Chapter 9. Performance Tuning

In this chapter, we will cover the following topics:

• Following best practices

• Speeding up session handling

• Using cache dependencies and chains

• Profiling an application with Yii

• Leveraging HTTP caching

• Combining and minimizing assets

• Running Yii2 on HHVM

Yii is one of the fastest frameworks available. Nevertheless, when developing and deploying an application, it is good to have some extra performance for free, and to follow the best practices for the application itself. In this chapter, you will see how to configure Yii to gain extra performance. In addition, you will learn some best practices for developing an application that will run smoothly until you have very high loads.

Following best practices

In this recipe, you will see how to configure Yii2 for the best performance and some additional principles of building responsive applications. These principles are both general and Yii-related. Therefore, we will be able to apply some of these even without using Yii2.

Getting ready

Create a new yii2-app-basic application using the Composer package manager, as described in the official guide at [http://www.yiiframework.com/doc-2.0/gurdestart-installation.html](http://www.yiiframework.com/doc-2.0/guidestart-installation.html).

How to do it...

1. Update your PHP to the latest stable version. Major releases of PHP may bring significant performance improvements. Turn off the debug mode and set the prod environment. This can be done by editing web/index .php as follows:

defined('YII\_DEBUG') or define('YII\_DEBUG', false); defined('YII\_ENV') or define('YII\_ENV', 'prod');

**Note**

Note: In the yii2-app-advanced application skeleton, you can use the shell command php init and opt production environment for loading optimized index. php and configuration files.

2. Enable the cache component:

'components' => [ 'cache' => [