

Das Zend Framework

Hält der Name, was er verspricht?

Sprecher: Carsten Lucke

- Software-Ingenieur bei der sd&m AG, München
- Aktives Mitglied der sd&m Client-Community
- PEAR und PECL Entwickler
- Schreibt für Fachzeitschriften/-bücher über diverse PHP-Themen

Agenda

- Einführung und Überblick über das ZF
- Framework oder Komponentenbibliothek?
- Fortgeschrittene Techniken
- Bewertung

Agenda

- > Einführung und Überblick über das ZF
Framework oder Komponentenbibliothek?
Fortgeschrittene Techniken
Bewertung

Was ist das ZF?

- Teils Application Framework, teils Component Library
- Open-Source
- Basiert auf PHP 5, objektorientiert entwickelt
- Liefert sauberen und stabilen Code mit sehr guter Dokumentation

Umfang des ZF

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...
- **MVC:** Zend_Controller, Zend_View, ...

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...
- **MVC:** Zend_Controller, Zend_View, ...
- **Auth:** Zend_Acl, Zend_Auth, Zend_Session

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...
- **MVC:** Zend_Controller, Zend_View, ...
- **Auth:** Zend_Acl, Zend_Auth, Zend_Session
- **Database:** Zend_Db, Zend_Db_Adapter, ...

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...
- **MVC:** Zend_Controller, Zend_View, ...
- **Auth:** Zend_Acl, Zend_Auth, Zend_Session
- **Database:** Zend_Db, Zend_Db_Adapter, ...
- **Mail, Formats & Search:** Zend_Json, Zend_Mail, Zend_Search, ...

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...
- **MVC:** Zend_Controller, Zend_View, ...
- **Auth:** Zend_Acl, Zend_Auth, Zend_Session
- **Database:** Zend_Db, Zend_Db_Adapter, ...
- **Mail, Formats & Search:** Zend_Json, Zend_Mail, Zend_Search, ...
- **i18n/l10n:** Zend_Date, Zend_Measure, Zend_Translate, ...

Umfang des ZF

- **Core Infrastructure:** Zend_Cache, Zend_Config, Zend_Log, ...
- **MVC:** Zend_Controller, Zend_View, ...
- **Auth:** Zend_Acl, Zend_Auth, Zend_Session
- **Database:** Zend_Db, Zend_Db_Adapter, ...
- **Mail, Formats & Search:** Zend_Json, Zend_Mail, Zend_Search, ...
- **i18n/l10n:** Zend_Date, Zend_Measure, Zend_Translate, ...
- **Web/Webservices:** Zend_Http, Zend_XmlRpc, Zend_Rest, Zend_Feed, Zend_Service_*, ...

Agenda

- Einführung und Überblick über das ZF
- > Framework oder Komponentenbibliothek?
- Fortgeschrittene Techniken
- Bewertung

Was bietet das ZF?

Framework-Teil:

Komponenten-Teil:

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten

Komponenten-Teil:

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung

Komponenten-Teil:

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung
- Requestverarbeitung: Controller/Action

Komponenten-Teil:

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung
- Requestverarbeitung: Controller/Action

Komponenten-Teil:

- Authentication & Authorization

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung
- Requestverarbeitung: Controller/Action

Komponenten-Teil:

- Authentication & Authorization
- Database

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung
- Requestverarbeitung: Controller/Action

Komponenten-Teil:

- Authentication & Authorization
- Database
- Mail, Formats & Search

Was bietet das ZF?

Framework-Teil:

- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung
- Requestverarbeitung: Controller/Action

Komponenten-Teil:

- Authentication & Authorization
- Database
- Mail, Formats & Search
- Internationalisierung

Was bietet das ZF?

Framework-Teil:

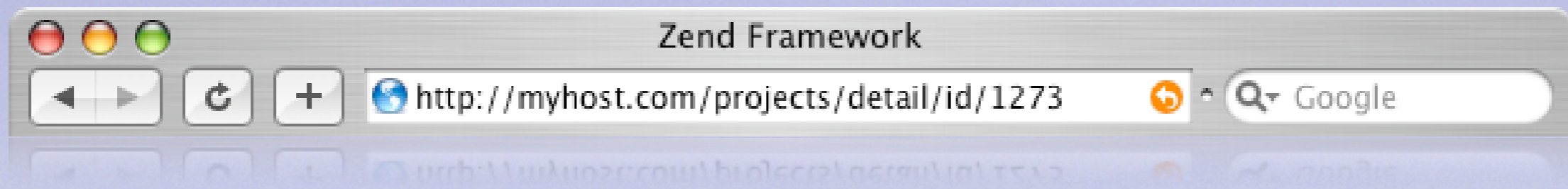
- MVC-Komponenten
- Front-Controller basierte MVC-Implementierung
- Requestverarbeitung: Controller/Action

Komponenten-Teil:

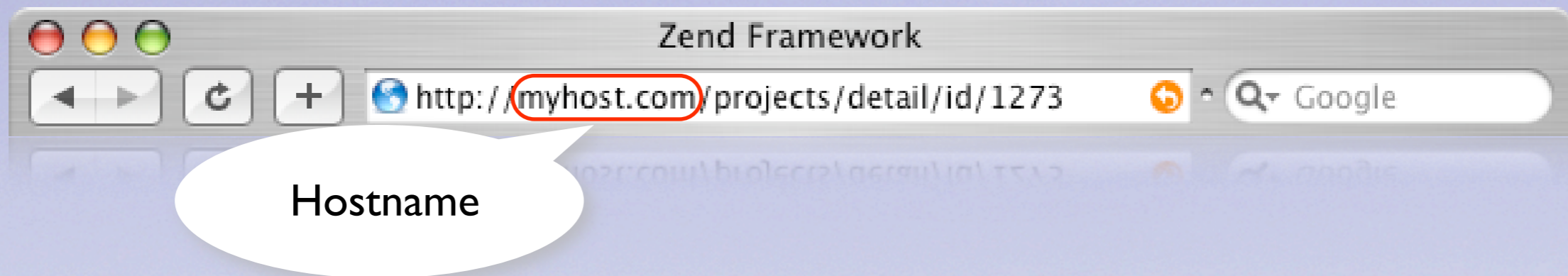
- Authentication & Authorization
- Database
- Mail, Formats & Search
- Internationalisierung
- Web / Webservices

Der „Framework-Anteil“ des Zend Framework

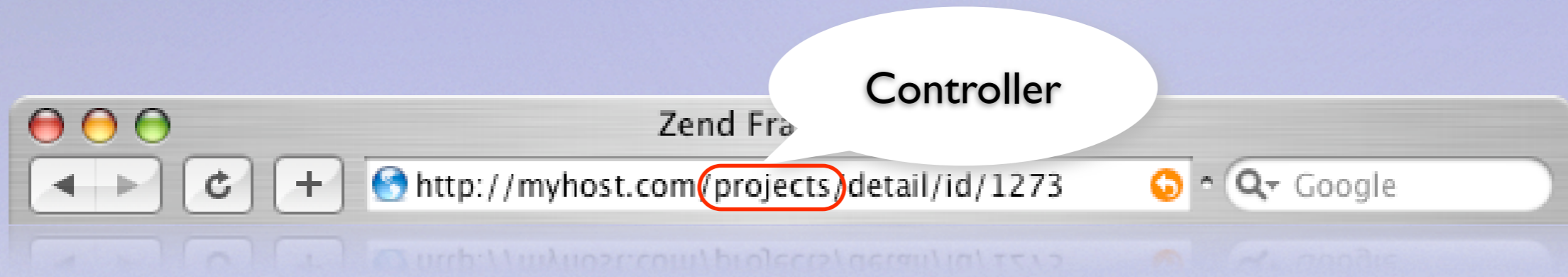
Requestverarbeitung des ZF



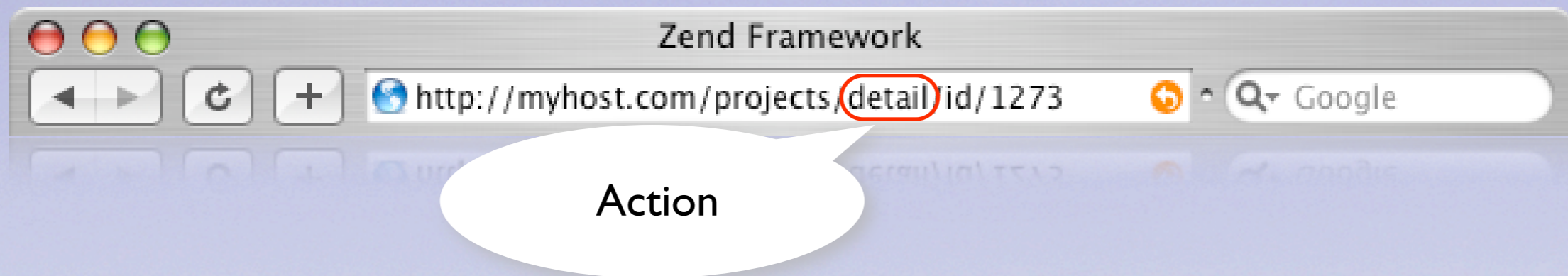
Requestverarbeitung des ZF



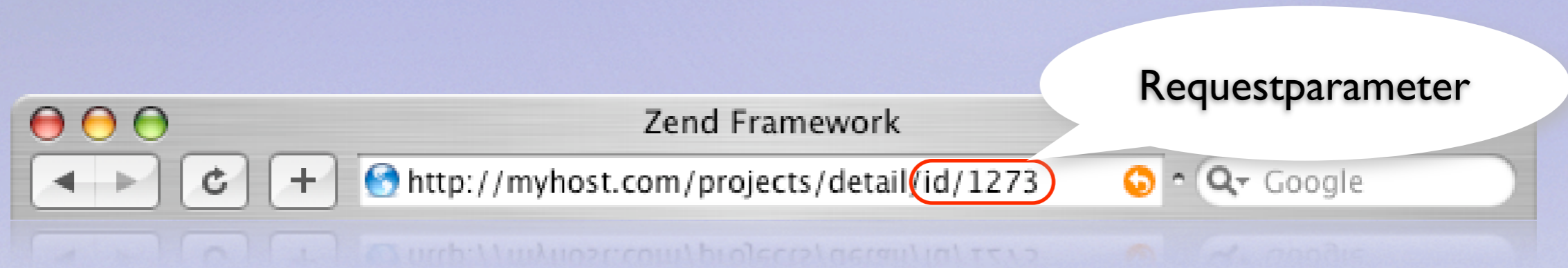
Requestverarbeitung des ZF



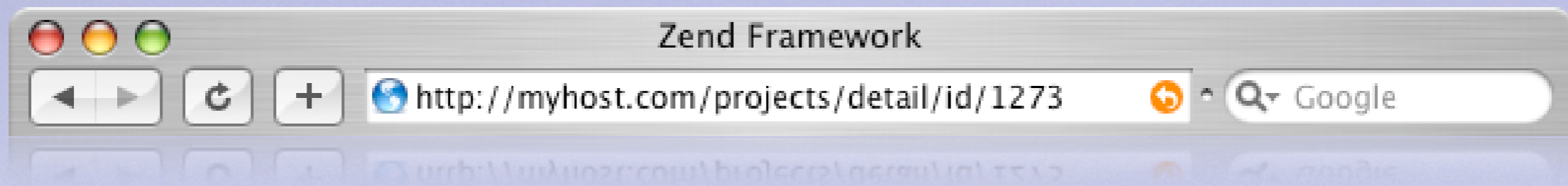
Requestverarbeitung des ZF



Requestverarbeitung des ZF



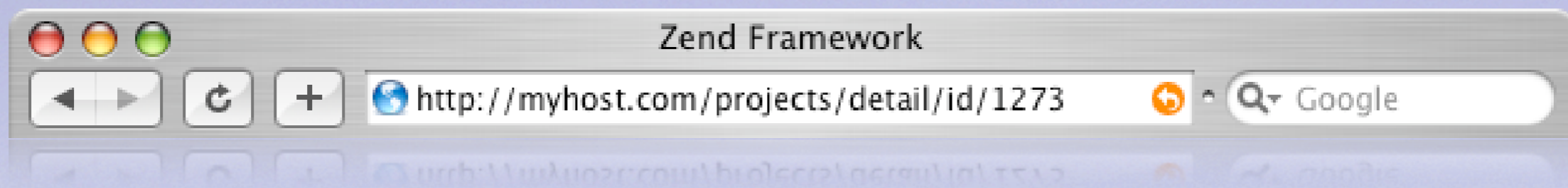
Requestverarbeitung des ZF



<http://myhost.com/>

IndexController :: indexAction

Requestverarbeitung des ZF



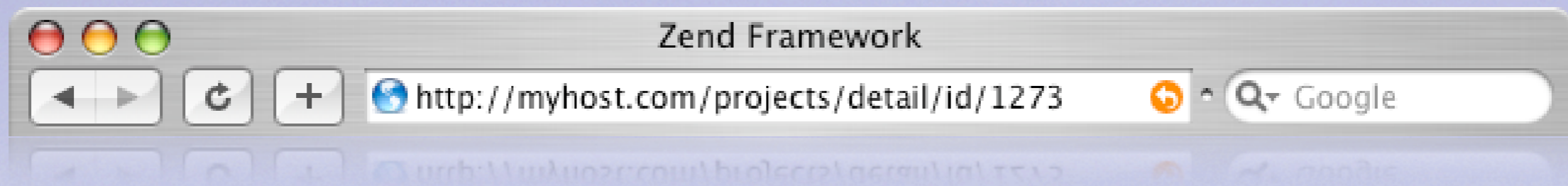
<http://myhost.com/>

IndexController :: indexAction

<http://myhost.com/projects/>

ProjectsController :: indexAction

Requestverarbeitung des ZF



<http://myhost.com/>

IndexController :: indexAction

<http://myhost.com/projects/>

ProjectsController :: indexAction

<http://myhost.com/projects/detail/>

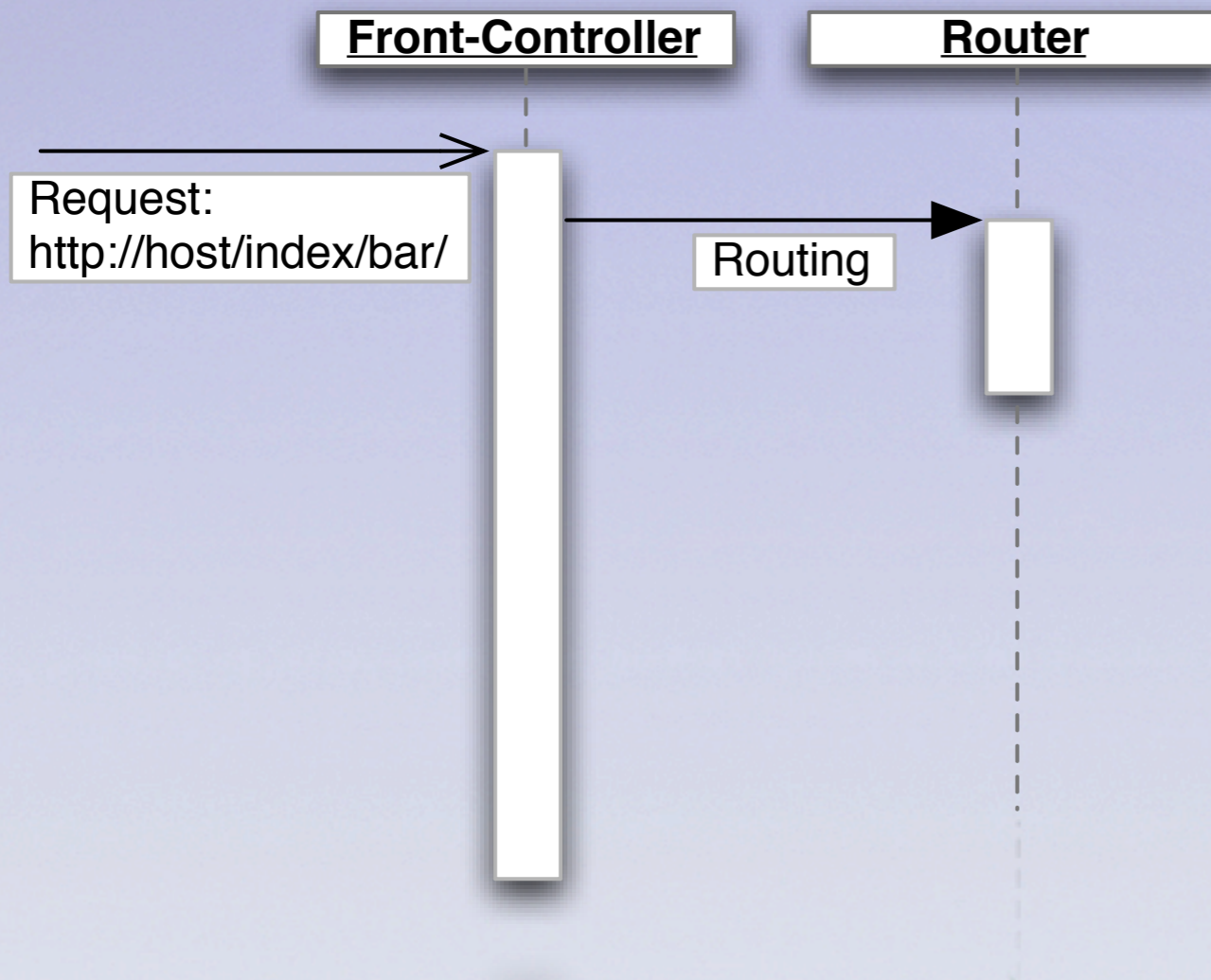
ProjectsController :: detailAction

Requestlebenszyklus des ZF

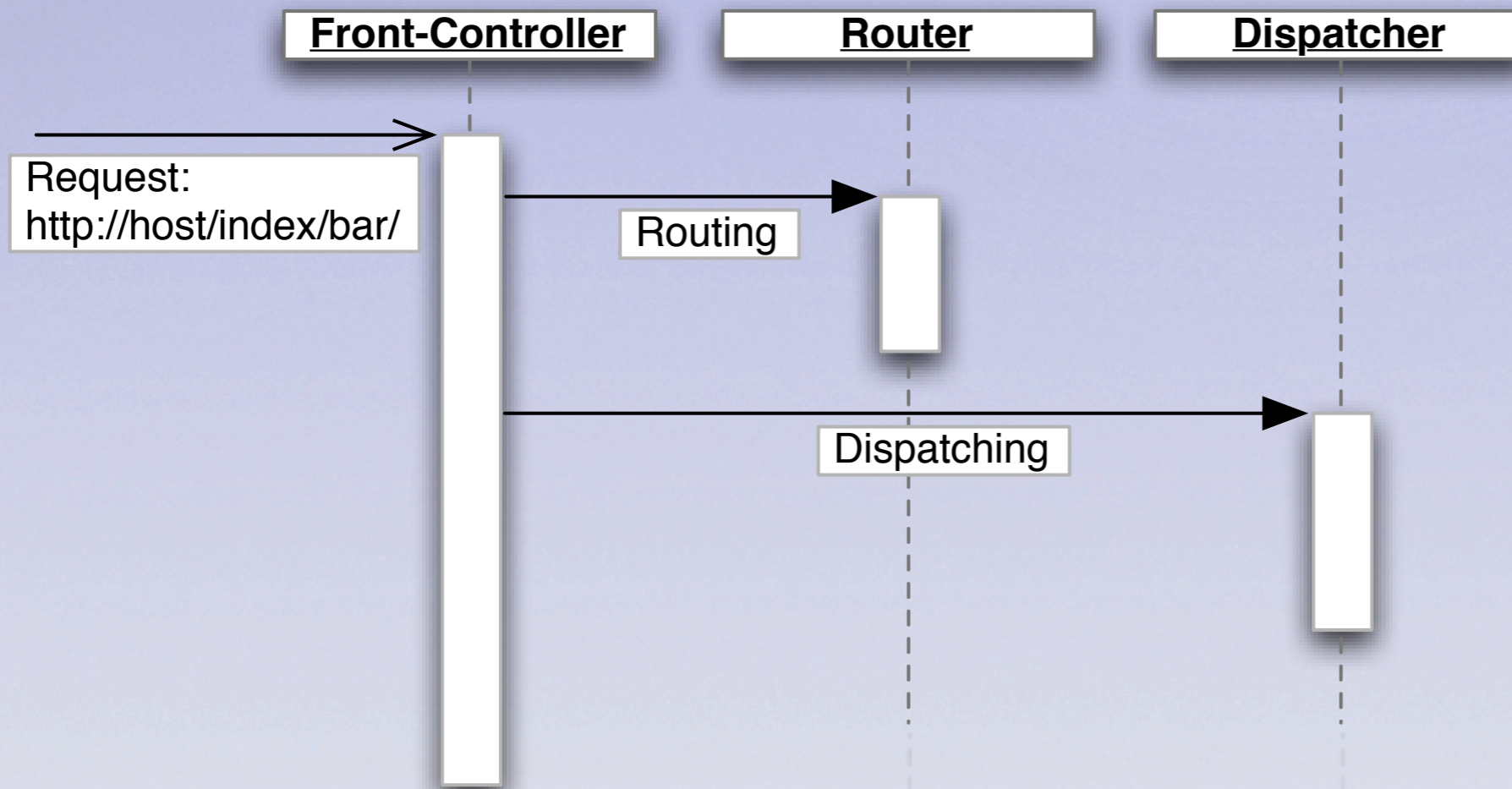
Requestlebenszyklus des ZF



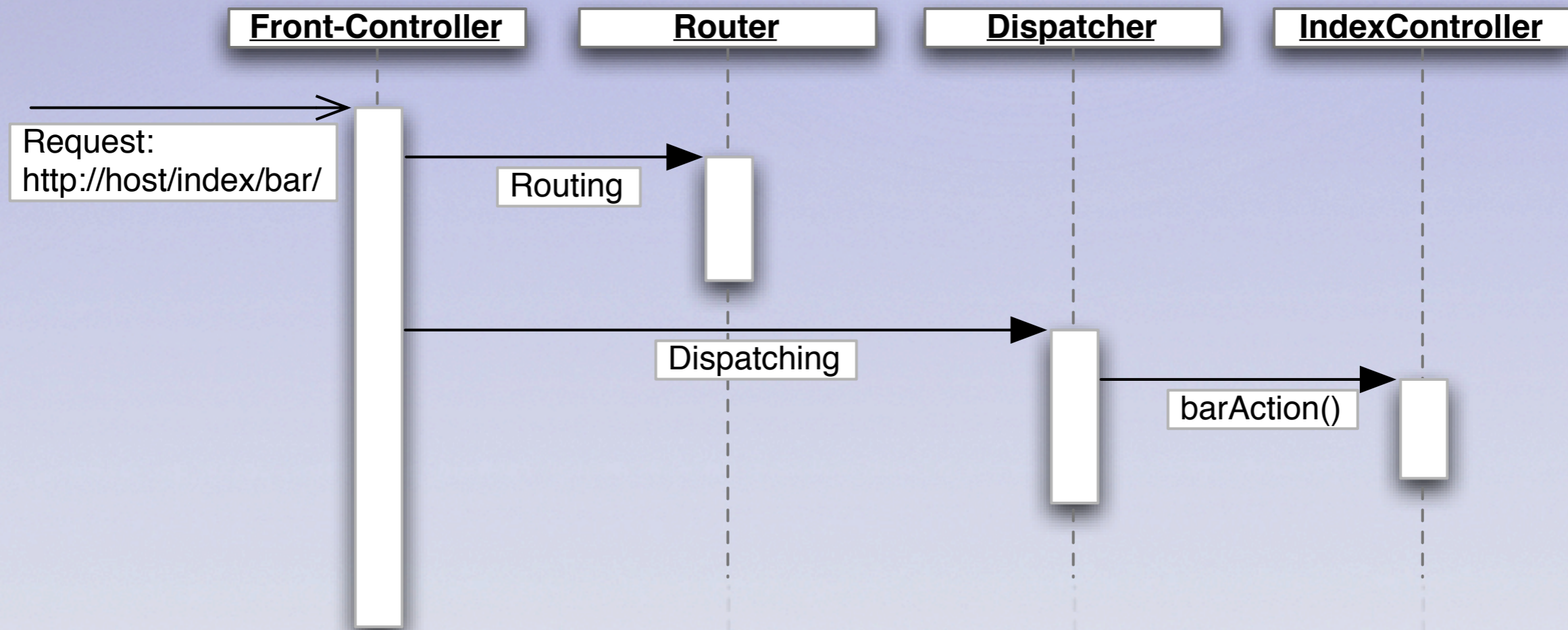
Requestlebenszyklus des ZF



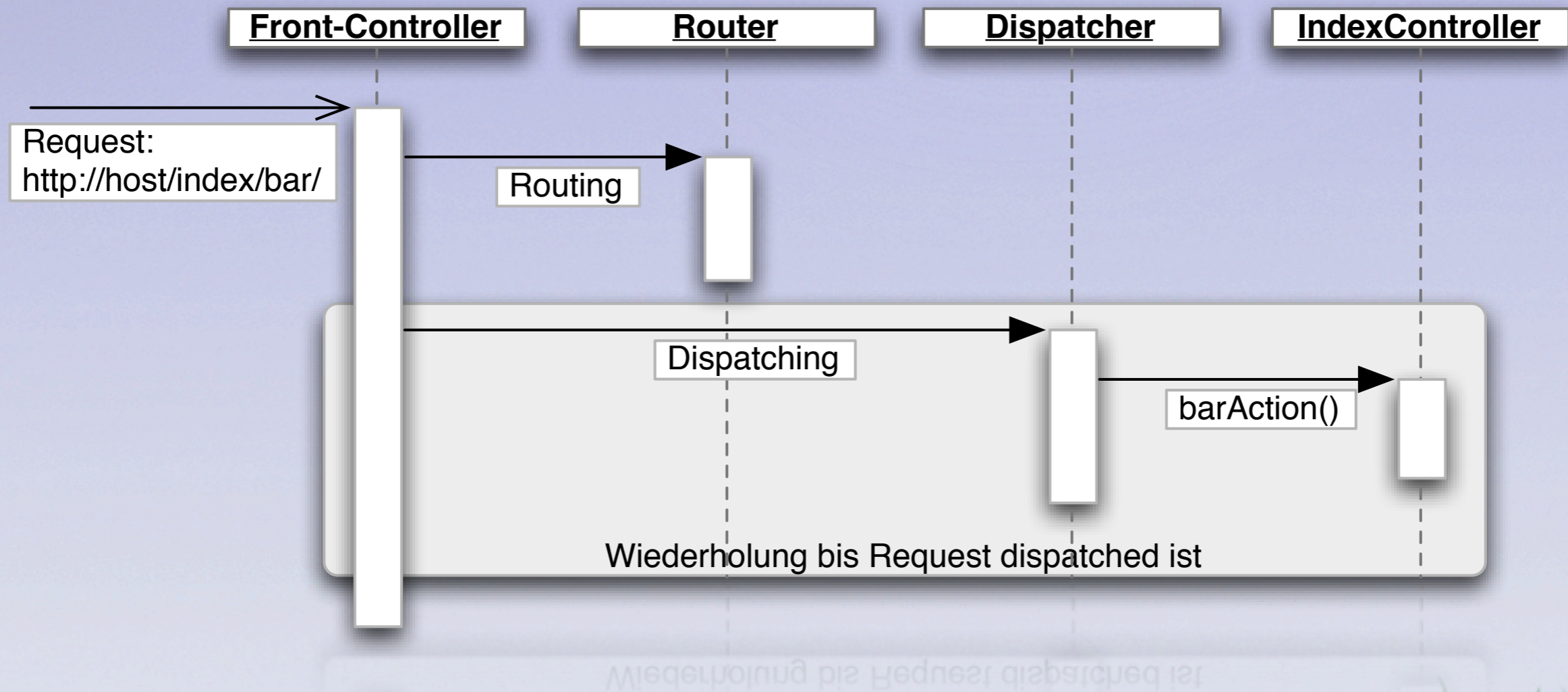
Requestlebenszyklus des ZF



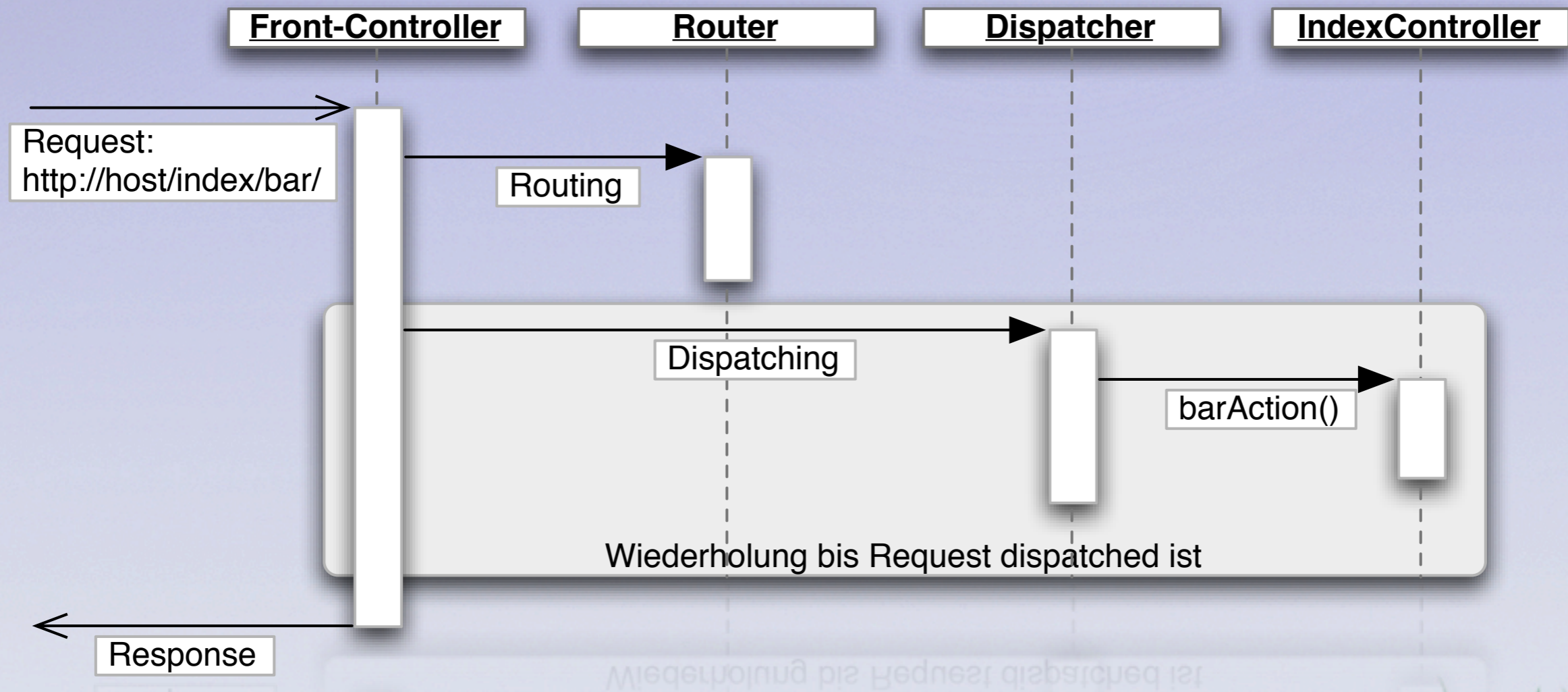
Requestlebenszyklus des ZF



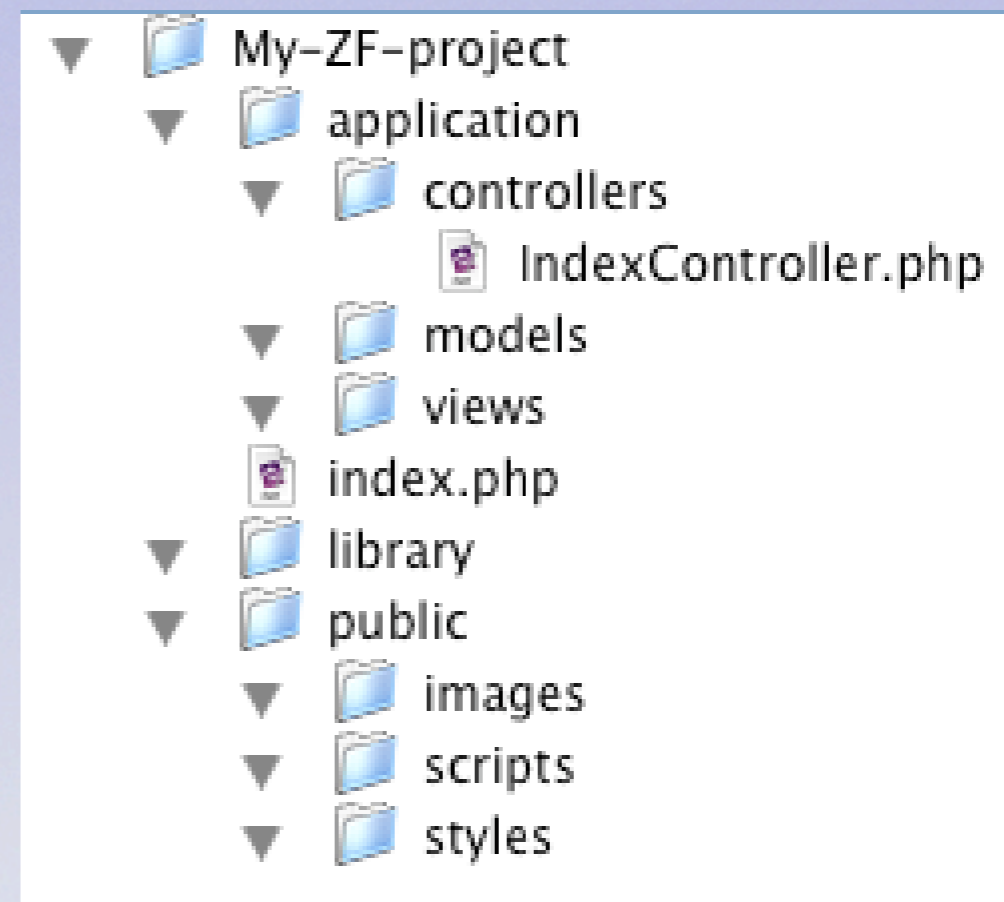
Requestlebenszyklus des ZF



Requestlebenszyklus des ZF

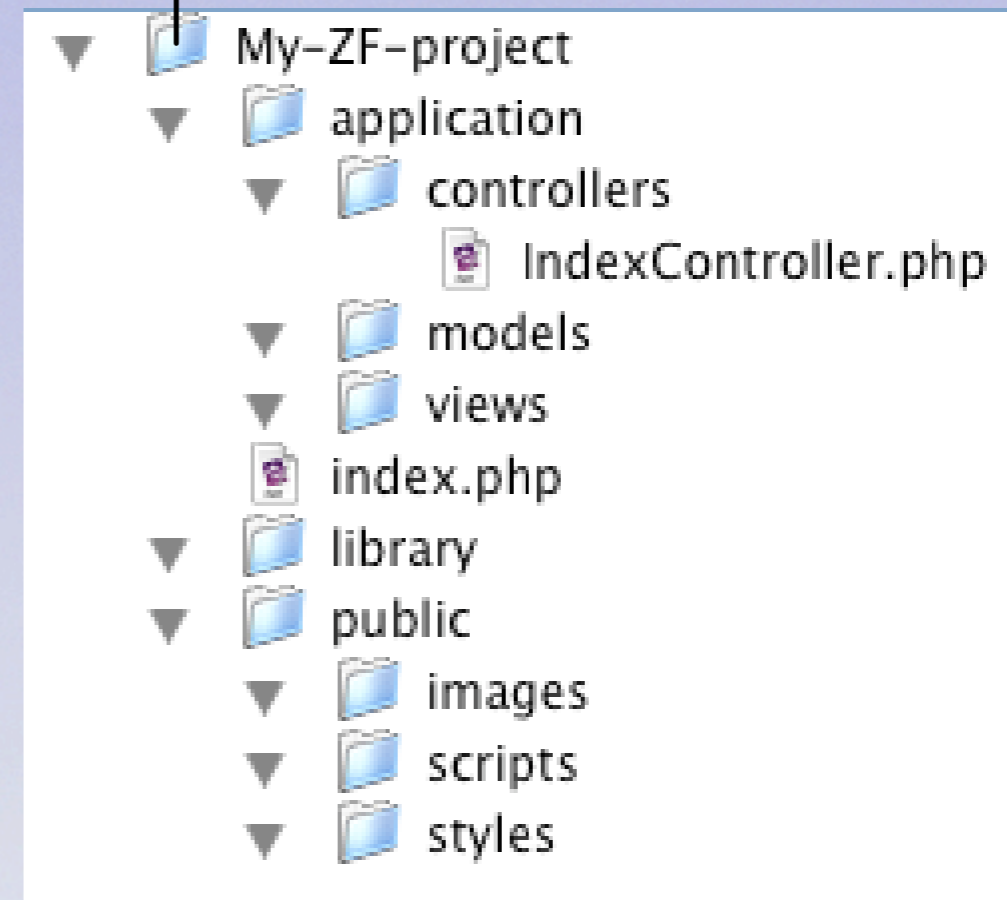


Typische Verzeichnisstruktur

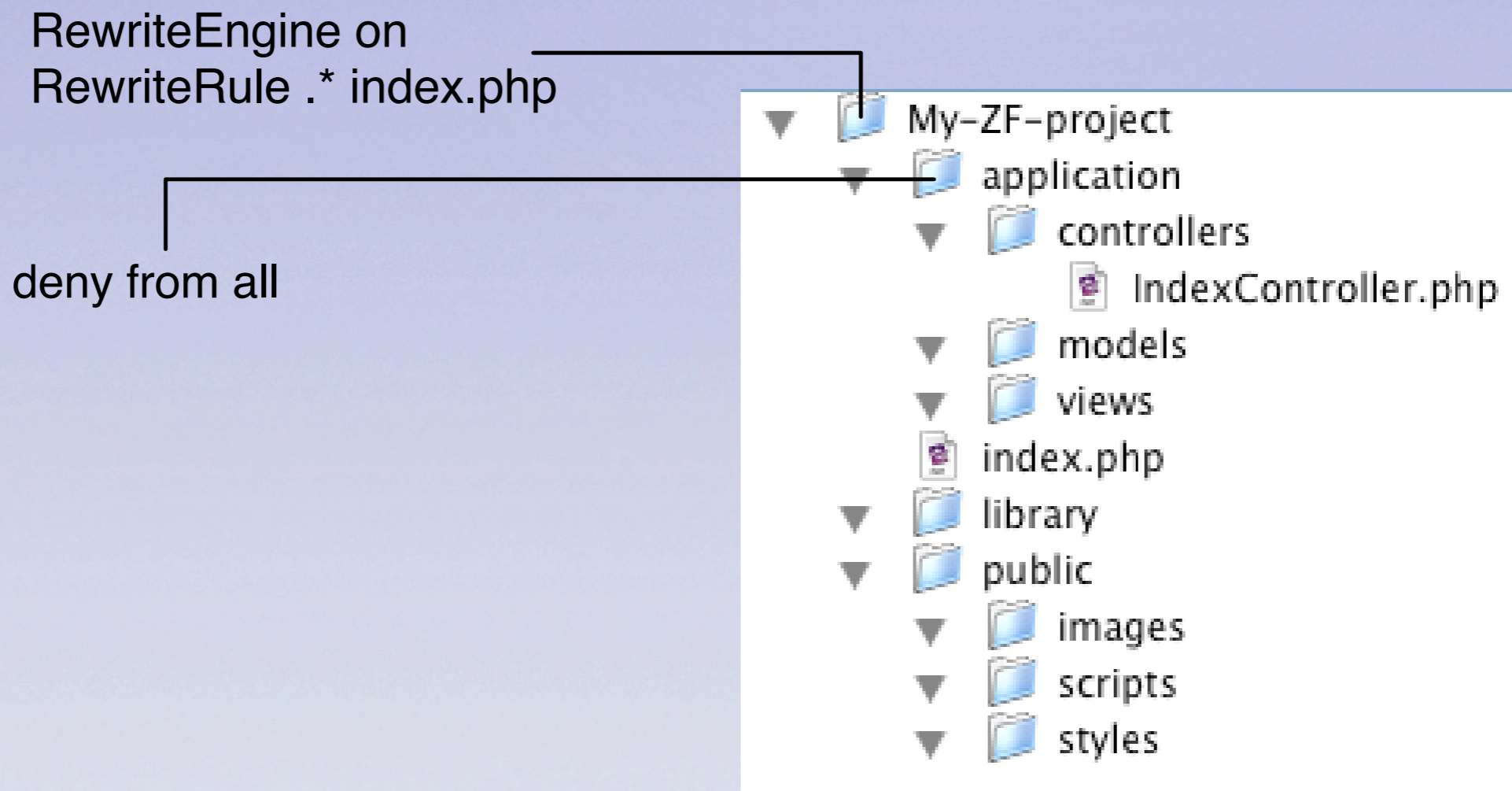


Typische Verzeichnisstruktur

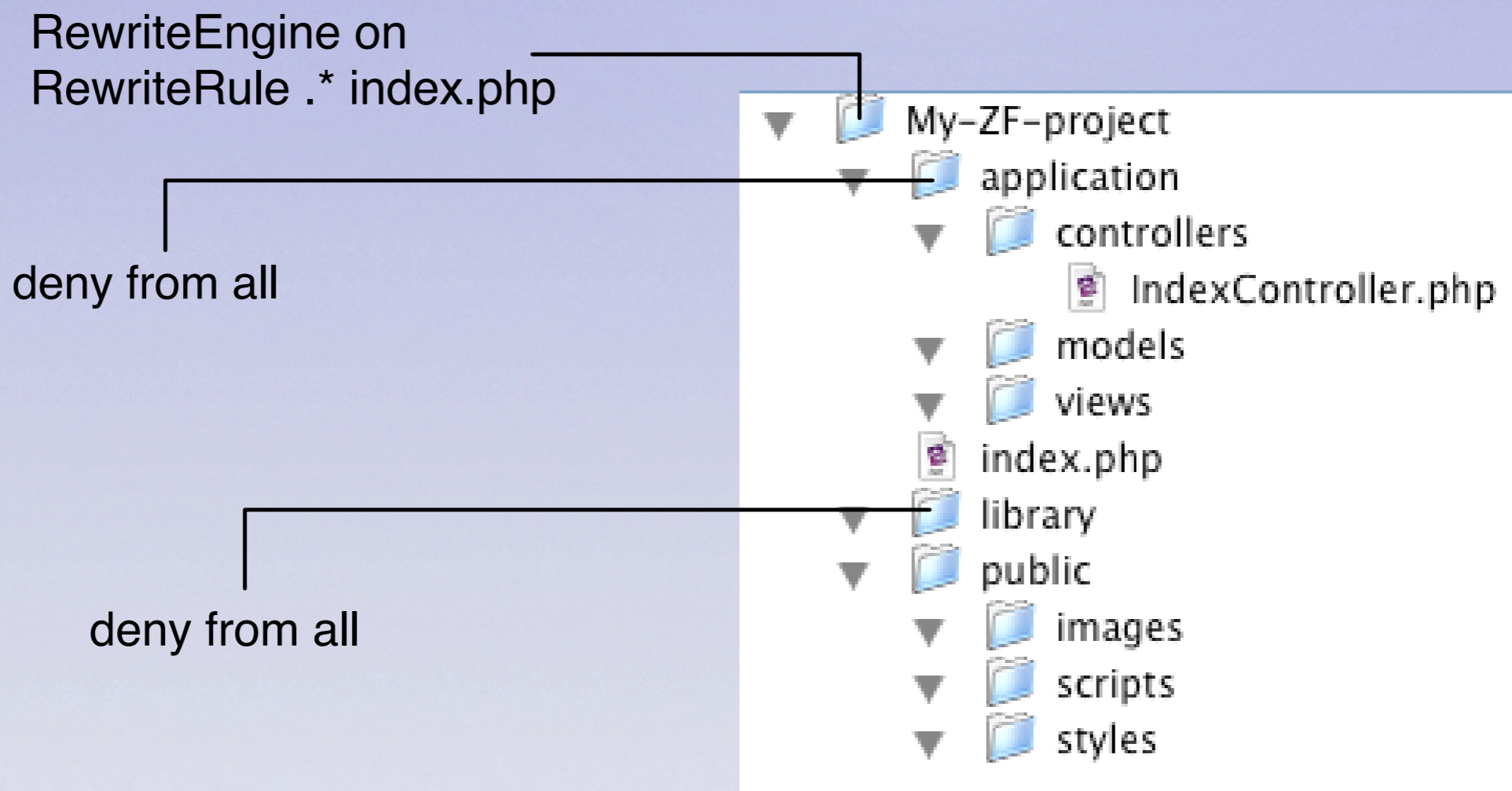
RewriteEngine on
RewriteRule .* index.php



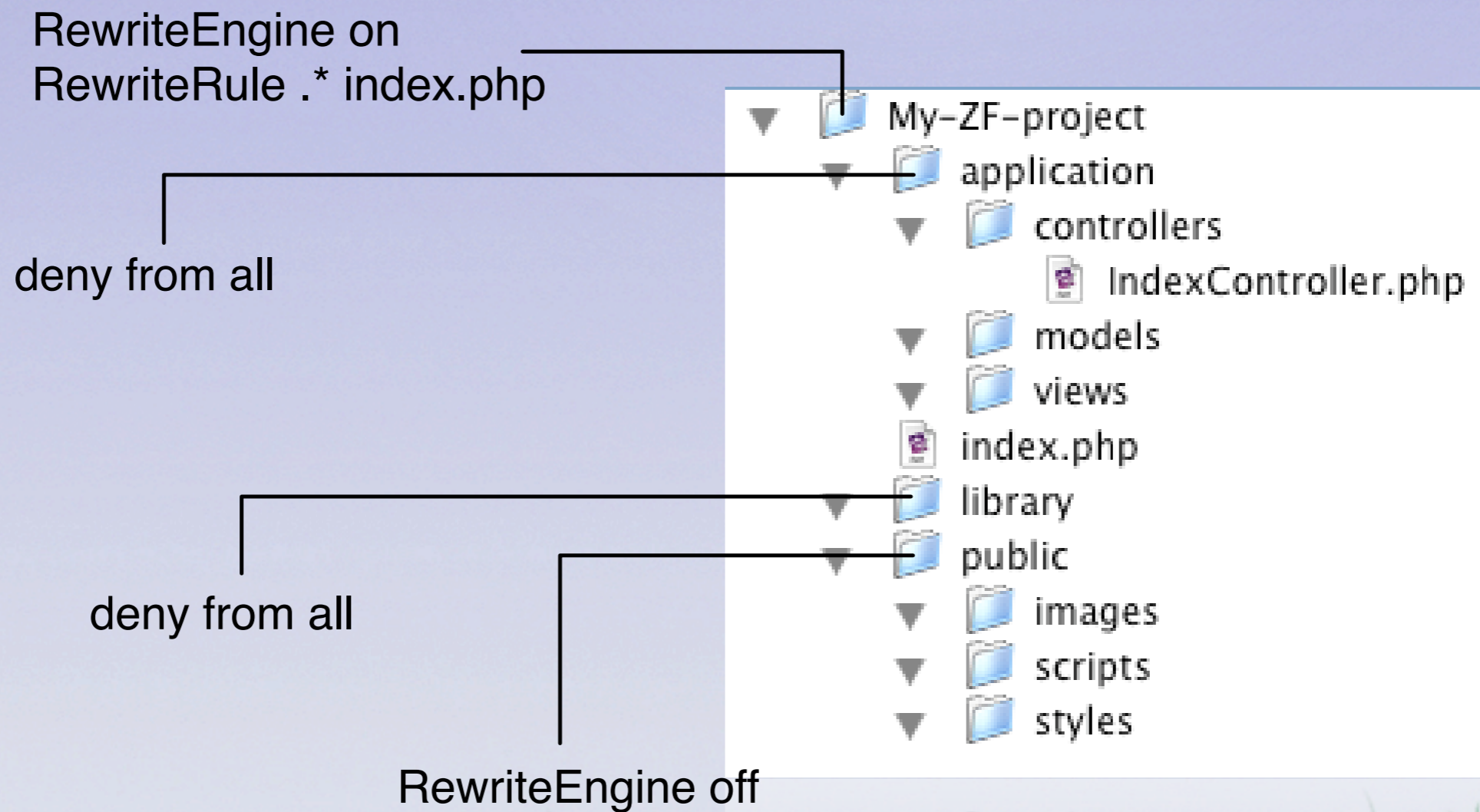
Typische Verzeichnisstruktur



Typische Verzeichnisstruktur



Typische Verzeichnisstruktur



Action-Controller für die Dialog/ Client-Logik

- Fachliche Verarbeitung eines Requests
 - Usecase-Aufrufe
 - Lesen anzuzeigender Daten aus DB etc.
 - Speichern eingegebener Daten
- Rendern der anzuzeigenden Seite mit Hilfe von Views

Action-Controller für die Dialog/Client-Logik

```
class IndexController extends Zend_Controller_Action
{
    protected $_view = null;

    public function init()
    {
        parent::init();
        $this->_view = $this->getInvokeArg('view');
    }

    public function indexAction() {
        $this->_view->title = 'Road-racing riders list';
        $this->_view->riders = array('Lance Armstrong', 'Erik Zabel', 'Tom Boonen');
        $this->_response->appendBody( $this->_view->render('riderslist.phtml') );
    }
}
```

Views für die Präsentation

- Separation von Präsentation und Logik bzw. Datenhaltung
- Ausschließlich Wissen über die Präsentation
- Eigene View-Datenhaltung möglich
- Rendering der anzuzeigenden Webseite (oder Teilen davon)

Views für die Präsentation

```
<html>
<head>
  <meta http-equiv="Content-type" content="text/html; charset=utf-8">
  <title><?php echo $this->escape($this->title); ?></title>
</head>

<body>
  <table border="0" cellspacing="5" cellpadding="5">
    <tr><th>Name</th></tr>
    <?php
      foreach ($this->riders as $rider) {
        echo "<tr><td>{$this->escape($rider)}</td></tr>";
      }
    ?>
  </table>
</body>
</html>
```

Der „Komponenten-Anteil“ des Zend Framework

Komponentenüberblick ZF

- Hohe Anzahl an Komponenten
- Abdeckung vieler Bereiche:
 - Auth, I18N, Webservices, Database, etc.
- Gute Dokumentation und viele Beispiele

Demonstration: ZF-Komponenten zur Abfrage eines RSS-Feed

Anforderungen:

Demonstration: ZF-Komponenten zur Abfrage eines RSS-Feed

Anforderungen:

- Abfrage der Feed-Informationen

Demonstration: ZF-Komponenten zur Abfrage eines RSS-Feed

Anforderungen:

- Abfrage der Feed-Informationen
- Fehlerbehandlung bei Nichterreichbarkeit

Demonstration: ZF-Komponenten zur Abfrage eines RSS-Feed

Anforderungen:

- Abfrage der Feed-Informationen
- Fehlerbehandlung bei Nichterreichbarkeit
- Caching der Ergebnisse

Demonstration: ZF-Komponenten zur Abfrage eines RSS-Feed

Anforderungen:

- Abfrage der Feed-Informationen
- Fehlerbehandlung bei Nichterreichbarkeit
- Caching der Ergebnisse

Eingesetzte Komponenten:

Demonstration: ZF-Komponenten zur Abfrage eines RSS-Feed

Anforderungen:

- Abfrage der Feed-Informationen
- Fehlerbehandlung bei Nichterreichbarkeit
- Caching der Ergebnisse

Eingesetzte Komponenten:

- Zend_Feed & Zend_Cache

Demonstration: Zend_Feed

```
try {  
    $feed = Zend_Feed::import('http://blog.lucke.info/feed');  
    $channel = array(  
        'title'      => $feed->title(),  
        'link'       => $feed->link(),  
        'description' => $feed->description(),  
        'items'     => array());  
  
    foreach ($feed as $item) {  
        $channel['items'][] = array(  
            'title'      => $item->title(),  
            'link'       => $item->link(),  
            'description' => $item->description());  
    }  
    print_r($channel);  
} catch (Zend_Feed_Exception $e) {  
    // Exception-Handling (Timeout, 404, etc.)  
}
```

Demonstration: Zend_Cache

- 4 Parameter zur Erzeugung einer Cache-Instanz:
 - Frontend, Backend, Frontend-Options, Backend-Options
- Frontends (*Was wird gecached?*):
 - Core, Output, Function, Class, File, Page
- Backends (*Wohin wird gecached?*):
 - File, Sqlite, Memcached, APC, ZendPlatform

Demonstration: Zend_Cache

```
$frontendOpt = array(
    'lifetime'           => 300,    // cache lifetime in seconds
    'automaticSerialization' => true);
$backendOpt = array( 'cacheDir' => './cache' );
$cache      = Zend_Cache::factory('Core', 'File', $frontendOpt, $backendOpt);

$cacheld    = 'feed_blog_lucke_info';
$feedInfo   = $cache->load($cacheld);

if (! $feedInfo) {
    // cache-miss
    $feedInfo = getFeedInfo('http://blog.lucke.info/feed');
    $cache->save($feedInfo, $cacheld);
} else {
    // cache-hit, Feed-Info bereits verfügbar
}
print_r($feedInfo);
```

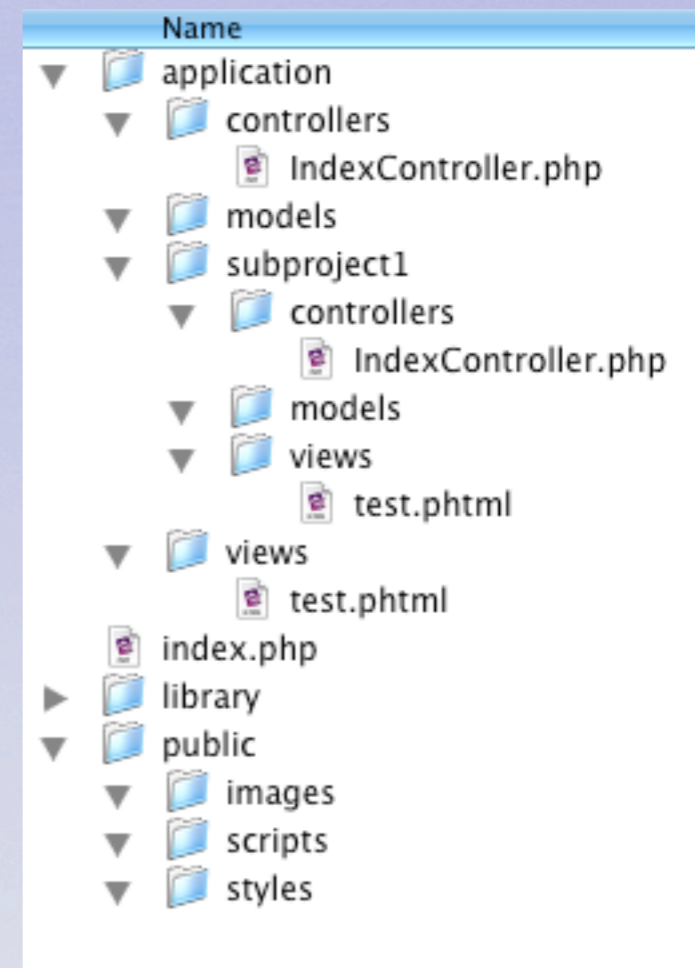
Agenda

Einführung und Überblick über das ZF
Framework oder Komponentenbibliothek?

> Fortgeschrittene Techniken
Bewertung

Projektstruktur für größere Projekte

- Projektstruktur erlaubt Subprojekte
- `Zend_Controller_Front`
`:: setControllerDirectory()`
- `Zend_Controller_Front`
`:: addControllerDirectory()`
- `Zend_View` `:: setScriptPath()`
- `Zend_View` `:: addScriptPath()`

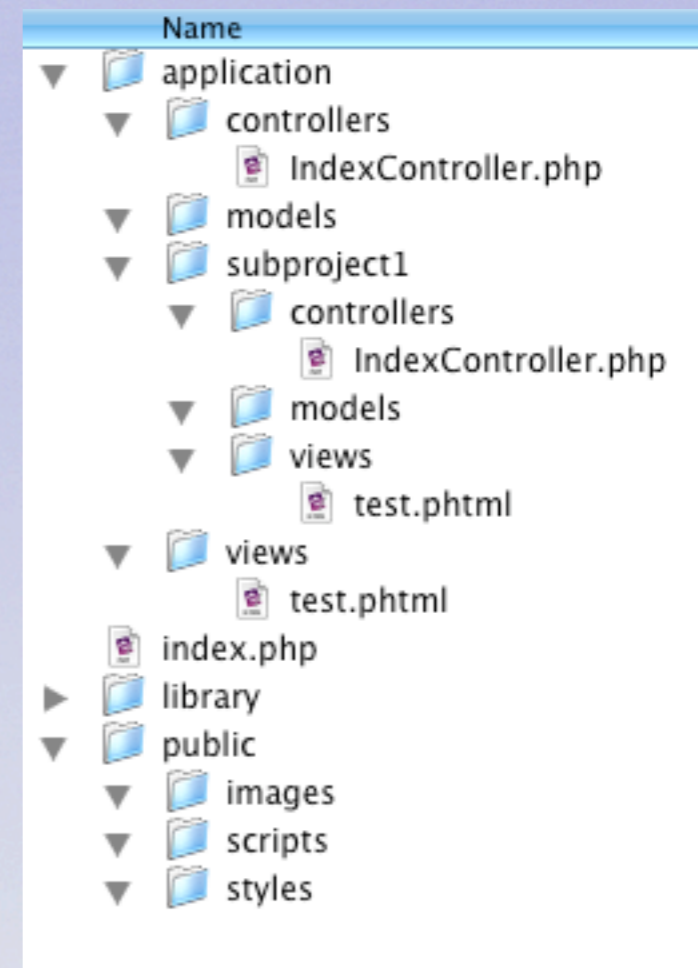


Projektstruktur für größere Projekte

[...]

```
$controller = Zend_Controller_Front::getInstance();  
$controller->setControllerDirectory(  
    'application/controllers');  
$controller->addControllerDirectory(  
    'application/subproject1/controllers', 'sub1');  
$controller->returnResponse(true);  
$controller->throwExceptions(true);  
  
$view = new Zend_View();  
$view->setScriptPath('application/views');  
$view->addScriptPath('application/subproject1/views');  
$controller->setParam('view', $view);
```

[...]

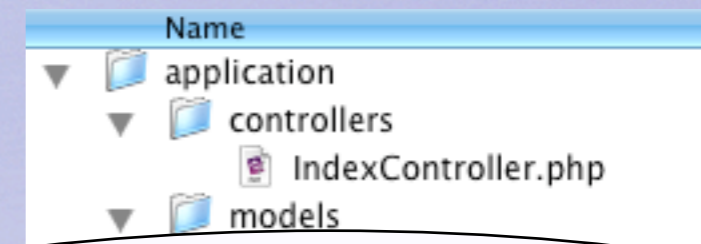


Projektstruktur für größere Projekte

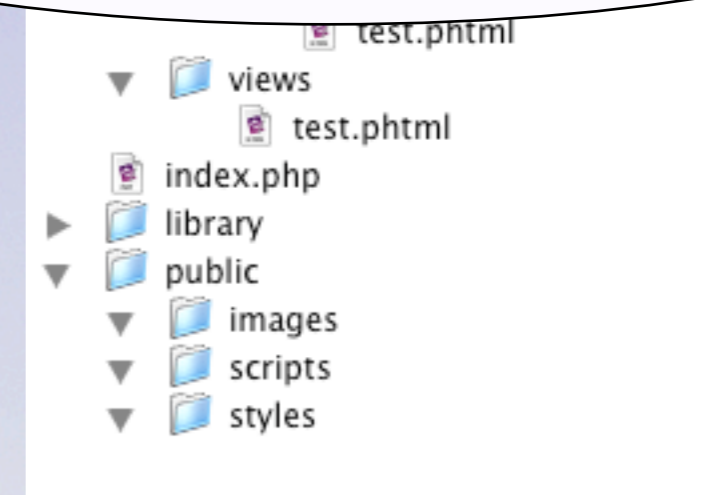
[...]

```
$controller = Zend_Controller_Front::getInstance();  
$controller->setControllerDirectory(  
    'application/controllers');  
$controller->addControllerDirectory(  
    'application/subproject1/controllers', 'sub1');  
$controller->returnResponse(true);  
$controller->throwExceptions(true);  
  
$view = new Zend_View();  
$view->setScriptPath('application/views');  
$view->addScriptPath('application/subproject1/views');  
$controller->setParam('view', $view);
```

[...]



<http://myhost/sub1/myctrl/myaction>



Template-basierte Views

- Zend_View_Interface als Schnittstelle
- Template-Engine Ihrer Wahl anstelle von PHP/HTML Views
- Rahmentemplate mit Platzhaltern anstelle von Includes innerhalb von Views

Mehrere Actions zur Request- abarbeitung

- Aufteilung der fachlichen Abarbeitung eines Request auf mehrere Controller
- Dispatch-Zyklus erlaubt Einsatz mehrerer Controller/Actions zur Requestabarbeitung
 - „Forwarding“: `$this->_forward()`
- Problem: Controller haben dadurch Kenntnis von einander

Mehrere Actions zur Request- abarbeitung :: Lösungsansatz

- Abstrakte Action-Basisklasse zur Verwaltung eines Navigation-Stack
 - `init()` zur Initialisierung des Stack
 - `myForward()` statt `_forward()`
 - `postDispatch()` für „Forwarding“ mit mod. Navigation-Stack als Kontextparameter

Agenda

Einführung und Überblick über das ZF
Framework oder Komponentenbibliothek?
Fortgeschrittene Techniken

> **Bewertung**

Bewertung des Zend Framework

Pro:

Contra:

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung

Contra:

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung
- Viele Komponenten für Querschnitts- und Spezialaufgaben

Contra:

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung
- Viele Komponenten für Querschnitts- und Spezialaufgaben
- Hohe Qualität

Contra:

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung
- Viele Komponenten für Querschnitts- und Spezialaufgaben
- Hohe Qualität
- Gute Dokumentation

Contra:

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung
- Viele Komponenten für Querschnitts- und Spezialaufgaben
- Hohe Qualität
- Gute Dokumentation

Contra:

- Einsatz von Zend_View in größeren Projekten / kompl. Views zu überdenken

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung
- Viele Komponenten für Querschnitts- und Spezialaufgaben
- Hohe Qualität
- Gute Dokumentation

Contra:

- Einsatz von Zend_View in größeren Projekten / kompl. Views zu überdenken
- Für größere Projekte evtl. noch „Framework-Arbeit“ notwendig

Bewertung des Zend Framework

Pro:

- Durchdachte MVC-Umsetzung
- Viele Komponenten für Querschnitts- und Spezialaufgaben
- Hohe Qualität
- Gute Dokumentation

Contra:

- Einsatz von Zend_View in größeren Projekten / kompl. Views zu überdenken
- Für größere Projekte evtl. noch „Framework-Arbeit“ notwendig
- Framework für stark dialogbasierte Systeme evtl. nicht der richtige Ansatz

Vielen Dank für Ihre Aufmerksamkeit!

Haben Sie Fragen?

<http://lucke.info>