

Happyr

Symfony 3.0

Tobias Nyholm

Happyr

Symfony 3

Lots of new features

- New LDAP component (csarrazi, lyrixx) (14602)
- New PropertyInfo component (dunglas) (15858 and 15966)
- Foundation form layout integration (totophe) (12587)
- Decoupled Twig from the Templating Component (fabpot) (15970, 16271)
- Warmup twig templates in non-standard paths (kbond) (14764)
- Added autowiring capabilities (dunglas) (15613)
- Deprecated the scope concept (dosten) (14984)
- Added a way to define the priority of service decoration (dosten) (15416)
- Added support for deprecated definitions (Taluu) (15491)
- Added logging of unused tags (Marmelatze, fabpot) (15963)
- Implemented resettable containers (stof) (15185)
- Automatically process extensions when they implement CompilerPassInterface (WouterJ) (13761)
- Deprecated non-escaped in double-quoted strings when parsing (fabpot) (16201)
- Deprecated usage of @ and ` at the beginning of an unquoted string (fabpot) (16285)
- Deprecated unquoted indicator characters (xabbuh) (16433)
- Added a progress indicator helper (kbond) (16409)
- Added table columns styles (MAXakaWIZARD) (14044)
- Deprecated the Shell Console class (fabpot) (15892)
- Add domain exceptions to replace generic exceptions (GromNaN) (14894)
- Use readline for user input when available #DX (michaelperrin) (15382)
- Made the exception output visible even in quiet mode (Seldaek) (15773)
- Converted Output::write's type to an options arg where verbosity can be passed in as well (Seldaek) (15772)
- Added casters for Generator, ReflectionGenerator, ReflectionType, OuterIterator, SplFileInfo, and pgsql resources (nicolas-grekas, lyrixx) (16325, 15787, 15555, 14424)
- Dump PHP+Twig code excerpts in backtraces (nicolas-grekas) (15838)
- Mock clock on @group time-sensitive annotations (nicolas-grekas) (16194)
- Added EventDispatcher::getListenerPriority() (fabpot) (16198)
- Added priorities to the debug:event-dispatcher command (Seldaek) (14563)
- New Guard Authentication System (weaverryan) (14673)
- Removed security-acl from core (iltar) (15013)
- Configuring a user checker per firewall (iltar) (14721)
- Deprecated supportsAttribute and supportsClass methods (WouterJ) (15151)
- Renamed Token#getKey() to getSecret() (WouterJ) (15141)
- Deprecated the SecureRandom class (pierredup) (15879)
- Changed dumpFile to allow dumping to streams (markchalloner, pierredup) (16156)
- Redesigned the Symfony Profiler (javiereguiluz) (15523)
- Redesigned the web debug toolbar (javiereguiluz) (15160)
- Added filter links in search results (Rvanlaak) (16344)
- Deprecated import/export commands (fabpot) (15709)
- Improved page for logs (hason) (13324)
- Removed profiler storages (javiereguiluz) (15944)
- Added a non-static API for the CssSelector component (stof) (15934)
- Added choice_translation_domain option (aitboudad) (15301)
- Deprecated FormTypeInterface::getName() and passing of type instances (webmozart) (15079)
- Deprecated "cascade_validation" (webmozart) (15019)
- Added the 'range' FormType (jaytaph) (12067)
- Deprecated read_only option (Tobion) (14403)
- Added XLIFF 2.0 support (xpHERE, aitboudad) (15717)
- Allowed get file content without writing in file. (aitboudad) (15786)
- Deprecated format method in favor of formatCatalogue. (aitboudad) (15699)
- Added parameters to DataCollectorTranslator (damienalexandre) (15139)
- Added an option to specify additional translation loading paths (Seldaek) (14561)
- Deprecated getMessages in favor of getCatalogue. (aitboudad) (14546)
- Added an --all option to the debug:translation command (sgehrig) (14320)
- Configurable Serializer name converter (dunglas) (14987)
- Support for array denormalization (derrabus) (14756)
- Deprecated methods inherited from SplObjectStorage (stof) (15907)
- Deprecated loading multiple documents in the same crawler (stof) (16057)
- Prevented adding non-DOMElement elements in DomCrawler (stof) (16058)
- Added BufferingLogger for errors that happen before a proper logger is configured (nicolas-grekas) (15521)
- Added ExceptionHandler::getHtml() to expose the full HTML of an exception (fabpot) (15710)
- Deprecated adapters and related classes (nicolas-grekas) (15805)
- Added a micro kernel (fabpot) (15990)
- Added UserLoaderInterface for loading users through Doctrine. (mtrojanowski) (15947)
- Rely on iconv and symfony/polyfill-* (nicolas-grekas) (16317)
- Added a new ClassCache cache warmer (tucksauN) (16263)
- Deprecate finding deep items in request parameters (xabbuh) (16007)
- is_granted no longer raise an exception if the token storage is empty (lyrixx) (15953)
- Easier Custom Authentication errors (weaverryan) (15882)
- Updating AbstractVoter so that the method receives the TokenInterface (weaverryan) (15870)
- Made template shortcuts be usable without Templating component (Koc) (15620)
- Added an option to force the built-in web server startup (xabbuh) (15134)
- Added a Doctrine cache service definition for validator mapping (jakzaL) (14429)

About me

- Tobias Nyholm, @tobiasnyholm
- Happyr.com
- Co-host of Sound of Symfony podcast
- Your organizer

Happyr

Backwards Compatibility

Semver

Happyr

Backwards Compatibility

3 . 0 . 0

Major

Minor

Patch

Happyr

Backwards Compatibility

If you use a Symfony class, it will not break BC between minor versions

Detect usage of deprecated code

@deprecated

Symfony 3

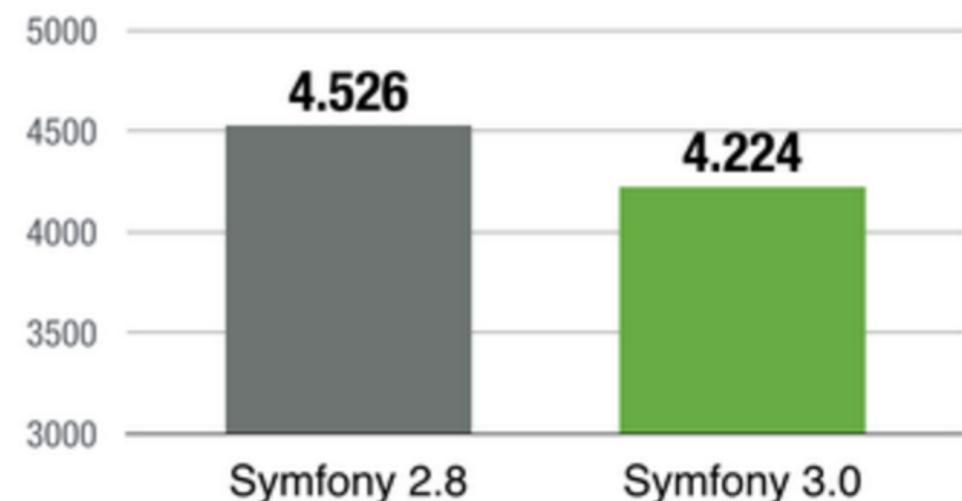
Symfony Deprecation Progress Report

Symfony 3, which will be released in November 2015, will deprecate lots of existing features and code.

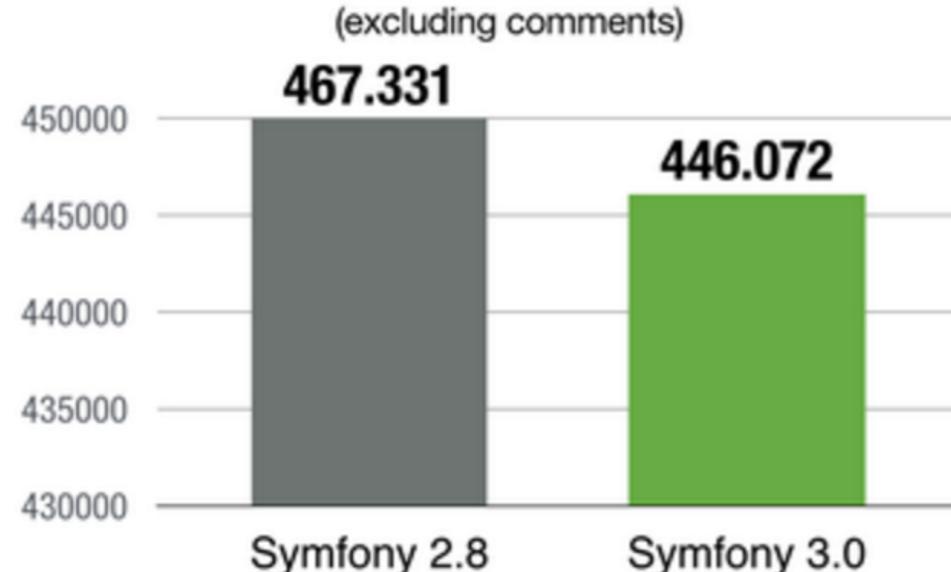
This report illustrates the progress of the deprecation progress in terms of files and lines of code as of 6th October 2015.

Metrics were gathered with the **cloc** command line tool found at <https://github.com/AIDanial/cloc>

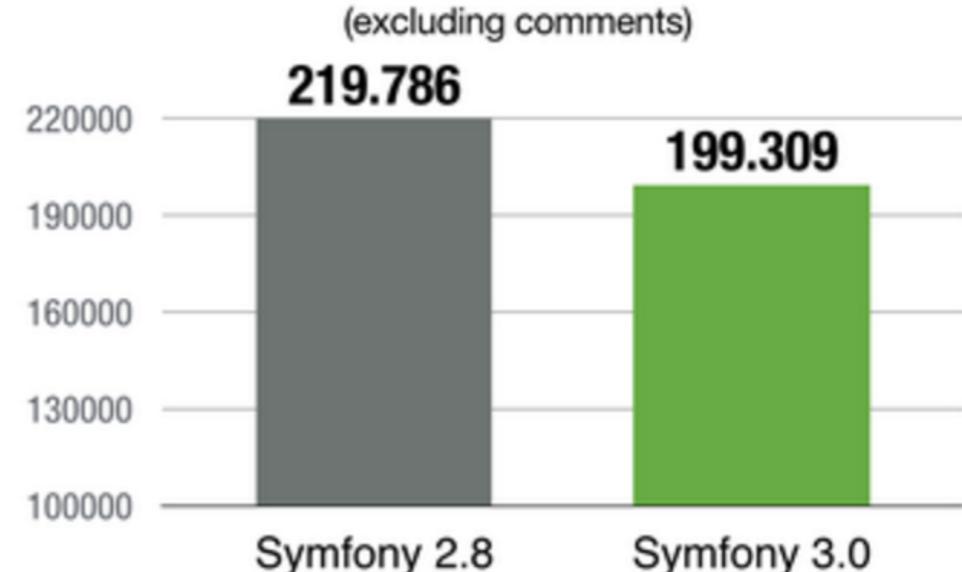
Total number of **files**



Total number of **lines of code**



Total number of **lines of PHP code**



What is new in SF3?

Happyr

Nothing

What is removed?

Removed the Request service

```
my_service:  
    class: Acme\DemoBundle\Service\WebsericeUserProvider  
    scope: request  
    calls:  
        - [setRequest , ["@request"]]
```

```
my_service:  
    class: Acme\DemoBundle\Service\WebsericeUserProvider  
    arguments: ["@request_stack"]
```

Removed the DI scope

Removed ACL from core

You should use Voters

Templating

Symfony

Abstraction layer

Blade

Twig

PHP

Happyr

Templating

Symfony

Twig

What is new?

(SF2.8)

Happyr

Redesign of debug toolbar

The screenshot displays the Happyr debug toolbar, which has been redesigned. It features a sidebar on the left containing key information:

Status	200 OK
Controller	UserController ::indexAction
Route name	_activity_index
Has session	yes

Below the sidebar is a horizontal bar with various icons and metrics:

- sf 2.3.35
- php
- 27cf7f
- 200 UserController :: indexAction
- 3 errors
- 1867 ms
- 35.5 MB
- tobiasnyholm
- 14 sessions

At the bottom of the toolbar is another set of metrics:

Errors	0
Deprecated Calls	33
Silenced Errors	0

Other items at the bottom include:

- 200 @ _admin_dashboard 1503 ms 65.5 MB
- 33 errors
- 10 silenced errors
- tobias@happyr.com
- 254 ms
- 5 in 115.84 ms
- Happyr 0
- sf 2.8.0
- X

Happyr

Redesign of web profiler



Search on Symfony website

OK

CONFIG [View last 10](#) Profile for: GET http://funka.dev/app_dev.php/company/219/show by 127.0.0.1 at Thu, 17 Dec 2015 09:24:52 +0100

REQUEST

EXCEPTION

EVENTS

LOGS 3

TIMELINE

ROUTING

E-MAILS

SECURITY

DOCTRINE 136 347 MS

MINIMIZE

SEARCH

IP

Timeline

Total time	1707 ms
Initialization time	231 ms
Threshold	1 ms

Main Request - 1476 ms

default section event_listener event_listener_loading template doctrine propel child_sections

kernel.request ~ 60 ms / ~ 16.8 MB

Symfony\Component\HttpKernel\EventListener\ProfilerListener ~ 57 ms / ~ 40.5 MB

Symfony\Bundle\FrameworkBundle\EventListener\SessionListener ~ 4 ms / ~ 14.8 MB

Symfony\Component\HttpKernel\EventListener\RouterListener ~ 9 ms / ~ 15.2 MB

Symfony\Component\Security\Http\Firewall ~ 41 ms / ~ 16.8 MB

doctrine ~ 46 ms / ~ 30.2 MB

controller.get_callable ~ 3 ms / ~ 17 MB

kernel.controller ~ 107 ms / ~ 19 MB

Lsw\SecureControllerBundle\Security\ControllerListener ~ 5 ms / ~ 17.2 MB

Sensio\Bundle\FrameworkExtraBundle\EventListener\ParamConverterListener ~ 99 ms / ~ 19 MB

[Last 10](#)[Latest](#) [Search](#) [Request / Response](#) [Performance](#) [Forms](#) [Exception](#) [Logs 275](#) [Events](#) [Routing](#) [Translation](#) [Security](#) [Twig](#) [Doctrine](#) [E-Mails](#) [Happy API 0](#) [Configuration](#)

Performance metrics

2685 ms

228 ms

1

206 ms

79.50 MB

Total execution time

Symfony initialization

Sub-Requests

Sub-Requests time

Peak memory usage

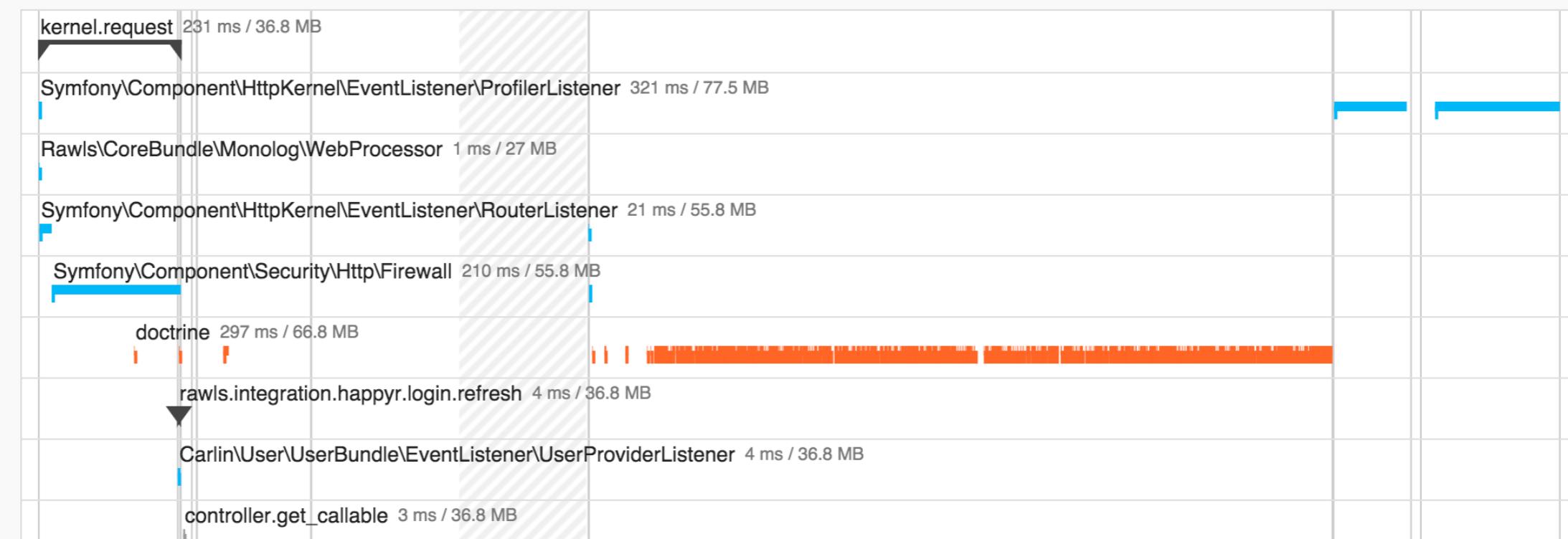
Execution timeline

Threshold

1

ms (timeline only displays events with a duration longer than this threshold)

Main Request 2457 ms

 [default](#) [section](#) [event_listener](#) [event_listener_loading](#) [template](#) [doctrine](#) [propel](#)

Happyr

DX: is_granted

Guard

Authentication

Who are you?

Authorization

Can you access X?

Authentication

1. Grab info from the request
2. Load a User
3. Validate the credentials
4. On success
5. On authentication failure
6. How to ask the user to login

Introducing Guard

```
/** The interface for all "guard" authenticators. ...*/
interface GuardAuthenticatorInterface extends AuthenticationEntryPointInterface
{
    /** Get the authentication credentials from the request and return them ...*/
    public function getCredentials(Request $request);

    /** Return a UserInterface object based on the credentials. ...*/
    public function getUser($credentials, UserProviderInterface $userProvider);

    /** Returns true if the credentials are valid. ...*/
    public function checkCredentials($credentials, UserInterface $user);

    /** Create an authenticated token for the given user. ...*/
    public function createAuthenticatedToken(UserInterface $user, $providerKey);

    /** Called when authentication executed, but failed (e.g. wrong username password). ...*/
    public function onAuthenticationFailure(Request $request, AuthenticationException $exception);

    /** Called when authentication executed and was successful! ...*/
    public function onAuthenticationSuccess(Request $request, TokenInterface $token, $providerKey);

    /** Does this method support remember me cookies? ...*/
    public function supportsRememberMe();
}
```

Happyr

Auto wiring

```
namespace AppBundle\Service;

class Service1
{



namespace AppBundle\Service;

use AppBundle\Service\Service1;

class Service2
{
    private $service1;

    public function __construct(Service1 $service1)
    {
        $this->service1 = $service1;
    }
}
```

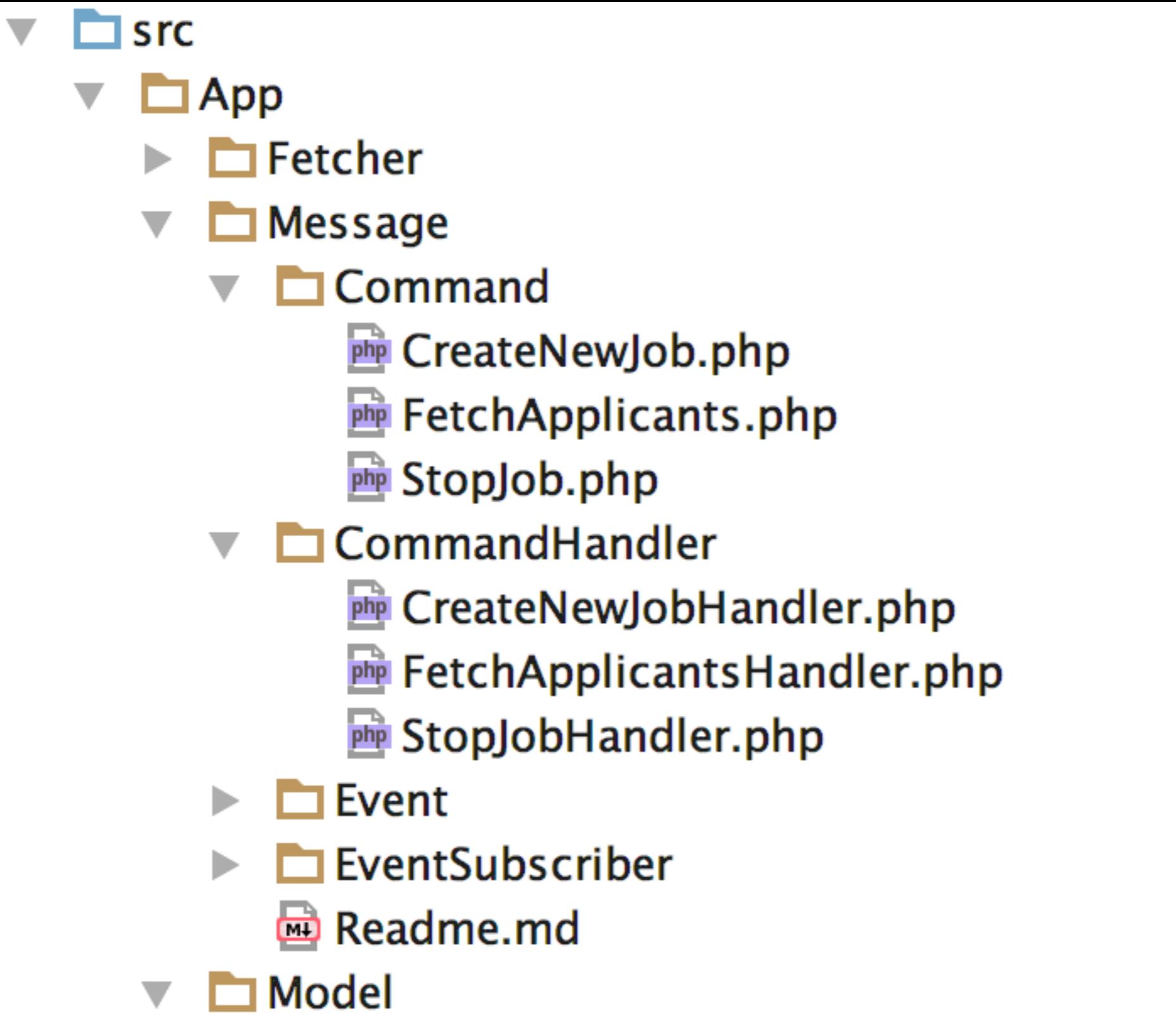
Auto wiring

```
# app/config/services.yml
services:
    service1:
        class: AppBundle\Service\Service1

    service2:
        class: AppBundle\Service\Service2
        arguments: ['@service1']
```

```
# app/config/services.yml
services:
    service2:
        class: AppBundle\Service\Service2
        autowire: true
```

CQRS example



Cache warmup

```
$container = $this->getContainer();
$dir = $container->getParameter('kernel.cache_dir');
$container->get('kernel.class_cache.cache_warmer')->warmUp($dir);
```

Console style

```
// alternative 1
$formatter = $this->getHelperSet()->get('formatter');
$formattedBlock = $formatter->formatBlock('Lorem Ipsum Dolor Sit Amet', '', true);
$output->writeln($formattedBlock);

// alternative 2
$title = 'Lorem Ipsum Dolor Sit Amet';
$output->writeln('<info>' . $title . '</info>');
$output->writeln(str_repeat('=', strlen($title))));

// alternative 3
$output->writeln('');
$output->writeln('<info>Lorem Ipsum Dolor Sit Amet</info>');
$output->writeln('<info>-----</info>');
$output->writeln('');
```

```
$io = new SymfonyStyle($input, $output);

// common output elements
$io->title(string $message);
$io->section(string $message);
$io->text(string|array $message);
$io->comment(string|array $message);

// more advanced output elements
$io->note(string|array $message);
$io->caution(string|array $message);
$io->listing(array $elements);
$io->table(array $headers, array $rows);

// ask for user's input
$io->ask(string $question, string|null $default = null, callable|null $validator = null);
$io->askHidden(string $question, callable|null $validator = null);
$io->confirm(string $question, bool $default = true);
$io->choice(string $question, array $choices, string|int|null $default = null);

// display the result of the command or some important task
$io->success(string|array $message);
$io->error(string|array $message);
$io->warning(string|array $message);
```

Micro kernel

- Symfony is difficult
- Symfony is huge
- Symfony is complex

```
use Symfony\Bundle\FrameworkBundle\Kernel\MicroKernelTrait;
use Symfony\Component\Config\Loader\LoaderInterface;
use Symfony\Component\DependencyInjection\ContainerBuilder;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\HttpFoundation\Request;
use Symfony\Component\HttpKernel\Kernel;
use Symfony\Component\Routing\RouteCollectionBuilder;

class MicroKernel extends Kernel {
    use MicroKernelTrait;

    public function registerBundles() {
        return array(new Symfony\Bundle\FrameworkBundle\FrameworkBundle());
    }

    protected function configureRoutes(RouteCollectionBuilder $routes) {
        $routes->add('/', 'kernel:indexAction', 'index');
    }

    protected function configureContainer(ContainerBuilder $c, LoaderInterface $loader) {
        $c->loadFromExtension('framework', ['secret' => '12345']);
    }

    public function indexAction() {
        return new Response('Hello World');
    }
}

$app = new MicroKernel('prod', false);
$app->loadClassCache();
$app->handle(Request::createFromGlobals())->send();
```

```
public function registerContainerConfiguration(LoaderInterface $loader)
{
    $loader->load(function (ContainerBuilder $container) use ($loader) {
        $container->loadFromExtension('framework', array(
            'router' => array(
                'resource' => 'kernel:loadRoutes',
                'type' => 'service',
            ),
        ));

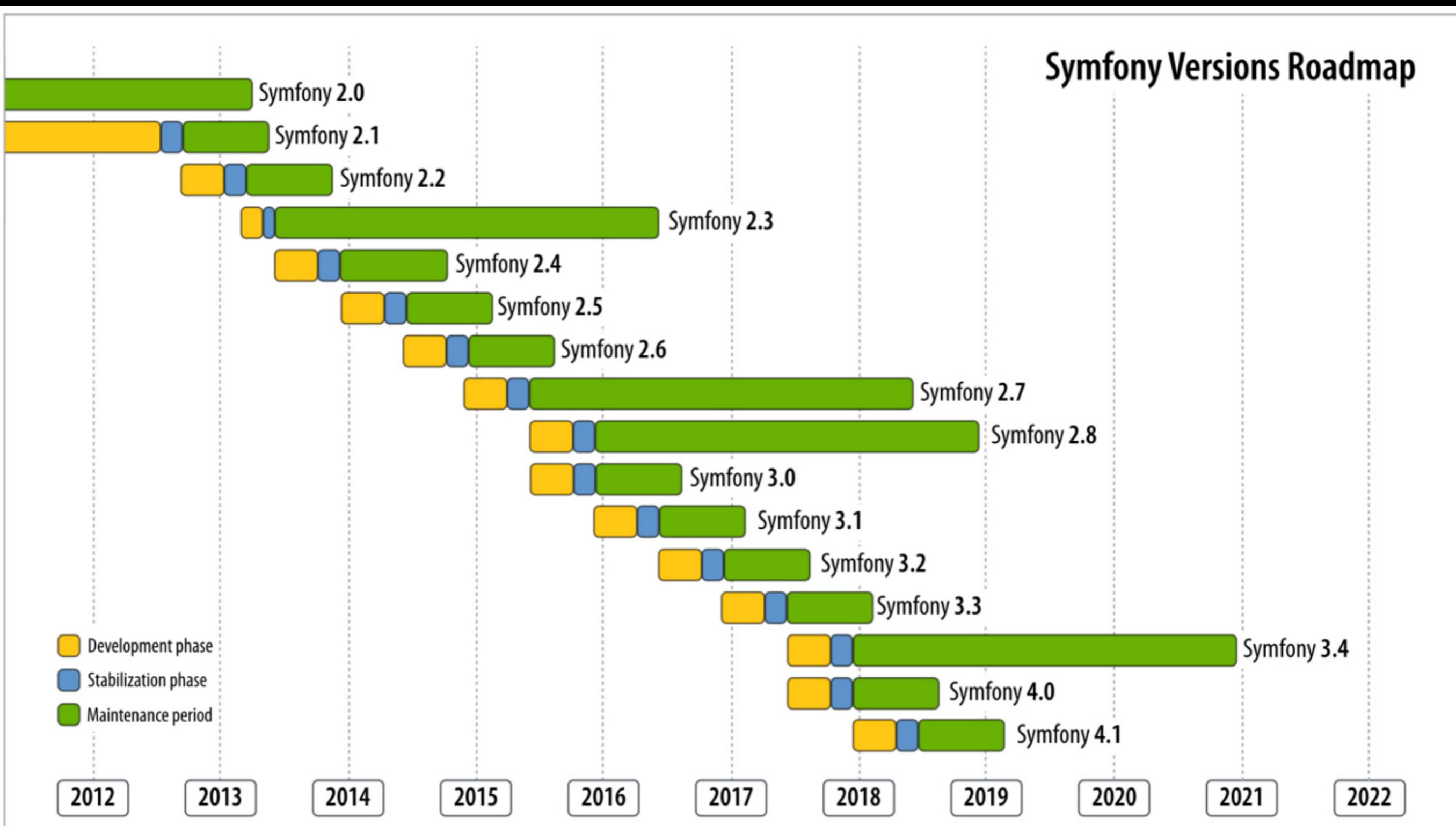
        $this->configureContainer($container, $loader);

        $container->addObjectResource($this);
    });
}

public function loadRoutes(LoaderInterface $loader)
{
    $routes = new RouteCollectionBuilder($loader);
    $this->configureRoutes($routes);

    return $routes->build();
}
```

Release schedule



Happyr

Thank you

@tobiasnyholm