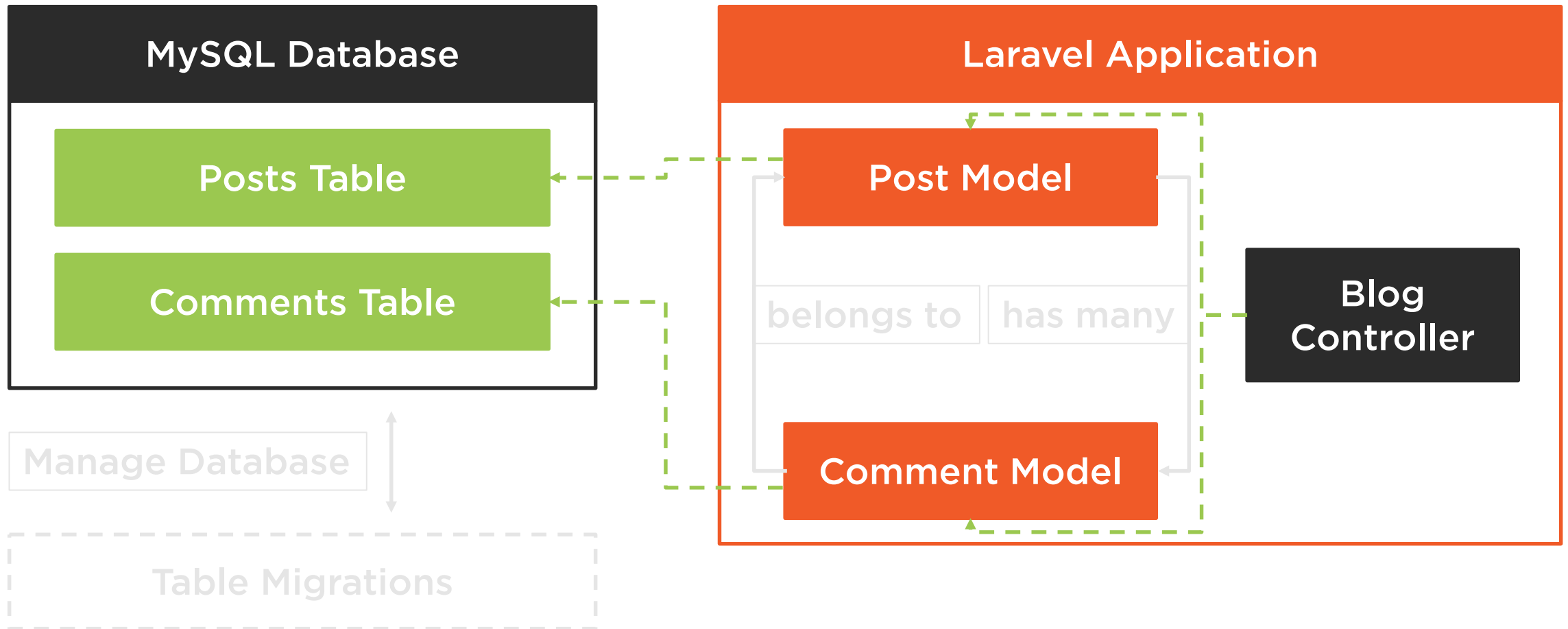
Relations are defined as functions in Eloquent models

Assumptions:

- Foreign key is snake_case version of model name, suffixed with _id (e.g. post_id)
- Many-to-many: Pivot table is derived from alphanumerical order of related models (e.g. comment_post)



Using Databases – Interacting with Data



Interact with Data

```
// Single entry
```

```
$post = new Post(data);
```

```
$post->save();
```

```
$post->delete();
```

```
// Relation
```

```
$post = Post::find(1);
```

```
$comment = new Comment(data);
```

```
$post->comments()->save($comment);
```



Interact with Data (Many-to-many)

```
// Inserts record into intermediate table
```

```
$post = Post::find(1);
```

```
$user = new User(data);
```

```
$post->users()->attach($user);
```

```
$post->users()->detach();
```



Summary



Utilize migrations to quickly setup database tables

Take advantage of Eloquent models to create relations and interact with data