

Drupal theme机制及主题开发

-Drupal培训Lesson 6

Rapid Technologies

www.chinaldap.com

课程安排

- Drupal 的theme介绍
- Theme的工作原理
- Theme的文件组成及创建新theme
- Theme的覆写机制
- 其他 - 子主题 (sub theme), 我们该怎么制作主题?

本课程适合对drupal能基本使用熟悉html,css等

Drupal 的theme介绍



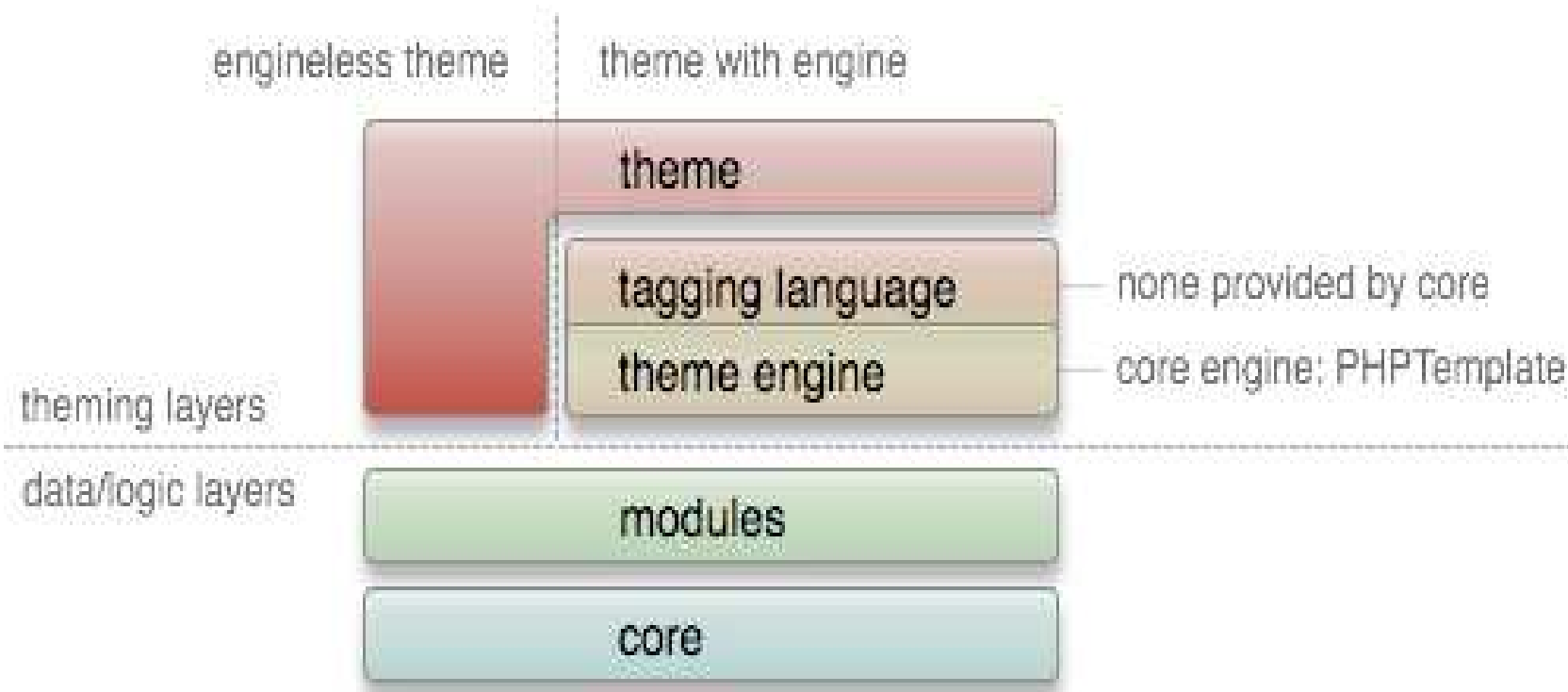
Drupal theme的介绍

- Drupal的theme,定义了整个网站的外观和样子
- Drupal安装完成后，没有前/后台之分
- Drupal的第三方提供了成百上千的免费的 themes,如果要求不高，[下载theme](#)就可以用
- Theme从物理意义上来说，是一组由各种文件组成的集合，这些文件包括javascript, css, tpl, .info 文件等。

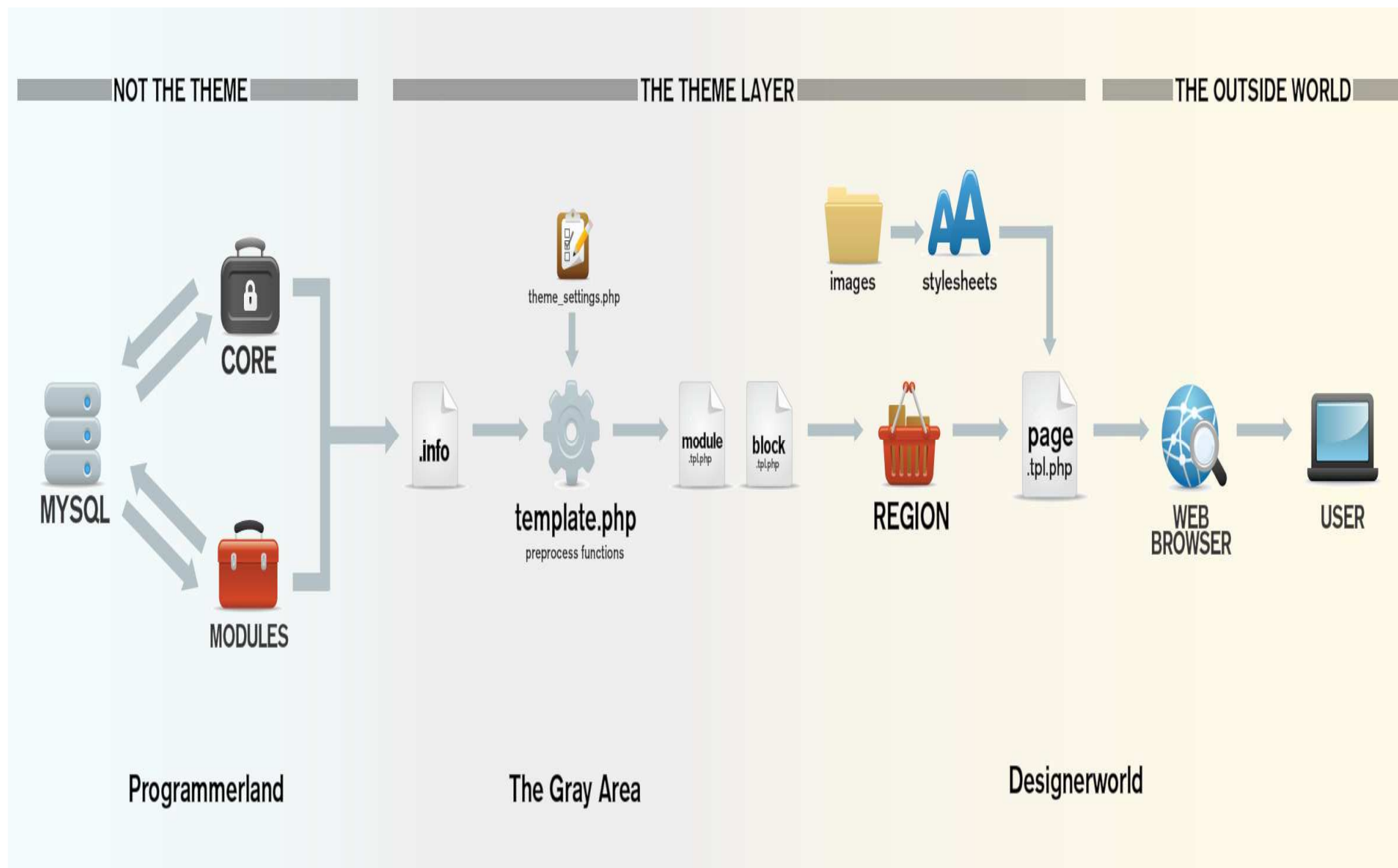
Theme的工作原理



Theme输出图解



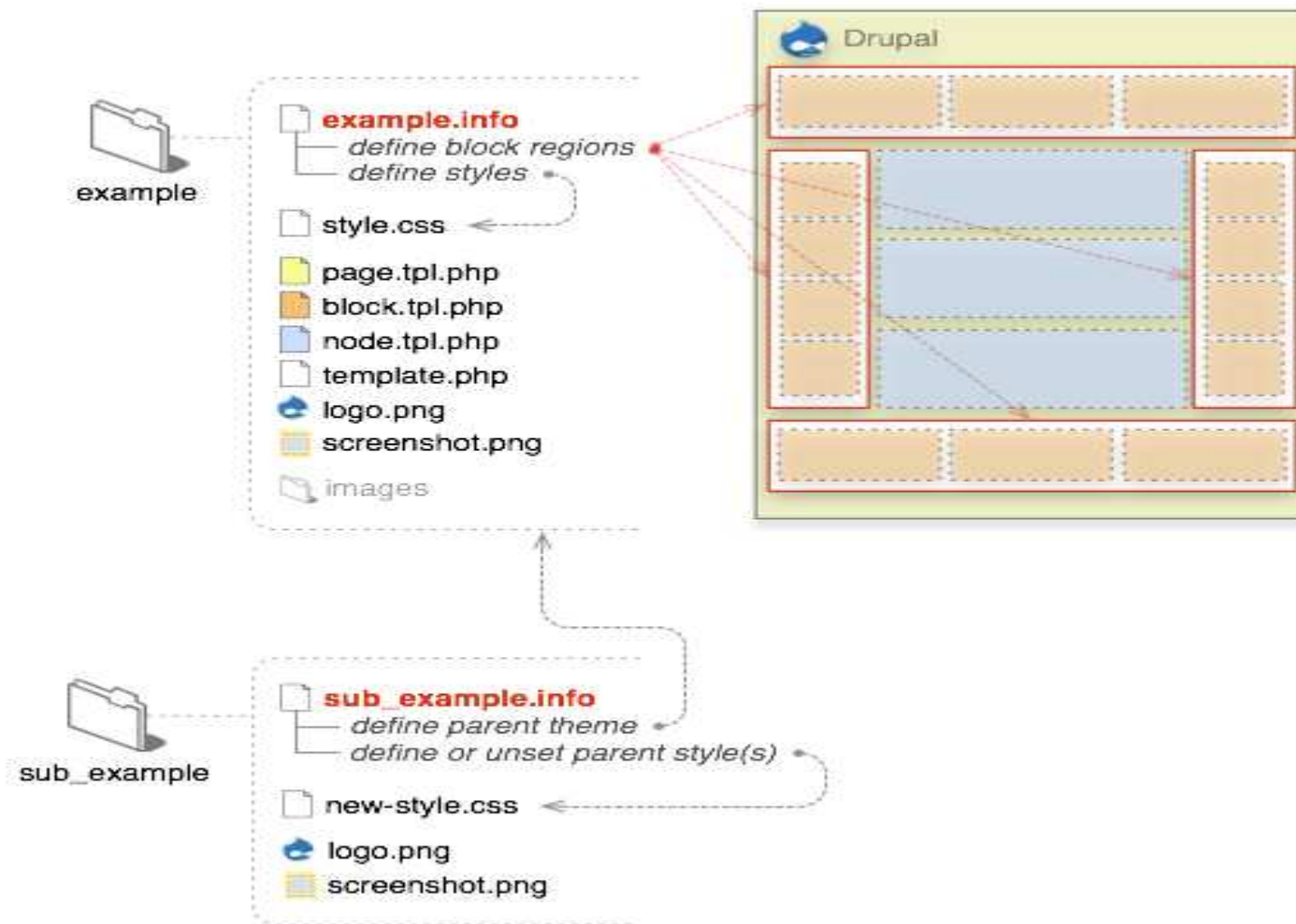
又一个theme输出图解



Theme的文件组成及创建新theme



Theme 的文件组成 – 概括



Theme 的文件组成 – 详细介绍

- 必须的文件 **.info** 。 比如 `garland.info`，主要用来存放这个主题的信息，比如版本号，主题名字等
- 其他可选的文件：（可选，表示如果没有，则使用默认的或者不显示）
 - **template files (.tpl.php)**，模板文件，主要是(x)html语言和php变量来组成，每个模板文件来处理不同的数据片段，比如
 - `page.tpl.php` - 页面的布局文件,上面定义的页面区域(regions)显示,以及其它站点信息会在此页面显示
 - `node.tpl.php` - 节点的模板文件,drupal 把所有的文章内容都视为节点,用于对一个节点的页面进行布局。
 - `block.tpl.php` - 区块的模板文件，定义区块的显示样子
 - **template.php** ，用来处理业务逻辑或者数据处理，比如可以来添加一个变量用在tpl.php里。

Theme 的文件组成 - .info详解

```
2
3 name = Clean
4 description = Lean, mean base theme with a focus on minimal implementation.
5 core = 6.x
6 engine = phptemplate
7 screenshot = images/screenshot.png
8 stylesheets[all][] = css/reset.css
9 stylesheets[all][] = css/html5.css
10 stylesheets[all][] = css/layout.css
11 stylesheets[all][] = css/navigation.css
12 stylesheets[all][] = css/typography.css
13 stylesheets[all][] = css/color.css
14 stylesheets[all][] = css/clean.css
15
16 scripts[] = js/clean.js
17 regions[left] = Left sidebar
18 regions[right] = Right sidebar
19 regions[content] = Content
20 regions[header] = Header
21 regions[footer] = Footer
22
23 ; Information added by drupal.org packaging script on 2010-02-02
24 version = "6.x-2.0-beta2"
25 core = "6.x"
26 project = "clean"
27 timestamp = "1265148615"
28
```

必填，主题的名字

主题的描述，
后台管理时候看到的信息

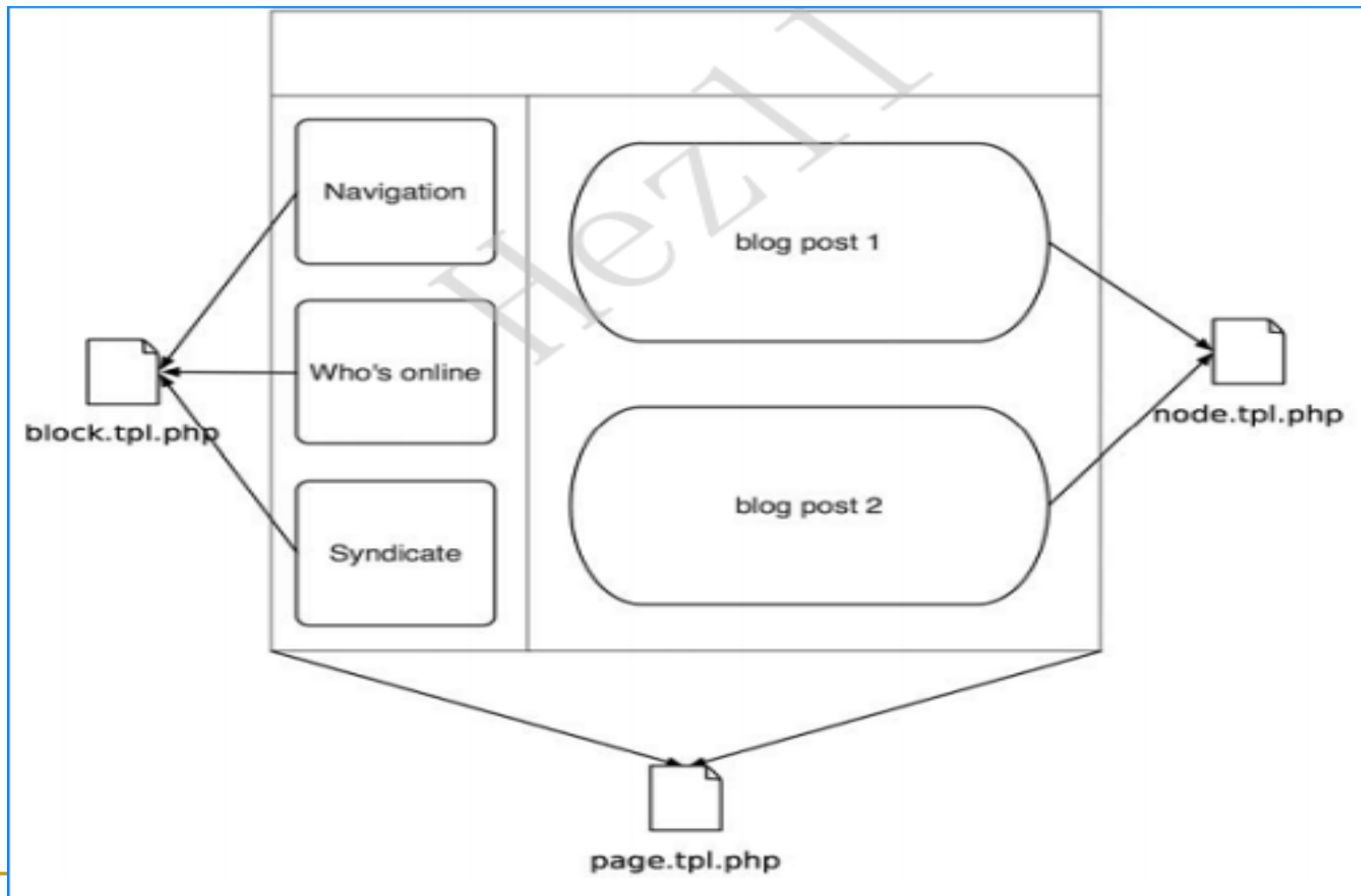
Drupal的版本号

截屏图片，用户后台直观显示主题的样子

CSS样式表，加载到所有页面

网页的整个区域(region)划分，默认下有
left,right, content, header ,footer,如果需要加
新的，则必须全部写上

Theme 的文件组成 – 模板文件



创建一个新的theme – doria 主题

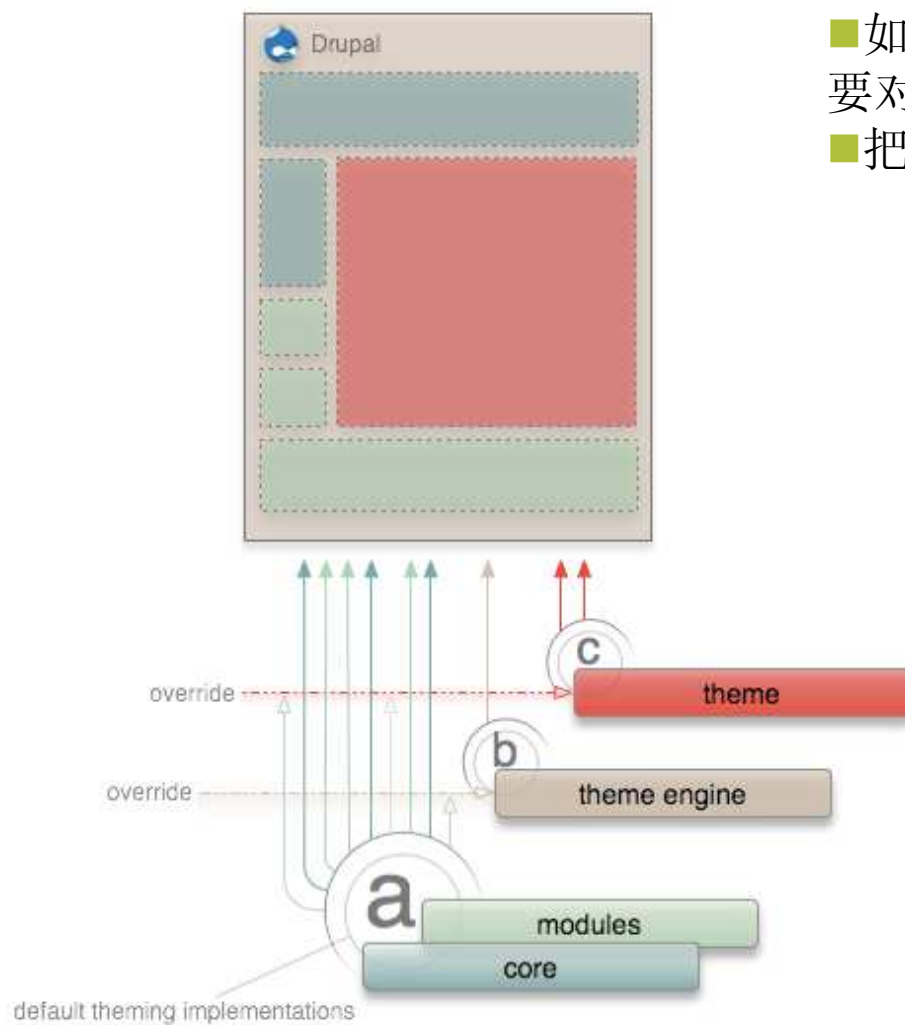
- 第一步： 创建新文件夹, `site/all/themes/doria`
- 第二步： 进入到`doria`目录，在这里创建新文件 `doria.info`,并添加必要内容（执行完这步后，到`drupal`后台，主题列表里应该显示本主题）
- 第三步: 在`doria`目录，创建一个新的文件 `page.tpl.php`（你可以先把它看作为一个`.html`文件),在这个文件里，定义了整个的页面布局结构等。

这样，一个简单的**theme**就完成了，到后台设置并使用此主题，那就看到了你要的效果。

Theme的覆写机制



什么是覆写？



- 如果默认的显示不符合我们的要求，那么就要对默认的输出进行覆写。比如用户登录框
- 把修改过的覆写文件放到我们自己的主题下面

实现主题钩子(theme hook)的函数VS模板

■注册一个主题钩子， `hook_theme`
让我们以`user`模块里的一个例子来看看

```
function user_theme() {  
  return array(  
    'user_picture' => array(  
      'arguments' => array('account' => NULL),  
      'template' => 'user-picture',  
    ),  
    'user_list' => array(  
      'arguments' => array('users' => array()),  
    ),  
  );  
}
```

使用了模板的方式，那肯定需要一个文件 `user-picture.tpl.php` 来对应输出。

覆写？在你的主题下面增加文件 `user-picture.tpl.php`，就可以覆写了。

使用了函数的方式，那肯定需要一个函数叫 `theme_user_list($users,$title)` 对应输出。

覆写？在你的主题文件里，`template.php`里，实现函数 `<你主题名字>_user_list()`

主题模板建议

- 模板建议是一套基于已经存在的`tpl.php`文件的备用的模板文件，只要符合一定的命名规则，那么一定的模板文件就会被调用而覆写原来的模板文件输出。
- 比如：`node.tpl.php` 是默认的输出文件，而`node-story.tpl.php`则是对`story`这个`node`的输出文件。当不存在`node-story.tpl.php`的时候，`story`的输出是根据`node.tpl.php`来的，但是一旦有了这个`node-story.tpl.php`文件，则使用这个文件输出。
- 参考更多 <http://drupal.org/node/1089642>

常用模板建议 - page

- **page-[front|internal/path].tpl.php**
 - 基础模板: *page.tpl.php* , 首页的模板 *page-front.tpl.php*
其他页面模板则是根据内部路径来写的。
 - .
 - 一个模板的建议是根据内部路径的顺序来决定的 , 一个建议文件是根据路径的元素来使用,比如,
“<http://www.example.com/node/1/edit>” 会对应到的如下的建议模板:
 - **page-node-edit.tpl.php** (如果没出现这个, 则使用page-node-1.tpl.php)
 - **page-node-1.tpl.php** (如果没出现这个, 则使用page-node.tpl.php)
 - **page-node.tpl.php** (如果没出现这个, 则使用page.tpl.php)
 - **page.tpl.php** (如果没出现这个, 则使用drupal默认的page)

常用模板建议 - Block

- **block-[region|[module|-delta]].tpl.php**
 - 基础模板: *block.tpl.php* 模板建议根据以下的顺序来制作:
 - *block-module-delta.tpl.php* 比如 *block-user-1.tpl.php*
 - *block-module.tpl.php* 比如: *block-user.tpl.php*
 - *block-region.tpl.php* 比如: *block-left.tpl.php*
 - “module” 是创建的module的名字, “delta”是那个模块产生的block的id, (见hook_block). 比如, “block-user-1.tpl.php”, 是默认的用户导航区块, 这个是由‘user’模块产生, ID 为1 .

常用模板建议 - Node

■ node-[type].tpl.php

- ❑ 基础模板: *node.tpl.php* , 可以为每个 *content type* 来指定一个模板,
 - Drupal 6: node-mytype.tpl.php
- ❑ 请参考 [node api](#) 来查看 可以使用的变量。



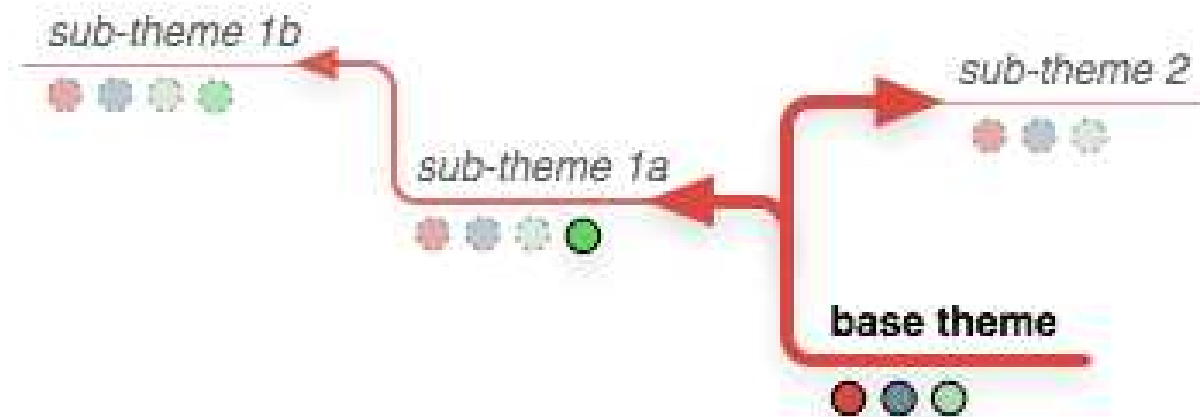
注意: 如果想让node_mytype.tpl.php有效, node.tpl.php必须同时存在

其他－子主题，怎么做个主题？



子主题 (sub theme)

- Sub theme也和其他普通的theme一样，只是它继承了父theme的所有资源，比如css, javascript等



- sub theme可以放在任何一个地方（在drupal 5.x以前，比如放在base theme地下）

为什么使用sub theme? 怎样定义?

- Drupal提供了大量的base theme,比如 zen, fusion等等, 他们定义好了基本的框架, 使用他们可以很简单的来创建自己sub theme.
- 一套base theme定义了一套设计资源, 那么可以使用多个sub theme来共享他们, 假设一个站点群, 那么就可以每个站点一个sub theme, 而整体包含了良好的外观一致性。
- 定义一个subtheme与其他常规 theme 一样, 只是在.info文件里多一个 `base theme = themeName`

我怎么做主题 （个人做法，可能不适合你）

- 美工设计好界面，切图假设生成html
- 我偏好fusion这个base theme,这个theme支持原生的下拉菜单，多个regions，满足了一般网站建设。
- 根据生成的静态页面，判断组织结构，比如可能某个轮播放在top region里，
- 大量使用block 而非panel来提到块状结构。
- 制作sub theme ,往往是修改css ,除非有大的页面结构调整进而修改page.tpl.php 等。

问题与解答



请把问题或者 意见发送到 qq1392137041@gmail.com ,或者论坛 www.chinaldap.com 或者直接了当加我qq: 1392137041,或者直接进入 qqtalk房间号码: 6289868

参考文献

- <http://drupal.org/node/171194>
- <http://drupal.org/node/173880>
- <http://zhupou.cn/book/export/html/8>
- <http://drupal.org/node/1089642> suggestion
- <http://api.drupal.org/api/drupal/modules--system--theme.api.php/group/themeable/7>
default theme function