

ColaPHP In Action

超群.com

@fuchaoqun

<http://www.fuchaoqun.com>

Why Cola?

- 你和我一样希望规范的MVC开发
- 你和我一样希望高性能的框架
- 你和我一样不想重新学习一门“框架语言”
- 你和我一样不希望改变已有的PHP开发方式
- 大多数框架代码是写给机器看的，Cola是写给程序员看的

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

Performance

- 测试代码：

```
<?php
```

```
echo 'Index@', time();
```

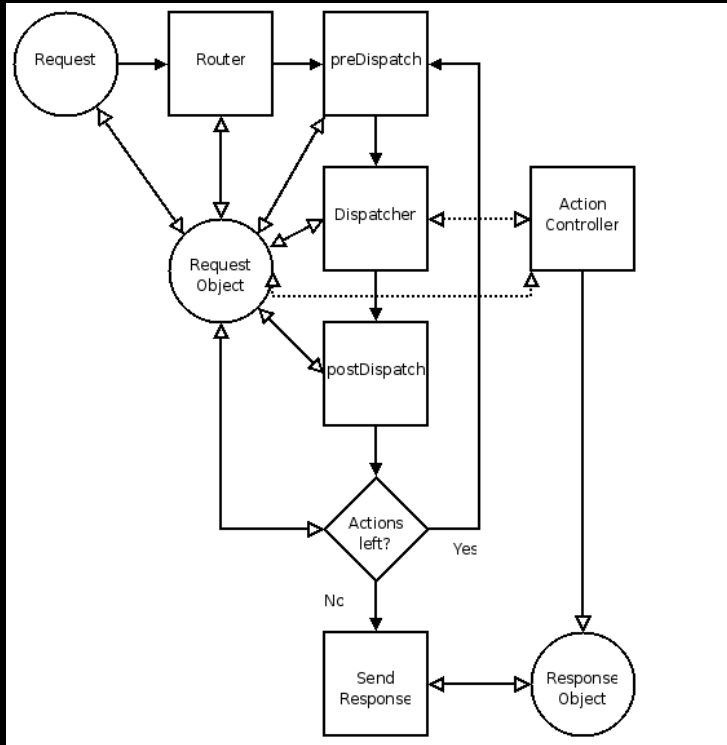
- 测试环境：PHP5.2.10 + eAccelerator `ab -c200 -n10000`

原生PHP：794 RPS

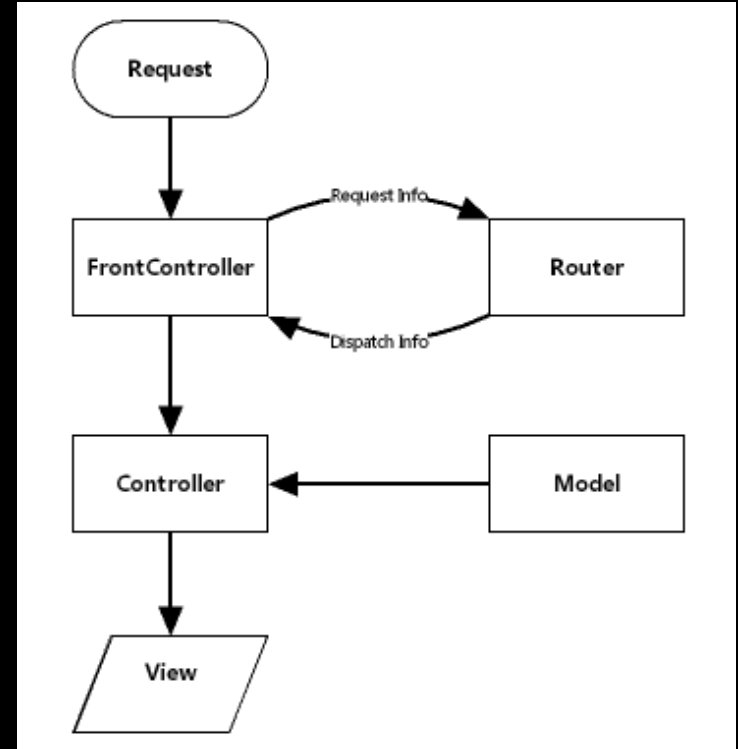
ColaPHP：392 RPS ~50%

- 框架消耗1~1.5ms

Workflow



Zend Framework



ColaPHP

FrontController

- 全部在Cola/Cola.php文件里面
- 依赖\$_SERVER['PATH_INFO']进行调度
- 有些Web Server (如nginx) , 没有PATH_INFO变量 , 还可以\$cola->setPahtInfo(\$pi)来指定
- Router、Dispatch都可以通过setRouter、setDispatchInfo自己定义
- 其他一些功能 , 如Auto Load、全局配置、寄存器

Router

- Cola/Router.php
- 支持默认调度和自定义规则调度
- 默认调度：/blog/view/id/9527

调度到BlogController中的viewAction，可通过param('id')获得9527

- 自定义正则调度：/blog/9527

规则：/^blog\(\d+\)\$/ => BlogController, viewAction

直接传入9527到viewAction函数中

Controller

- `$ctl->get()`获得`$_GET`参数，`$ctl->post()`获得`$_POST`参数，亦可指定key获得对应变量的值
- `$ctl->form($keys)`获得整个表单数据，`$keys`可以通过`$this->_keys`来定义
- `$ctl->model('foo')`可获得FooModel
- 只有调用`$ctl->display()`才会显示模版内容，这就意味着你也可以用Smarty（假如你喜欢）
- 魔术变量：

`$ctl->model`：获得与Controller同名Model

`$ctl->com->pager`：返回Com_Pager分页组件

`$ctl->request`：请求对象，`$ctl->response`：响应对象

Model

- 便宜的CRUD操作
- `$mdl->sql($sql)`直接返回SQL结果

如果是SELECT语句则返回结果集，INSERT语句则返回Last Insert Id，UPDATE、DELETE则返回影响行数

- `$mdl->cache()`返回Cola_Cache对象
- `$mdl->cached($func, ***)`魔术方法，可对Model中任何一个函数加缓存，类似装饰模式
- `$mdl->load($id)`按主键返回单条数据
- `$mdl->find()`按条件获得数据集

View

- 原生的PHP模版
- 支持Slot模版片段
- `$view->escape()` 支持HTML转义
- `$view->truncate()`支持中文截字

Cola_Request

- Cola/Request.php
- `$rqt->get()`获得`$_GET`参数, `$rqt->post()`获得`$_POST`
- 获取请求头信息
- `$rqt->clientIp()`获得访问者IP

Cola_Response

- Com/Response.php
- 输出响应头信息，如Cookie、Status Code之类
- \$rps->etag()：输出ETAG信息
- \$rps->lastModified()：输出最后修改信息
- \$rps->expire()：输出过去时间

Components

- Cola/Com目录下
- 除基本的MVC外，其他功能全部组件化，Mini Core

- 目前提供的组件有：

Com_Benchmark：性能测试组件

Com_Cache：缓存组件

Com_Captcha：验证码组件

Com_Db：MySQL组件

Com_Encrypt：加密、解密组件

Com_Http：HTTP组件

Com_Log：日志组件

Com_Mongo：MongoDb组件

Com_Pager：分页组件

Com_Upload：上传组件

Com_Validate：数据校验组件

Com_Yaml：YAML组件

Com_Db

- 工厂模式，支持mysql、mysqli、pdo_mysql、MySQL主从模式
- 字段自动转义，防止SQL注入
- 便宜的CRUD操作
- 支持事务
- 支持Debug模式，详尽的记录SQL

Com_Cache

- 工厂模式，支持Memcache、APC、eAccelerator、Xcache、File、Dbal、Null
- 基本用法\$cache->set()、\$cache->get()、\$cache->delete()
- 魔术方法：

\$cache->foo = 'bar' 相当于 \$cache->set('foo' , 'bar')

\$cache->foo = null 相当于 \$cache->delete('foo')

Com_Pager

- 一开始就想做，但最后才集成的一个组件
- 自带默认显示效果 
- 兼容易用性、扩展性和可定制性

```
protected $_config = array(  
    'prevNums'      => 2,  
    'nextNums'      => 7,  
    'showSinglePage' => false,  
    'prefix'        => '<style type="text/css">.pages{float:right;font-size:12px;}',  
    'first'          => '<a href="%link%">%page%...</a>',  
    'last'           => '<a href="%link%">...%page%</a>',  
    'prev'           => '<a class="pn" href="%link%">&lt;&lt;</a>',  
    'next'           => '<a class="pn" href="%link%">&gt;&gt;</a>',  
    'current'        => '<b>%page%</b>',  
    'page'           => '<a href="%link%">%page%</a>',  
    'suffix'         => '</div>',  
);
```

```
$p = array(  
    '%link%'      => $this->url,  
    '%curPage%'   => $this->curPage,  
    '%pageSize%'  => $this->pageSize,  
    '%totalItems%' => $this->totalItems,  
    '%totalPages%' => $this->totalPages,  
    '%startPage%' => $this->startPage,  
    '%endPage%'   => $this->endPage,  
    '%page%'      => $page  
);
```

Com_HTTP

- 基于PHP自带的file_get_contents和stream_context , 不依赖第三方扩展 (如curl之类)
- 实现GET、POST请求
- 可设置HTTP请求头信息 , 比如带Cookie , Host设置等
- 可设置访问超时
- 可获取HTTP返回状态码等

Com_Upload

- 支持多文件上传，自动格式化上传文件信息
- 基本的文件上传过滤，如大小、类型等
- 可自动保存文件
- 可通过\$upload->files()获得上传文件列表，然后自定义处理

```
protected $_config = array(  
    'savePath' => '/tmp',  
    'maxSize' => 0,  
    'maxWidth' => 0,  
    'maxHeight' => 0,  
    'allowedExts' => '*',  
    'allowedTypes' => '*',  
    'override' => false,  
);
```

```
$this->_files[] = array(  
    'field' => $field,  
    'name' => $file['name'][$i],  
    'type' => $file['type'][$i],  
    'tmp_name' => $file['tmp_name'][$i],  
    'error' => $file['error'][$i],  
    'size' => $file['size'][$i],  
    'ext' => $this->getExt($file['name'][$i], true)  
);
```

Com_Validate

- 数据校验处理
- 支持非空、Email、Url、IP、时间、数字、范围等常见数据校验
- 支持批量校验
- 可自定义错误提示信息

```
protected static $_message = array(  
    'email'      => 'invalid_email',  
    'required'  => 'empty',  
    'max'       => 'above_max',  
    'min'       => 'below_min',  
    'range'     => 'not_in_rang',  
    'ip'        => 'invalid_ip',  
    'number'    => 'not_all_numbers',  
    'int'       => 'not_int',  
    'digit'     => 'not_digit',  
    'string'    => 'not_string'  
);
```

Plans

- ~~ColaPHP目前最新的版本是0.8beta，已经非常稳定，可以在生产环境使用，~1.0GA~~
- 1.0GA将在今年内已经发布，基于PHP5.2系列，兼容PHP5.3
- 未来的某个时间启动ColaPHP2.0开发，全面拥抱PHP5.3各种特性

Thank You & QA