

ECAE

历程与风景

ECAE?

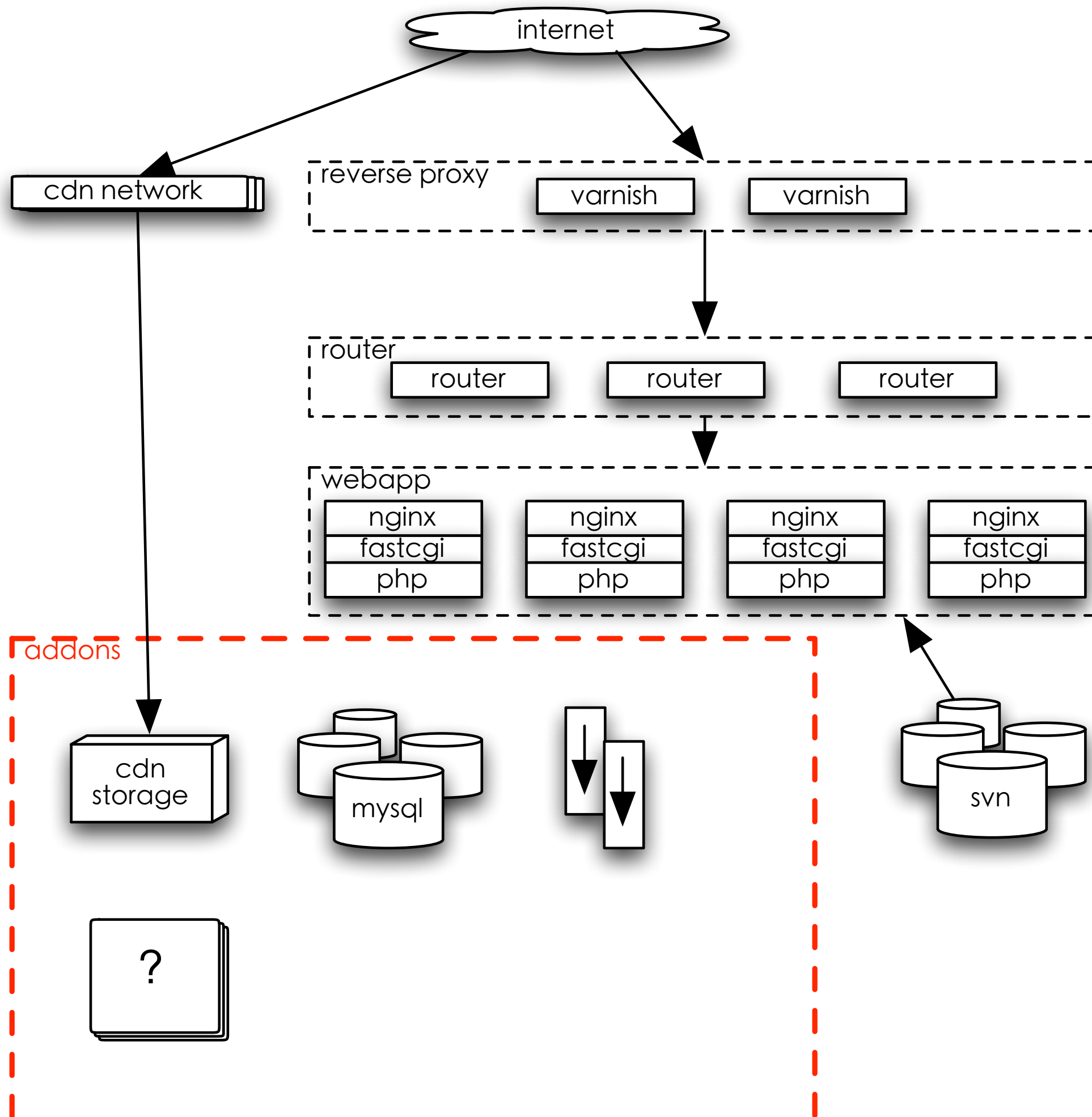
- ECAE = AE for EC
- 前端PaaS暂时仅支持php
- 两种产品支持
 - 中高端: PaaS
 - 低端: SaaS

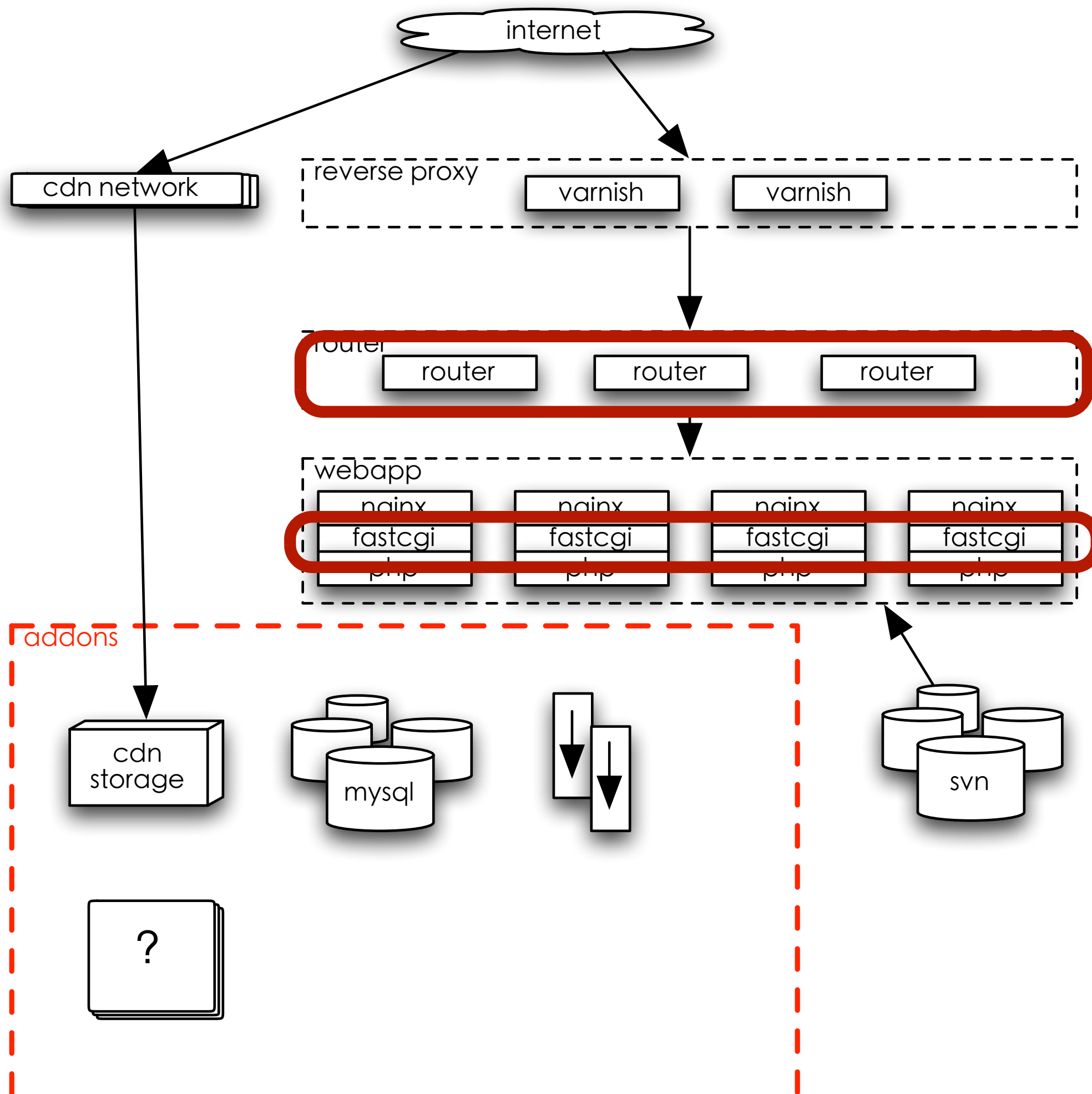
业务模式

- 支持多级代理商：代理商的域
- saas用户可以升级为paas：源码库fork

主要矛盾

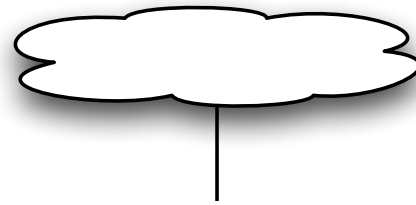
以精确控制**PHP**进程为切入点





Erlang

Router



www.abc.com

支持ab test, path的多种格式

```
[{"site/2002/trunk", 80},  
 {"site/2002/branch/a", 10},  
 {"site/2002/branch/b", 10}]
```

accept

查site

mnesia:domain

domain	site
www.xxx.com	2001
www.abc.com	2002
www.foo.com	2003
...	...

mnesia:site

site	scm	active	...	path
2001	{svn,xxx}	true
2002	...	true	...	site/2002
2003	...	false
...

mnesia:worker

site	server	free	worker	status
..
2002	{192,168,0,31}	10	50	ok
2002	{192,168,0,30}	40	50	ok
2002	{192,168,0,33}	50	50	syncing
...

按比例随机选取

192,168,0,31 = 0.2
192,168,0,32 = 0.8

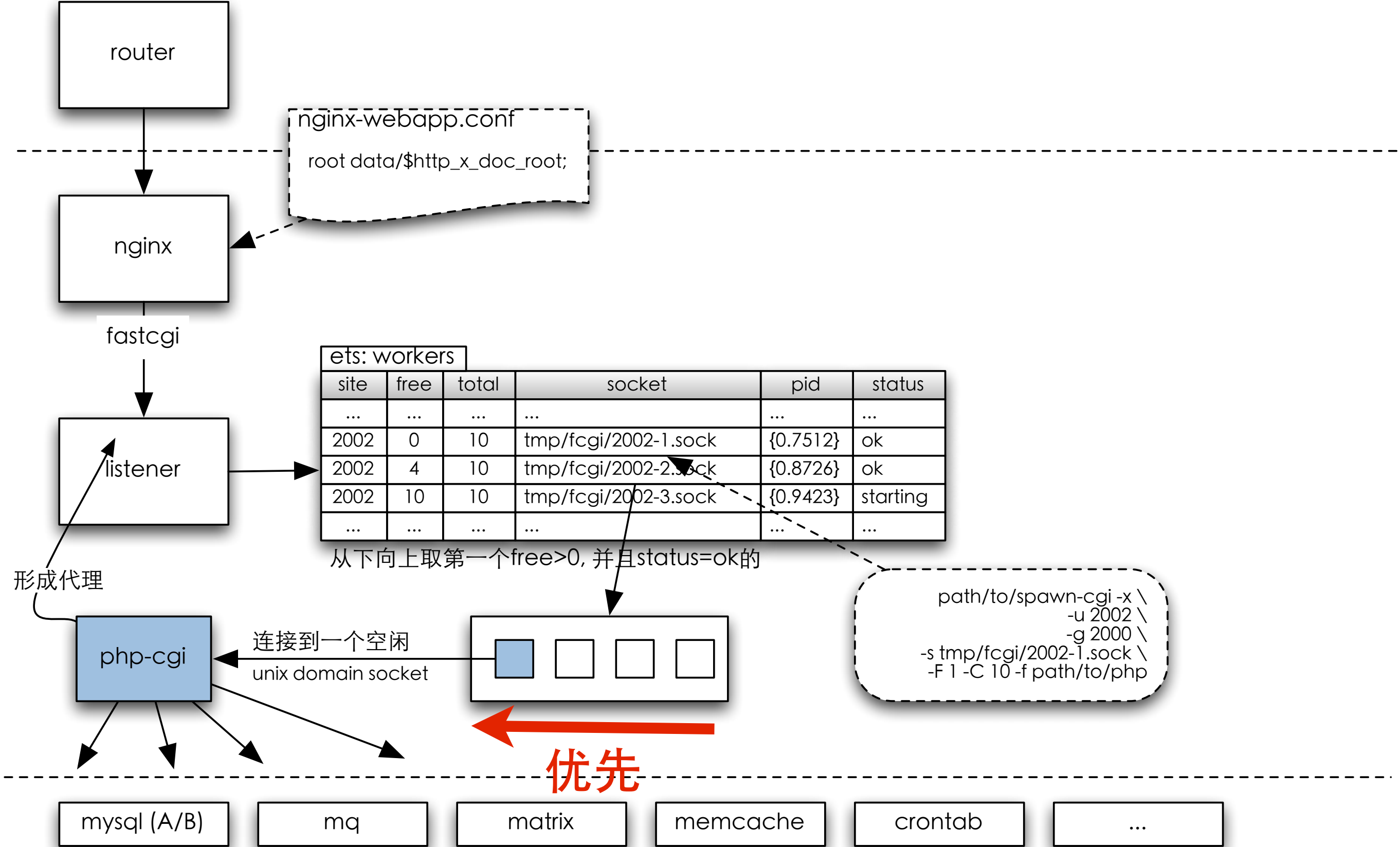
http header

X-Site-Id: 2002
X-Doc-Root: site/2002

传给后端, 记录传输的字节

192,168,0,32 (nginx)

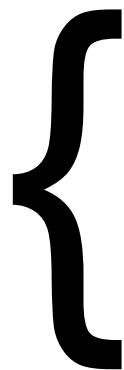
webapp



fastcgi代理

来自Router

来自nginx



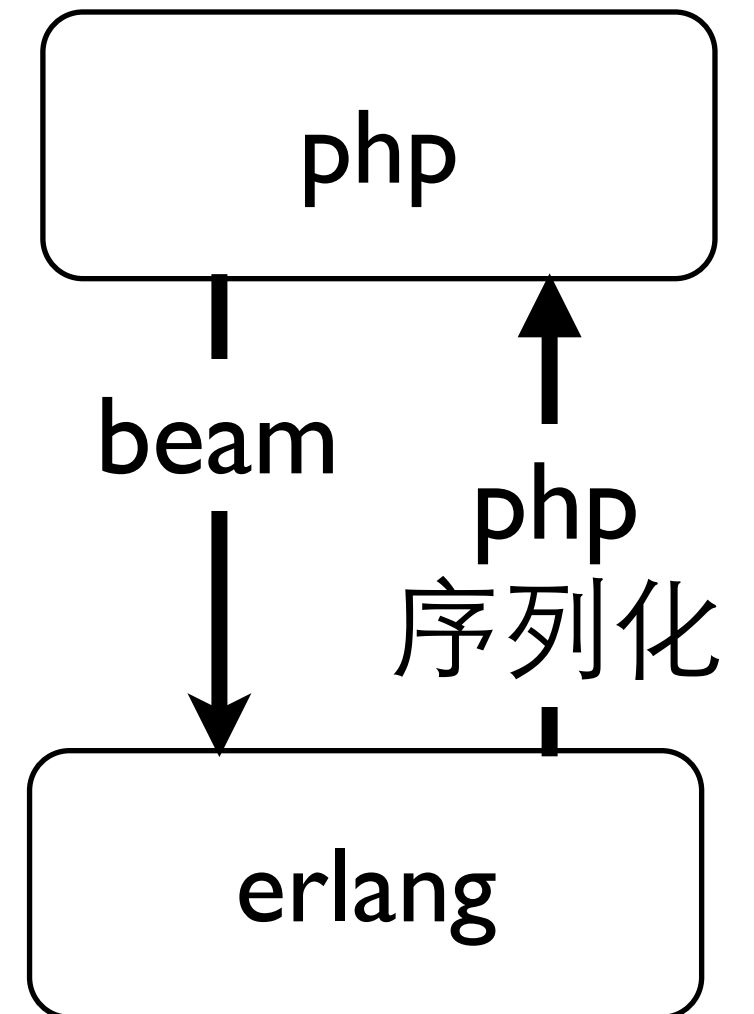
HTTP_X_SITE_ID
PATH_INFO
QUERY_STRING
SCRIPT_NAME
ECAE_MYSQL_XXX
ECAE_MCACHE_XXX
_ECAE_SESSION_SERVER



环境信息

php-erlang api

super api	agent_list/2 is_agent/2 ...
代理商api	site_delete/3 site_list_user/2 site_create/3 ...
普通api	storage_save/4 ...



事件决策引擎

- 当某机器当机时，自动迁移上面的应用到其他机器
- 当某webapp节点死掉时，自动迁移进程到其他机器
- 当xxx时， xxxxxx.

脚本引擎

```
#!/bin/sh
# title = 备份MySQL数据库
# args=备份文件名 storage:/private/backup/?

${mysqldump} --add-drop-database \
  --comments ${comment} \
  --password="${ECAE_MYSQL_PASS}" \
  --user="${ECAE_MYSQL_USER}" \
  --host=`echo "${ECAE_MYSQL_HOST_S}" | awk -F':' '{print $1}'` \
  --port=`echo "${ECAE_MYSQL_HOST_S}" | awk -F':' '{print $2}'` \
  --result-file=backup.sql \
  --verbose \
  ${ECAE_MYSQL_DB} && \
storage-save private backup.sql /backup/$1
rm -f backup.sql%
```

内建服务

memcache	增加密码支持
mongo	replset形成环状集群
session	php插件负责连接: 从_ECAC变量中获取环境信息

crond

- 二进制存储时间信息
- 每分钟扫描一次放入任务队列
- 使用erlang的httpc的流量控制特性

启动外部程序

- 一切外部程序必须由erlang控制启动
- 不仅仅要启动， 还要在erlang退出时一起退出

启动外部程序

```
#!/bin/sh
pid=${tmpdir}"/memcached-${memcache_name}.pid"
${memcached} -P ${pid} &
cat && \
if [ -e ${pid} ]
then
    kill `cat ${pid}`
fi
```

Ports退出时会发送\0

- 无法知晓新进程的状况
- 有的程序启动到服务需要一段时间(mongo等...)

启动外部程序

Usage: launch [Options] Program [args]

Options:

-u <user> change to user-id
-g <group> change to group-id (default: primary group of user if -u
-k <code> kill signal

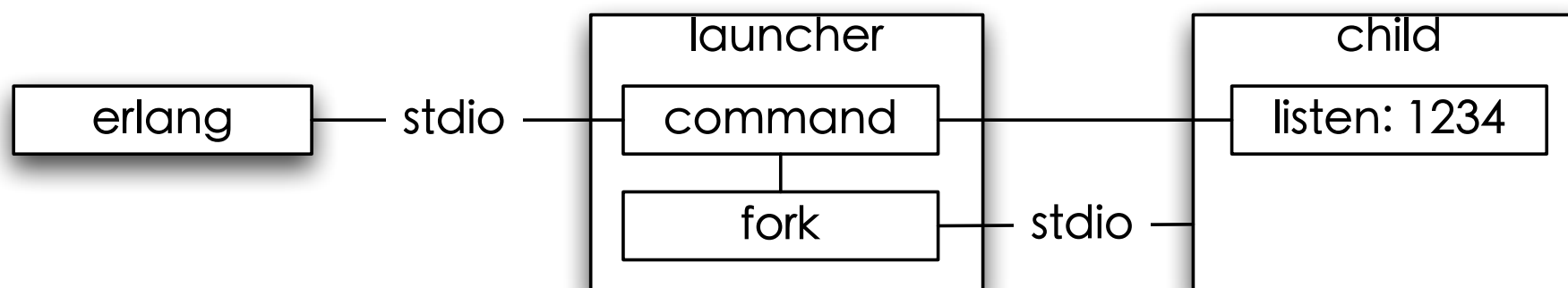
Running test:

-p <port> test to TCP-port

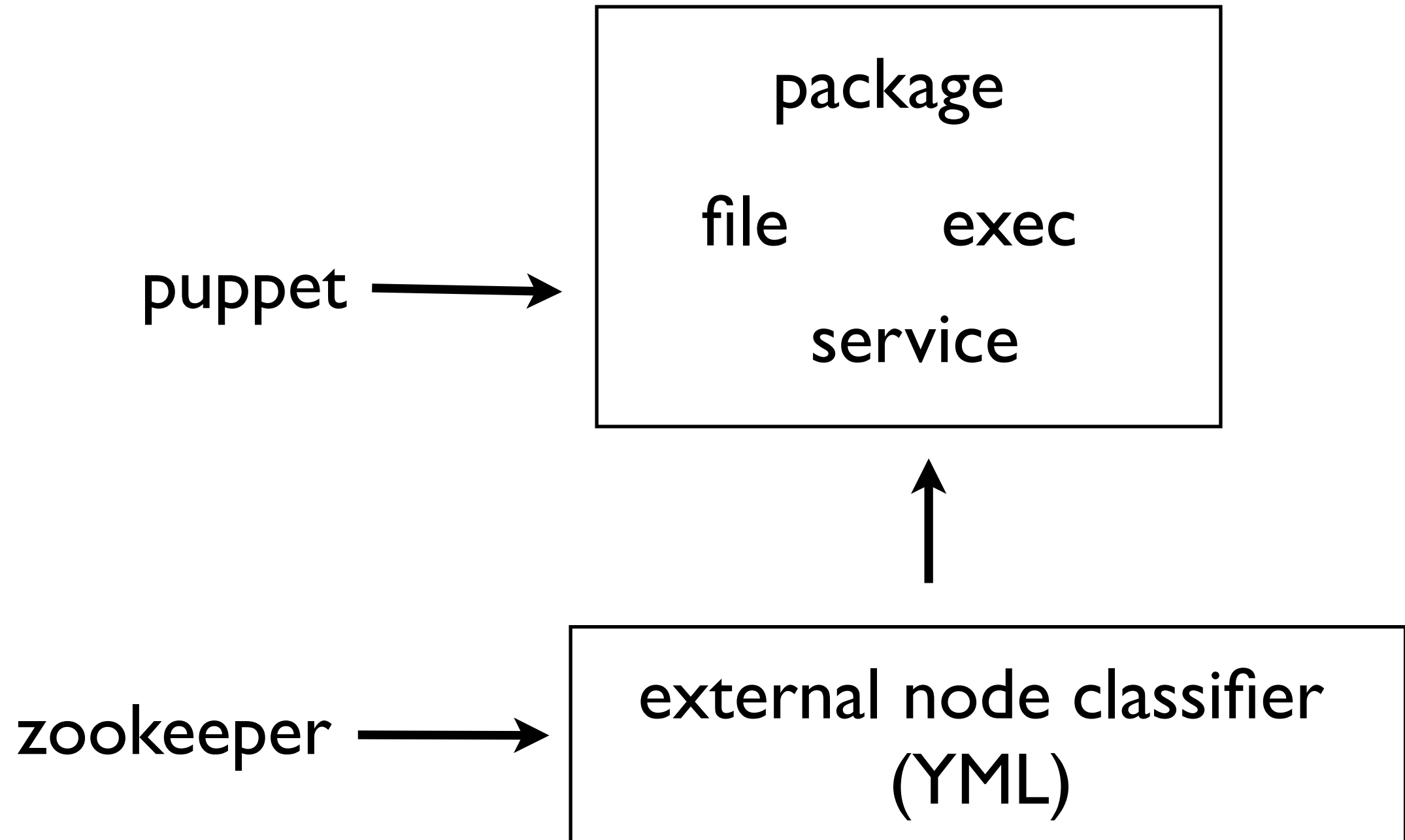
Process:

-d <pidfile> daemon mode

改进版

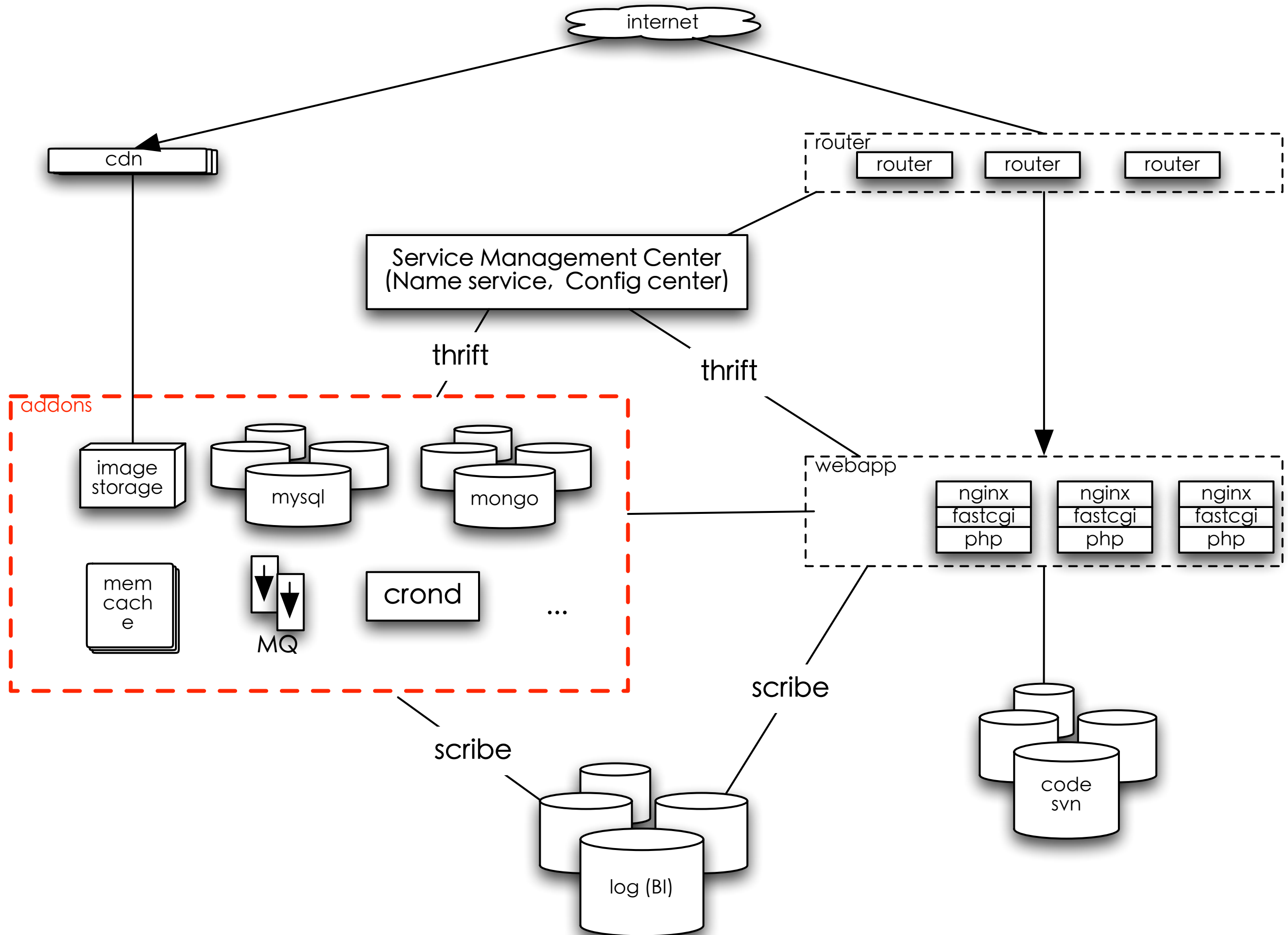


工具



现状

- 过度依赖erlang，整体需要“去erlang化”
- 很多环节只是临时方案(代码分发机制，存储等等)
- puppet和thrift会保留， zookeeper不可控



谢谢大家

- mail/gtalk: flaboy.cn@gmail.com
- <http://weibo.com/ecae>
- pin: 27dd54ab