合久必分，分久必合

洪强宁

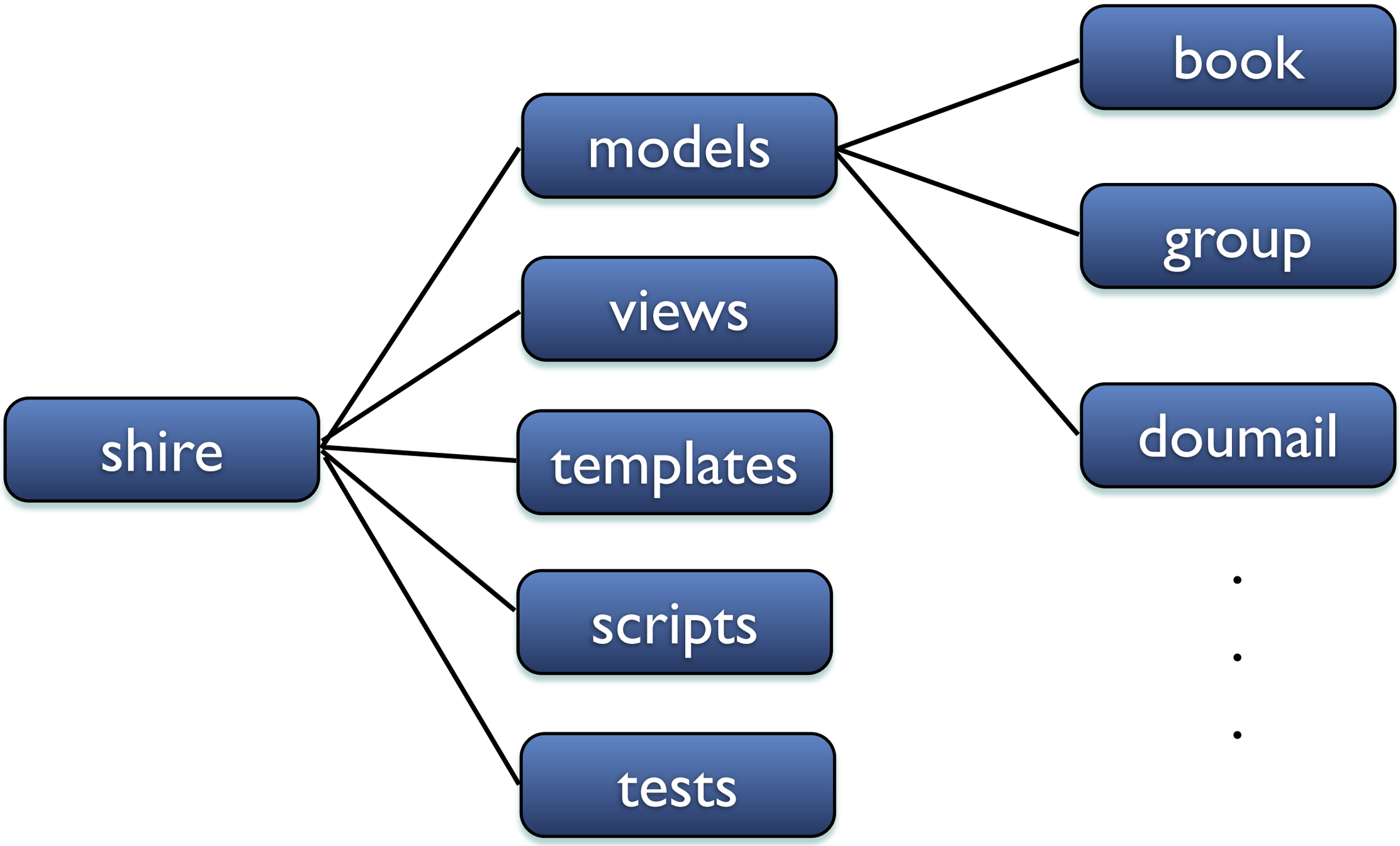
QCon Beijing 2012

豆瓣 douban

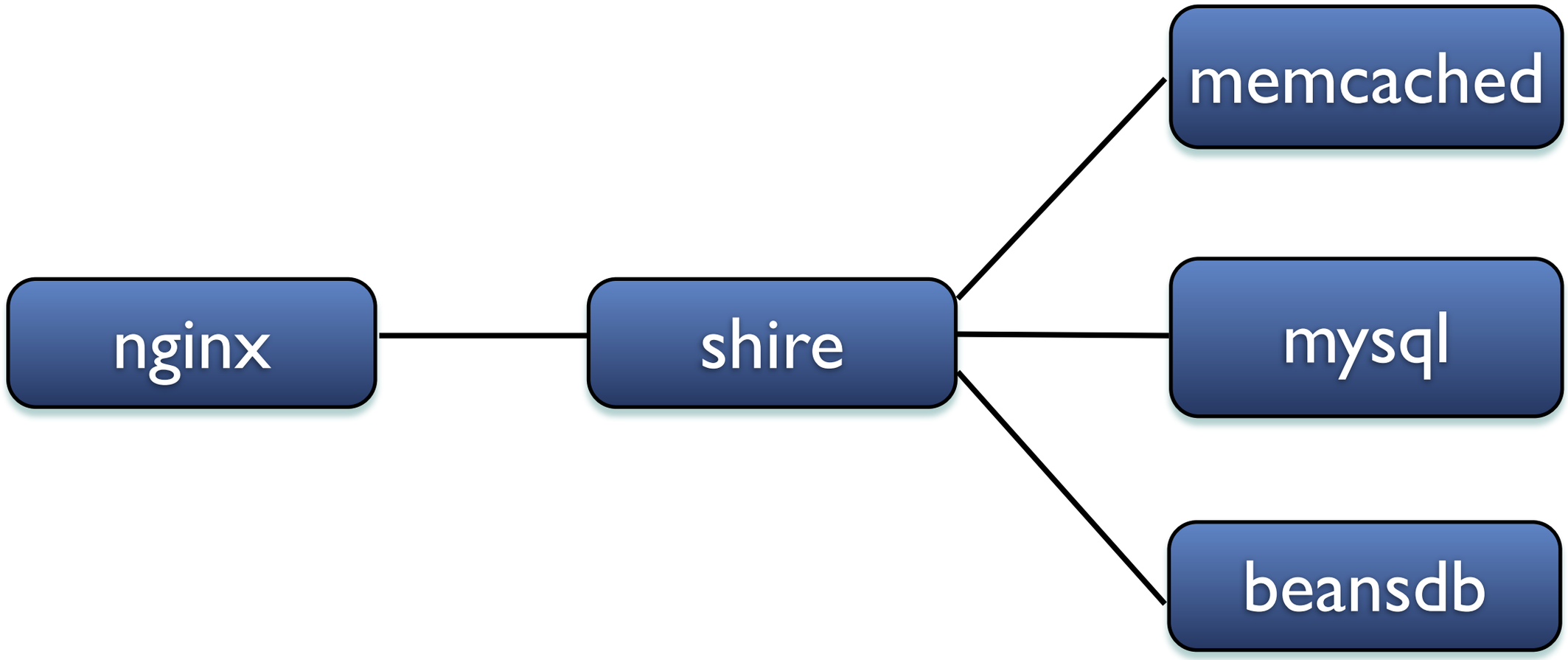
美好时代

豆瓣 douban

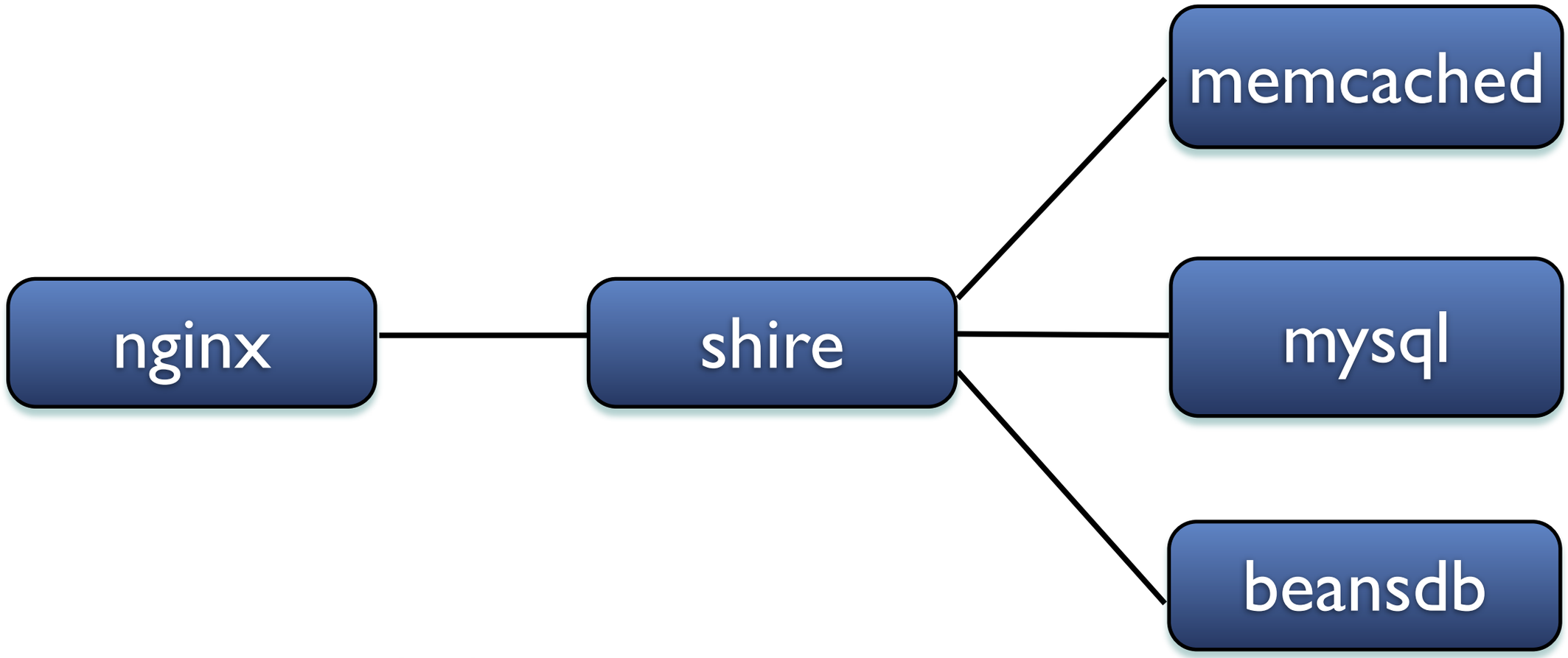
代码结构



部署方式



部署方式




```
server_name  9.douban.com;
rewrite (.*) /ninetaps/$1 last;
```

豆瓣 douban

然而

然而

- 时代在前进

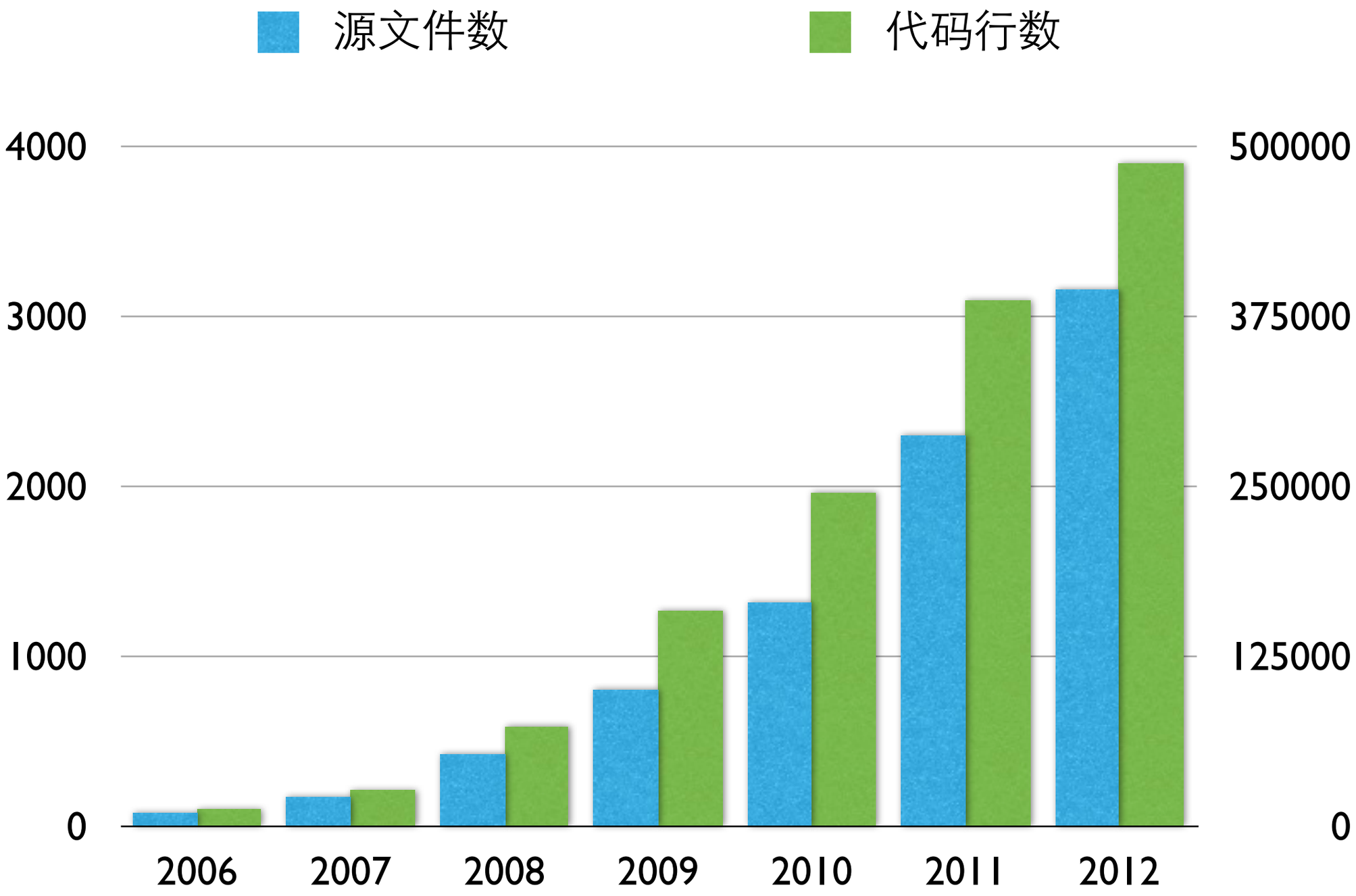
然而

- 时代在前进
- 产品在扩张

然而

- 时代在前进
- 产品在扩张
- 代码在膨胀

shire项目Python源文件数与代码行数



问题浮现

问题浮现

- 难以掌控全局

问题浮现

- 难以掌控全局
  - 新人上手变慢

问题浮现

- 难以掌控全局
  - 新人上手变慢
  - 新需求实现变困难

问题浮现

- 难以掌控全局
  - 新人上手变慢
  - 新需求实现变困难
- 代码质量下降

问题浮现

- 难以掌控全局
  - 新人上手变慢
  - 新需求实现变困难
- 代码质量下降
- 代码耦合度变高

问题浮现

- 难以掌控全局
  - 新人上手变慢
  - 新需求实现变困难
- 代码质量下降
- 代码耦合度变高
- 难以实现不同产品不同节奏

首页宕了?

首页宕了?

电影那边做了个改动……

豆瓣douban

解决方案：分

自发选择

自发选择

- 新项目想使用新技术

自发选择

- 新项目想使用新技术
- m.douban.com: Pylons

豆瓣douban

自发选择

- 新项目想使用新技术
- m.douban.com: Pylons
- 内部管理系统: Django

豆瓣 douban

自然选择

自然选择

- 产品拆分

自然选择

- 产品拆分
  - book站、movie站、alphatown...

豆瓣douban

自然选择

- 产品拆分
  - book站、movie站、alphatown...
- 建立独立服务的需求

豆瓣douban

自然选择

- 产品拆分
  - book站、movie站、alphatown...
- 建立独立服务的需求
  - IP定位服务

豆瓣 douban

自然选择

- 产品拆分
  - book站、movie站、alphatown...
- 建立独立服务的需求
  - IP定位服务
  - 小豆服务

如何分?

如何分?

- 库

如何分?

- 库
- 服务

库

库

- 性能无损耗

库

- 性能无损耗
- 升级发布麻烦

库

- 性能无损耗
- 升级发布麻烦
- 依赖注入

服务

服务

- 升级发布方便

服务

- 升级发布方便
- 客户端依赖轻

服务

- 升级发布方便
- 客户端依赖轻
- 性能损耗

服务

- 升级发布方便
- 客户端依赖轻
- 性能损耗
- 额外的故障点

服务

- 升级发布方便
- 客户端依赖轻
- 性能损耗
- 额外的故障点
- 开发环境搭建复杂

适合服务的场景

适合服务的场景

- 需要分开部署

适合服务的场景

- 需要分开部署
- 需要独立维护

适合服务的场景

- 需要分开部署
- 需要独立维护
- 需要错误隔离

适合服务的场景

- 需要分开部署
- 需要独立维护
- 需要错误隔离
- 需要节省资源

适合服务的场景

- 需要分开部署
- 需要独立维护
- 需要错误隔离
- 需要节省资源
- 跨语言/跨平台

重新调整主站代码结构

重新调整主站代码结构

- 按照产品线切分包

重新调整主站代码结构

- 按照产品线切分包
- 定义包接口

重新调整主站代码结构

- 按照产品线切分包
- 定义包接口
- 公共代码单独建包

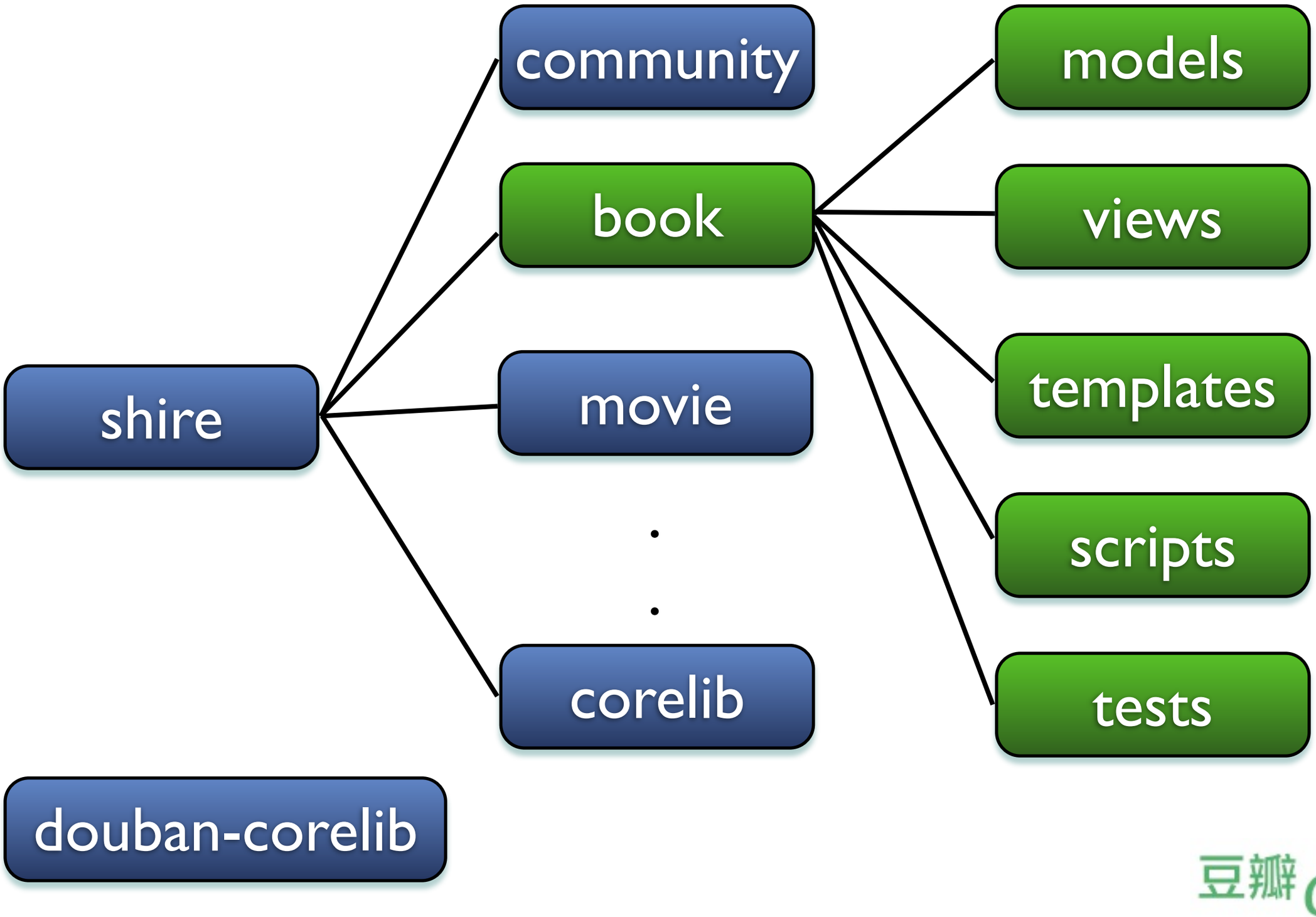
重新调整主站代码结构

- 按照产品线切分包
- 定义包接口
- 公共代码单独建包
- 限制提交权限

重新调整主站代码结构

- 按照产品线切分包
- 定义包接口
- 公共代码单独建包
- 限制提交权限
- 基础代码独立成 douban-corelib 库

新代码结构



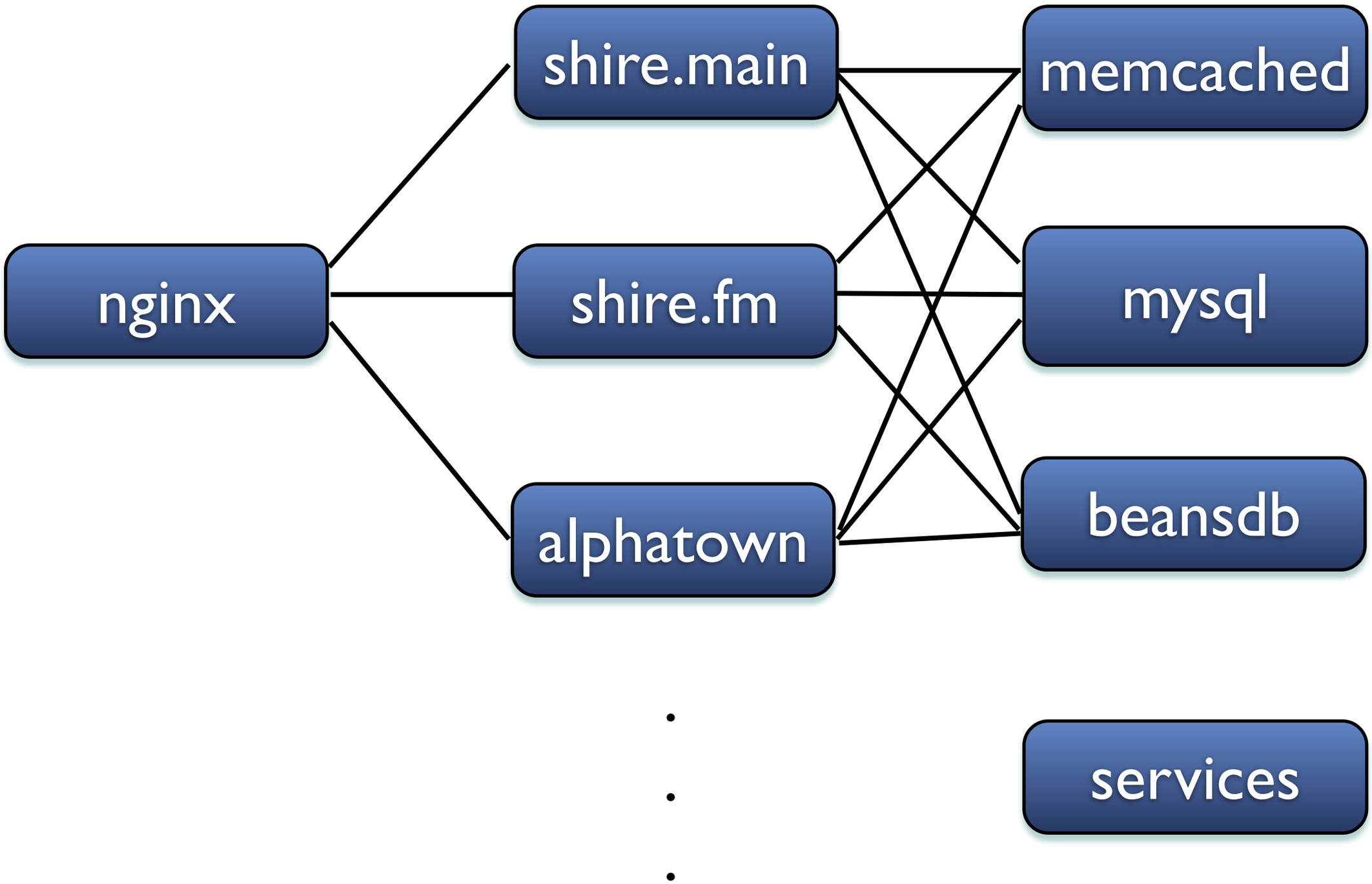
部署方式调整

部署方式调整

- 建立多个app server实例

部署方式调整

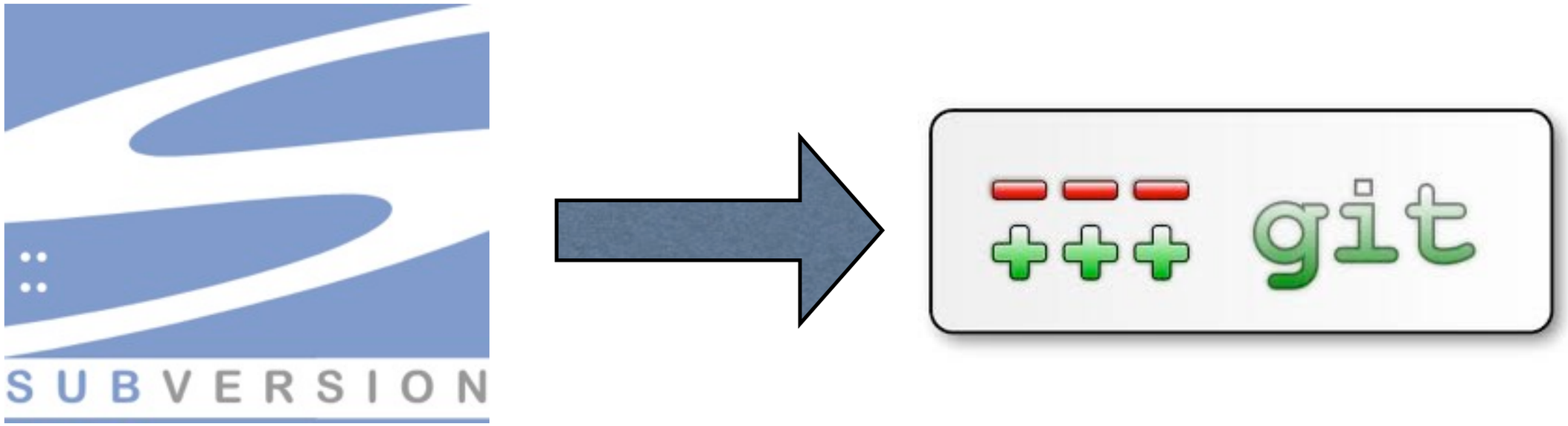
- 建立多个app server实例
- nginx 分发到不同后端


shire.main
memcached
nginx
shire.fm
mysql
alphatown
beansdb
services

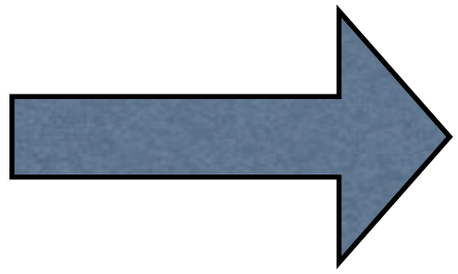
douban

开发方式调整

开发方式调整



开发方式调整



github:enterprise

豆瓣douban

分带来的问题

重复开发

重复开发

- web框架

豆瓣douban

重复开发

- web框架
- 测试框架和持续集成

重复开发

- web框架
- 测试框架和持续集成
- 日志处理

重复开发

- web框架
- 测试框架和持续集成
- 日志处理
- 开发环境

重复开发

- web框架
- 测试框架和持续集成
- 日志处理
- 开发环境
- 上线脚本

管理混乱

管理混乱

- 资源申请随意

管理混乱

- 资源申请随意
- 权限不能统一管理

管理混乱

- 资源申请随意
- 权限不能统一管理
- 出现资源竞争情况

管理混乱

- 资源申请随意
- 权限不能统一管理
- 出现资源竞争情况
- 监控方式随意、缺失、不一致、不可控

管理混乱

- 资源申请随意
- 权限不能统一管理
- 出现资源竞争情况
- 监控方式随意、缺失、不一致、不可控
- 配置管理不可靠

管理混乱

- 资源申请随意
- 权限不能统一管理
- 出现资源竞争情况
- 监控方式随意、缺失、不一致、不可控
- 配置管理不可靠
- 对服务的调用方式千差万别，难以控制

历史经验未能最大化应用

历史经验未能最大化应用

- 服务软件参数配置

历史经验未能最大化应用

- 服务软件参数配置
- fail over 处理

不能同步进化

不能同步进化

- 代码复用率下降

一个应用想使用另一个应用的代码麻烦

一个应用想使用另一个应用的代码麻烦

- 修改 sys.path

一个应用想使用另一个应用的代码麻烦

- 修改 sys.path
- 依赖代码变更后需要重启，否则可能出现诡异问题

解决方案：合

需求

需求

- 整合最基础的框架

豆瓣douban

需求

- 整合最基础的框架
- 创建新项目无痛化

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式
- 统一监控

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式
- 统一监控
- 统一资源管理

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式
- 统一监控
- 统一资源管理
- 以API形式提供基础设施使用接口

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式
- 统一监控
- 统一资源管理
- 以API形式提供基础设施使用接口
- 依赖管理

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式
- 统一监控
- 统一资源管理
- 以API形式提供基础设施使用接口
- 依赖管理
- 提供统一的开发、测试、调试环境

需求

- 整合最基础的框架
- 创建新项目无痛化
- 统一部署方式
- 统一监控
- 统一资源管理
- 以API形式提供基础设施使用接口
- 依赖管理
- 提供统一的开发、测试、调试环境
- 同时尽量少的限制应用开发者

私有云

私有云

- DAE (Douban App Engine)

豆瓣 douban

PaaS

豆瓣douban

PaaS

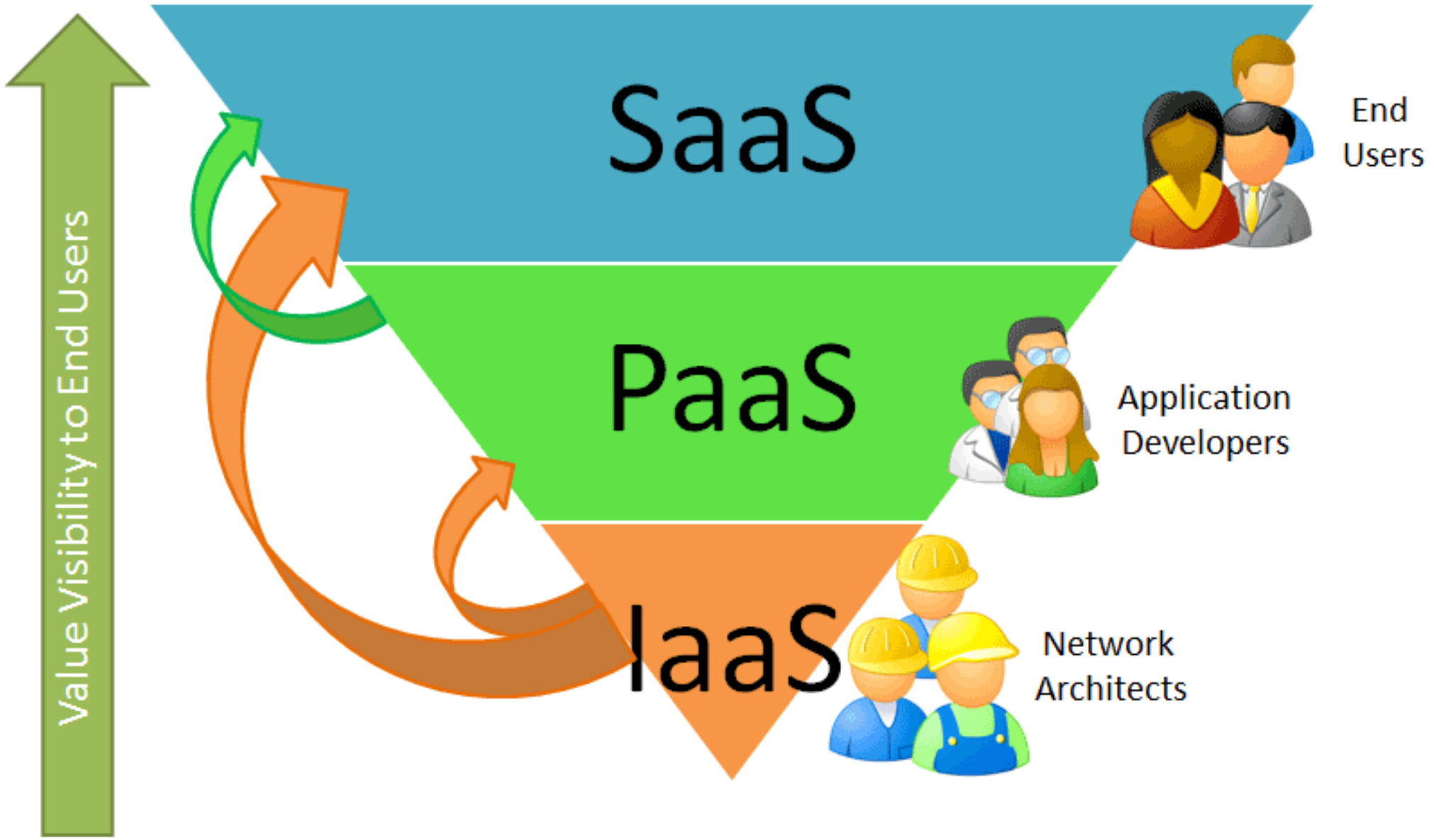
- IaaS - Infrastructure as a Service (EC2)

PaaS

- IaaS - Infrastructure as a Service (EC2)
- PaaS - Platform as a Service (GAE)

PaaS

- IaaS - Infrastructure as a Service (EC2)
- PaaS - Platform as a Service (GAE)
- SaaS - Software as a Service (Gmail)


Value Visibility to End Users
SaaS
End Users
PaaS
Application Developers
IaaS
Network Architects

面向内部开发者

面向内部开发者

- 强调功能和性能

豆瓣douban

面向内部开发者

- 强调功能和性能
- 安全性可以放宽

面向内部开发者

- 强调功能和性能
- 安全性可以放宽
- 主要防范无心做错事

面向内部开发者

- 强调功能和性能
- 安全性可以放宽
- 主要防范无心做错事
- 但也需要有完善的操作权限和操作日志

为百至千级别的应用设计

为百至千级别的应用设计

- 权限隔离直接使用UNIX用户权限

为百至千级别的应用设计

- 权限隔离直接使用UNIX用户权限
- 对应用区分优先级别

为百至千级别的应用设计

- 权限隔离直接使用UNIX用户权限
- 对应用区分优先级别
  - 资源紧张时优先保证高优先级应用

开发语言支持

开发语言支持

- Python

开发语言支持

- Python
- 支持Python的C扩展

开发语言支持

- Python
- 支持Python的C扩展
- Python/C API

开发语言支持

- Python
- 支持Python的C扩展
  - Python/C API
  - Cython/Pyrex

开发语言支持

- Python
- 支持Python的C扩展
  - Python/C API
  - Cython/Pyrex
- 未来会支持其他语言（如 go）

依赖管理和隔离

依赖管理和隔离

- virtualenv + pip

豆瓣douban

依赖管理和隔离

- virtualenv + pip
- 每个app有自己独立的virtualenv

豆瓣 douban

依赖管理和隔离

- virtualenv + pip
  - 每个app有自己独立的virtualenv
- dae install

依赖管理和隔离

- virtualenv + pip
  - 每个app有自己独立的virtualenv
- dae install
  - 是 pip install 的封装，更新 pip-req.txt

示例

示例

```
$ dae create daetest
```

示例

```
$ dae create daetest
$ dae install web.py
```

豆瓣 douban

示例

```
$ dae create daetest
$ dae install web.py
$ vim app.py
```

示例

```
$ dae create daetest
$ dae install web.py
$ vim app.py
```

```
import web

urls = ('/', 'index')

class index:
    def GET(self):
        return "Hello, DAE"

app = web.application(urls, globals()).wsgifunc()
```

示例

```
$ dae create daetest
$ dae install web.py
$ vim app.py
```

```
import web

urls = ('/', 'index')

class index:
    def GET(self):
        return "Hello, DAE"

app = web.application(urls, globals()).wsgifunc()
```

```
$ dae serve
```

示例

```
$ dae create daetest
$ dae install web.py
$ vim app.py
```

```
import web

urls = ('/', 'index')

class index:
    def GET(self):
        return "Hello, DAE"

app = web.application(urls, globals()).wsgifunc()
```

```
$ dae serve
$ dae deploy
```

豆瓣douban

示例

```
$ dae create daetest
$ dae install web.py
$ vim app.py
```

```
import web

urls = ('/', 'index')

class index:
    def GET(self):
        return "Hello, DAE"

app = web.application(urls, globals()).wsgifunc()
```

```
$ dae serve
$ dae deploy
```

open http://daetest.dapps.douban.com

豆瓣 douban

app.yaml

app.yaml

- 应用配置中心

app.yaml

- 应用配置中心
- wsgi apps

app.yaml

- 应用配置中心
- wsgi apps
- daemons

app.yaml

- 应用配置中心
- wsgi apps
- daemons
- cron

app.yaml

- 应用配置中心
- wsgi apps
- daemons
- cron
- services

app.yaml

- 应用配置中心
- wsgi apps
- daemons
- cron
- services
- protected_files

用API形式提供基础设施功能

用API形式提供基础设施功能

- mysql

用API形式提供基础设施功能

- mysql
- memcache

用API形式提供基础设施功能

- mysql
- memcache
- beansdb

豆瓣douban

用API形式提供基础设施功能

- mysql
- memcache
- beansdb
- filesystem storage (MooseFS)

豆瓣 douban

用API形式提供基础设施功能

- mysql
- memcache
- beansdb
- filesystem storage (MooseFS)
- task queue

用API形式提供基础设施功能

- mysql
- memcache
- beansdb
- filesystem storage (MooseFS)
- task queue
- scheduled task

用API形式提供基础设施功能

- mysql
- memcache
- beansdb
- filesystem storage (MooseFS)
- task queue
- scheduled task
- user login

用API形式提供基础设施功能

- mysql
- memcache
- beansdb
- filesystem storage (MooseFS)
- task queue
- scheduled task
- user login
- dpark

用API形式提供基础设施功能

- mysql
- memcache
- beansdb
- filesystem storage (MooseFS)
- task queue
- scheduled task
- user login
- dpark
- 每个app拥有独立的namespace

整合已有技术经验

整合已有技术经验

- LVS

整合已有技术经验

- LVS
- nginx

整合已有技术经验

- LVS
- nginx
- deploy 流程

整合已有技术经验

- LVS
- nginx
- deploy 流程
- thrift

整合已有技术经验

- LVS
- nginx
- deploy 流程
- thrift
- onimaru - error collector based on django-sentry

整合已有技术经验

- LVS
- nginx
- deploy 流程
- thrift
- onimaru - error collector based on django-sentry
- scribe

整合已有技术经验

- LVS
- nginx
- deploy 流程
- thrift
- onimaru - error collector based on django-sentry
- scribe
- puppet

整合已有技术经验

- LVS
- nginx
- deploy 流程
- thrift
- onimaru - error collector based on django-sentry
- scribe
- puppet
- 监控系统

尝试新技术

尝试新技术

- gunicorn -- a fast wsgi server

豆瓣douban

尝试新技术

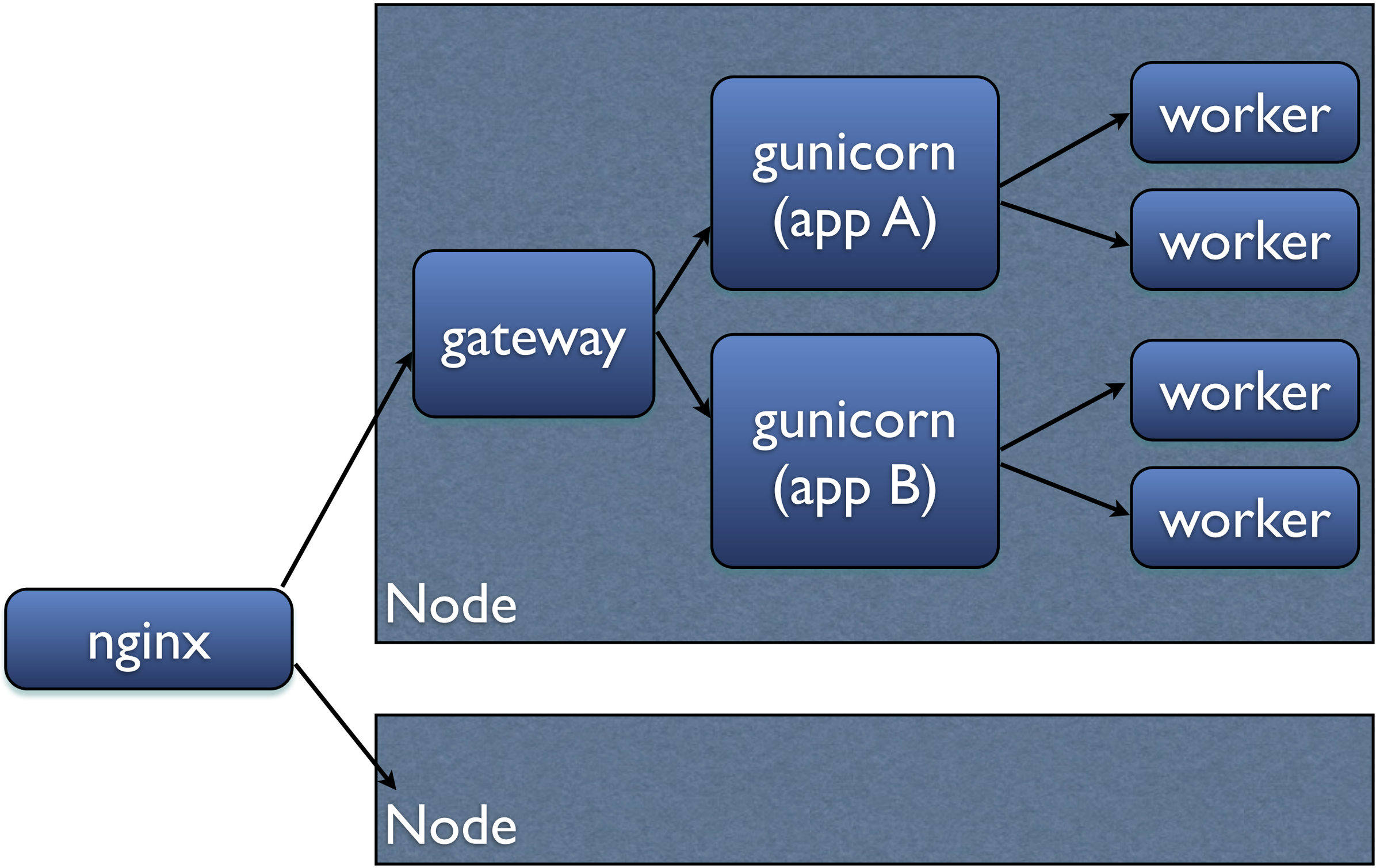
- gunicorn -- a fast wsgi server
- gevent -- coroutine library based on greenlet and libev

尝试新技术

- gunicorn -- a fast wsgi server
- gevent -- coroutine library based on greenlet and libev
- websocket support

尝试新技术

- gunicorn -- a fast wsgi server
- gevent -- coroutine library based on greenlet and libev
- websocket support
- mesos


gateway
gunicorn
(app A)
worker
worker
gunicorn
(app B)
worker
worker
Node
nginx
Node

豆瓣 douban

项目间代码调用

项目间代码调用

- 库

项目间代码调用

- 库
- 服务

库

库

- setuptools (setup.py)

库

- setuptools (setup.py)
- DAE系统记录下安装的版本（利用 pip freeze）

库

- setuptools (setup.py)
- DAE系统记录下安装的版本（利用 pip freeze）
- 部署时保证服务器上版本一致（利用 pip install -r）

库

- setuptools (setup.py)
- DAE系统记录下安装的版本（利用 pip freeze）
- 部署时保证服务器上版本一致（利用 pip install -r）
- 提供更新通知机制，向库使用者通知代码更新

依赖库版本

依赖库版本

- 自动记录在 pip-req.txt 中

依赖库版本

- 自动记录在 pip-req.txt 中
- PyPI上发布的软件：发布版本

依赖库版本

- 自动记录在 pip-req.txt 中
- PyPI上发布的软件：发布版本
- 代码仓库中的软件：revision

依赖库版本

- 自动记录在 pip-req.txt 中
- PyPI上发布的软件：发布版本
- 代码仓库中的软件：revision

```
-e hg+http://hghub.dapps.douban.com/douban-
corelib@fb367759be2e1b37c77701d59be6514a3b39837e#egg=DoubanCoreLib
distribute==0.6.19
web.py==0.36
wsgiref==0.1.2
```

豆瓣 douban

服务

服务

- DAE Service

服务

- DAE Service
- thrift 接口定义

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口
- gunicorn gevent worker

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口
- gunicorn gevent worker
- 用DNS实现服务寻址和权重

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口
- gunicorn gevent worker
- 用DNS实现服务寻址和权重
  - 直接定位到服务提供者

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口
- gunicorn gevent worker
- 用DNS实现服务寻址和权重
  - 直接定位到服务提供者
  - 未来会使用zookeeper实现路由推送

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口
- gunicorn gevent worker
- 用DNS实现服务寻址和权重
  - 直接定位到服务提供者
  - 未来会使用zookeeper实现路由推送
- 在客户端实现fail over、异常处理

服务

- DAE Service
- thrift 接口定义
- 在 app.yaml 中定义实现代码入口
- gunicorn gevent worker
- 用DNS实现服务寻址和权重
  - 直接定位到服务提供者
  - 未来会使用zookeeper实现路由推送
- 在客户端实现fail over、异常处理
- 与DAE集成

user.thrift

```
struct UserProfile {
      1: i32 uid,
      2: string name,
      3: string blurb
}
service UserStorage {
      void store(1: UserProfile user),
      UserProfile retrieve(1: i32 uid)
}
```

user.thrift

```
struct UserProfile {
      1: i32 uid,
      2: string name,
      3: string blurb
}
service UserStorage {
      void store(1: UserProfile user),
      UserProfile retrieve(1: i32 uid)
}
```

app.yaml

```
services:
- interface: user.UserStorage
  handler: user.handler:UserStorageHandler
```

豆瓣 douban

user.thrift

```
struct UserProfile {
      1: i32 uid,
      2: string name,
      3: string blurb
}
service UserStorage {
      void store(1: UserProfile user),
      UserProfile retrieve(1: i32 uid)
}
```

app.yaml

```
services:
- interface: user.UserStorage
  handler: user.handler:UserStorageHandler
```

handler.py

```
class UserStorageHandler(object):
    def store(self, user):
        ...
    def retrieve(self, uid):
        ...
```

豆瓣 douban

DAE Service 客户端

DAE Service 客户端

```
$ dae service gen_client
```

豆瓣douban

DAE Service 客户端

```
$ dae service gen_client
```

```
from daetest_client import UserStorage
user = UserStorage.retrieve(1)
```

DAE Service 客户端

```
$ dae service gen_client
```

```
from daetest_client import UserStorage
user = UserStorage.retrieve(1)
```

在客户端可进行超时、重试、调用失败时的默认值等配置

服务治理

服务治理

- The next big issue

服务治理

- The next big issue
- Done is better than perfect

豆瓣douban

超级战舰 - 豆瓣电影

m.d

0/2

24

豆瓣电影

热映中 | 标签 | Top250 | 我看

搜电影

超级战舰 6.9

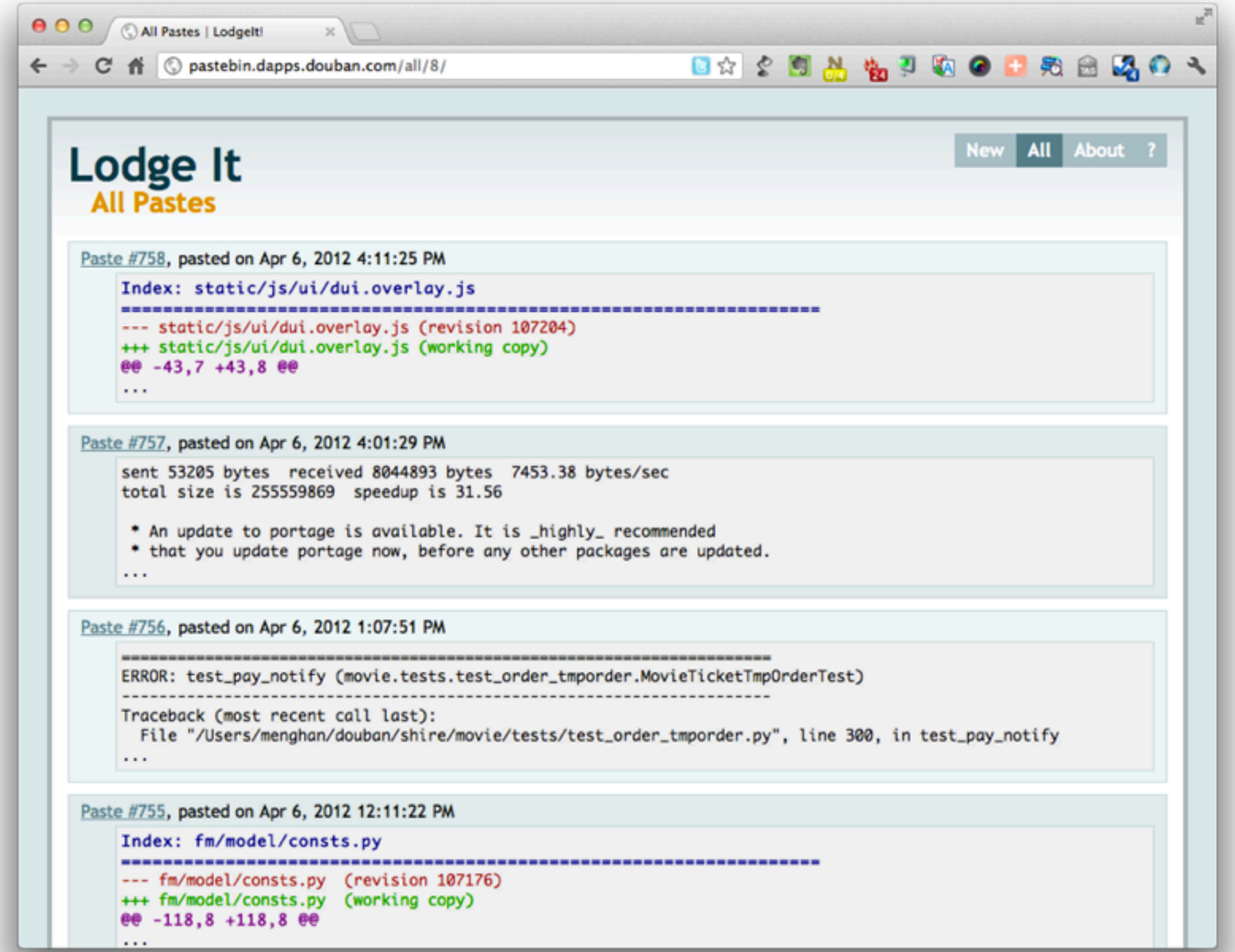
查询上映影院和时间

想看 | 看过 | 47人推荐 | 推荐 (请先登录)

2012 / 动作 战争 科幻 惊悚 / 132分钟 / 美国

彼得·博格 (导演) / 连姆·尼森 / 亚历山大·斯卡斯加德 / 泰勒·克奇 / 乔什·平茨 / 布鲁克琳·黛克 / 蕾哈娜 / 杰西·普莱蒙 / 浅野忠信

故事灵感来自孩之宝同名畅销战棋游戏“战舰攻防战”， 电影讲述海军中尉Alex（泰勒·克奇 Taylor Kitsch 饰）被上级派往美国飞弹驱逐舰 (USS John Paul


All Pastes | LodgeIt!
pastebin.dapps.douban.com/all/8/
Lodge It
All Pastes
New
All
About
?
Paste #758, pasted on Apr 6, 2012 4:11:25 PM
Index: static/js/ui/dui.overlay.js
===
--- static/js/ui/dui.overlay.js (revision 107204)
+++ static/js/ui/dui.overlay.js (working copy)
@@ -43,7 +43,8 @@
...
Paste #757, pasted on Apr 6, 2012 4:01:29 PM
sent 53205 bytes received 8044893 bytes 7453.38 bytes/sec
total size is 255559869 speedup is 31.56
* An update to portage is available. It is _highly_ recommended
* that you update portage now, before any other packages are updated.
...
Paste #756, pasted on Apr 6, 2012 1:07:51 PM
==
ERROR: test_pay_notify (movie.tests.test_order_tmporder.MovieTicketTmpOrderTest)

Traceback (most recent call last):
File "/Users/menghan/douban/shire/movie/tests/test_order_tmporder.py", line 300, in test_pay_notify
...
Paste #755, pasted on Apr 6, 2012 12:11:22 PM
Index: fm/model/consts.py
===
--- fm/model/consts.py (revision 107176)
+++ fm/model/consts.py (working copy)
@@ -118,8 +118,8 @@
...

豆瓣 douban

豆娘订购系统

doutea.dapps.douban.com/create_order

豆瓣 douban 豆娘订购系统

欢迎，hongqn 首页 退出

请先选择餐厅

CoCo都可茶饮 电话: 137-1891-1577 6杯起送，满10杯送1杯	**东池便当（饭不干净，送的慢，打电话骂人）** 电话: 84564779,84564819 10:00-20:00	**九格便当** 电话: 5722-5064,5745-1115 6份送大桶饮料
桂林米粉 电话: 86511179,15101649448	**久昌** 电话: 84798701、64364712、64361779 无	**MISSMILK酸奶吧** 电话: 158 1011 2693 营业9am-11pm, 20元起免费送
佳味多台湾食品 电话: 51357745 套餐里可饭、汤分开点	**椅子团购(by酱油海盗)** 电话: @PhayTsukiming http://www.douban.com/people/PhayTsukiming/status/894646453/	

请选择截团时间

- ◉ 在 60 分钟后结束
- ○ 在 16 : 30 结束 （小于当前时间的将被当作第二天）

豆瓣 douban

分分合合

分：应用专心于应用自身的逻辑

合：平台成为应用和基础设施之间的纽带

一点感想

- 分要疾风骤雨
- 合要和风细雨
- 作为平台开发者，需要从分的过程中观察合的需求
- 平台的可靠性和稳定性非常重要
- 好用的才会有人用

Q & A

你也可以通过下列方式找到我：
http://www.douban.com/people/hongqn/
hongqn@douban.com
twitter: @hongqn
新浪微博: @hongqn

豆瓣 douban

Thanks

豆瓣 douban

ArchSummit

中国·深圳 2012.08

INTERNATIONAL ARCHITECT SUMMIT

全 球 架 构 师 峰 会

详情请访问：architectsummit.com

- 3天 • 6场主题演讲
- 3场圆桌论坛 • 9场专题会议
- 国内外30余家IT、互联网公司的50多位来自一线的讲师齐聚一堂

主办方：InfoQ 战略合作伙伴：Tencent 腾讯 特别支持：

http://architectsummit.com


QCon
杭州站 · 2012年10月25日~27日
www.qconhangzhou.com（6月启动）
QCon北京站官方网站和资料下载
www.qconbeijing.com