

PHP程序猿应该知道的Nginx(上)

Nginx起手式

特别感谢:周晶老师<http://weibo.com/idevz>

Panda
知识的搬运工

Index

- Nginx与PHP
- Nginx命令行
- Nginx配置
- LNMP实战
- Nginx与Lua

PHP与Nginx

WAMP
XAMPP

一键安装包

LNMP
LOMP

Windows
Apache
MySQL
PHP

LNMP.ORG
oneinstack.com

PHP程序员发展之路

- 1. LNMP 专攻业务 => 技术经理
- 2. LNMP 研究更深的技术 => C

初识Nginx

L · N · M · P

NGINX

命令行参数

Usage: nginx [-?hvVtTq] [-s signal] [-c filename] [-p prefix] [-g directives]

Options:

-?, -h : this help
-v : show version and exit
-V : show version and configure options then exit
-t : test configuration and exit
-T : test configuration, dump it and exit
-q : suppress non-error messages during configuration testing
-s signal : send signal to a master process: stop, quit, reopen, reload
-p prefix : set prefix path (default: /usr/local/openresty-1.11.2.1/nginx/)
-c filename : set configuration file (default: conf/nginx.conf)
-g directives : set global directives out of configuration file

-p

-V

-T

-g

命令行参数 (含义及用法)

-V

```
→ ~ /usr/local/openresty-1.11.2.1/nginx/sbin/nginx -V
nginx version: openresty/1.11.2.1
built by gcc 4.8.5 20150623 (Red Hat 4.8.5-4) (GCC)
built with OpenSSL 1.0.2f 28 Jan 2016
TLS SNI support enabled
configure arguments: --prefix=/usr/local/openresty-1.11.2.1/nginx
inx-module-0.60 --add-module=../xss-nginx-module-0.05 --add-module
ession-nginx-module-0.06 --add-module=../srcache-nginx-module-0.31
-var-nginx-module-0.05 --add-module=../memc-nginx-module-0.17 --ac
ds-csv-nginx-module-0.07 --with-ld-opt=-Wl,-rpath,/usr/local/open
-1.11.2.1/../../pcre-8.38 --with-http_ssl_module
→ ~ █
```

```
→ ~ /usr/local/openresty-1.11.2.1/nginx/sbin/nginx -v
nginx version: openresty/1.11.2.1
→ ~ █
```

命令行参数（含义及用法）

-T DUMP

```
http {  
    include      mime.types;
```

include 展开，按节追加

```
    #}  
    include vhost/*.conf;  
}  
  
# configuration file /usr/local/openresty-1.11.2.1/nginx/conf/mime.types:  
  
types {  
    text/html          html htm shtml;  
    text/css           css;  
    text/xml           xml;  
    text/xsl           xsl;  
    text/xspf+xml     xspf;  
    text/yaml          yaml;  
    text/x-yaml        x-yaml;  
    text/x-perl        perl;  
    text/x-javascript javascript;  
    text/x-jquery      jquery;  
    text/x-toml        toml;  
}
```

```
→ nginx sudo /usr/local/openresty-1.11.2.1/nginx/sbin/nginx -t  
nginx: the configuration file /usr/local/openresty-1.11.2.1/nginx/conf/nginx.conf syntax is ok  
nginx: configuration file /usr/local/openresty-1.11.2.1/nginx/conf/nginx.conf test is successful  
→ nginx █
```


命令行参数 (含义及用法)

-p

```

→ nginx /usr/local/openresty-1.11.2.1-nodebug/nginx/sbin/nginx -h
nginx version: openresty/1.11.2.1
Usage: nginx [-?hvVtTq] [-s signal] [-c filename] [-p prefix] [-g directives]

Options:
  -?, -h      : this help
  -v          : show version and exit
  -V          : show version and configure options then exit
  -t          : test configuration and exit
  -T          : test configuration, dump it and exit
  -q          : suppress non-error messages during configuration testing
  -s signal   : send signal to a master process: stop, quit, reopen, reload
  -p prefix   : set prefix path (default: /usr/local/openresty-1.11.2.1-nodebug/nginx/)
  -c filename : set configuration file (default: conf/nginx.conf)
  -g directives : set global directives out of configuration file

→ nginx /usr/local/openresty-1.11.2.1-nodebug/nginx/sbin/nginx -h
nginx version: openresty/1.11.2.1
Usage: nginx [-?hvVtTq] [-s signal] [-c filename] [-p prefix] [-g directives]

Options:
  -?, -h      : this help
  -v          : show version and exit
  -V          : show version and configure options then exit
  -t          : test configuration and exit
  -T          : test configuration, dump it and exit
  -q          : suppress non-error messages during configuration testing
  -s signal   : send signal to a master process: stop, quit, reopen, reload
  -p prefix   : set prefix path (default: /usr/local/openresty-1.11.2.1-nodebug/nginx/)
  -c filename : set configuration file (default: conf/nginx.conf)
  -g directives : set global directives out of configuration file
  
```

命令行参数 (含义及用法)

-p

```

→ nginx /usr/local/openresty-1.11.2.1-nodebug/nginx/sbin/nginx -h
nginx version: openresty/1.11.2.1
Usage: nginx [-?hvVtTq] [-s signal] [-c filename] [-p prefix] [-g directives]

```

Options:

- ?, -h : this help
- v : show version and exit
- V : show version and configure options then exit
- t : test configuration and exit
- T : test configuration, dump it and exit
- q : suppress non-error messages during configuration testing
- s signal : send signal to a master process: stop, quit, reopen, reload
- p prefix : set prefix path (default: /usr/local/openresty-1.11.2.1-nodebug/nginx/)
- c filename : set configuration file (default: conf/nginx.conf)
- g directives : set global directives out of configuration file

```

→ nginx /usr/local/openresty-1.11.2.1-nodebug/nginx/sbin/nginx -h
nginx version: openresty/1.11.2.1
Usage: nginx [-?hvVtTq] [-s signal] [-c filename] [-p prefix] [-g directives]

```

Options:

- ?, -h : this help
- v : show version and exit
- V : show version and configure options then exit
- t : test configuration and exit
- T : test configuration, dump it and exit
- q : suppress non-error messages during configuration testing
- s signal : send signal to a master process: stop, quit, reopen, reload
- p prefix : set prefix path (default: /usr/local/openresty-1.11.2.1-nodebug/nginx/)
- c filename : set configuration file (default: conf/nginx.conf)
- g directives : set global directives out of configuration file

logs

nginx.pid

...

命令行参数 (含义及用法)

-g 全局指令 (Directives) 设置【来自ngx的核心、基础工具等模块】

```
→ stuq sudo /usr/local/openresty-1.11.2.1/nginx/sbin/nginx -T | grep pid
nginx: the configuration file /usr/local/openresty-1.11.2.1/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/openresty-1.11.2.1/nginx/conf/nginx.conf test is successful
#pid          logs/nginx.pid;
→ stuq sudo /usr/local/openresty-1.11.2.1/nginx/sbin/nginx
→ stuq ll /usr/local/openresty-1.11.2.1/nginx/logs/nginx.pid
-rw-r--r-- 1 root root 5 9月 28 19:09 /usr/local/openresty-1.11.2.1/nginx/logs/nginx.pid
→ stuq sudo /usr/local/openresty-1.11.2.1/nginx/sbin/nginx -s stop
→ stuq sudo /usr/local/openresty-1.11.2.1/nginx/sbin/nginx -g "pid ./nginx.pid;"
→ stuq ll /usr/local/openresty-1.11.2.1/nginx/logs/nginx.pid
ls: 无法访问 /usr/local/openresty-1.11.2.1/nginx/logs/nginx.pid: 没有那个文件或目录
→ stuq ll /usr/local/openresty-1.11.2.1/nginx/nginx.pid
-rw-r--r-- 1 root root 5 9月 28 19:09 /usr/local/openresty-1.11.2.1/nginx/nginx.pid
```

nginx.conf

配置语法

Syntax: **user** *user* [*group*];
 Default: user nobody nobody;
 Context: main

Syntax: **set** *\$variable* *value*;
 Default: —
 Context: server, location, if

变量群:arg_*, cookie_*

内建变量:ngx_http_core_module / \$scheme(http / https)

用户变量: set \$a a_value

变量、配置指令、作用域 / 作用阶段

段落 / 节

细节请参阅nginx 官方 wiki

```
user nobody;
worker_processes 1;
pid logs/nginx.pid;
events {
    worker_connections 1024;
}
http {
    server {
        location / {
        }
    }
}
```

配置语法

```
user nobody;
worker_processes 1;

error_log logs/error.log;
error_log logs/error.log notice;
error_log logs/error.log info;

pid logs/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log logs/access.log main;

    sendfile on;
    keepalive_timeout 65;

    gzip on;
    server {
        listen 8000;
        listen somename:8080;
        server_name somename alias another.alias;

        location / {
            root html;
            index index.html index.htm;
        }
    }
    include vhost/*.conf;
}
```

```
location ~ /\.php$ {
    set $name idevz;
    fastcgi_pass localhost:9000;
    fastcgi_param SCRIPT_FILENAME
        $document_root$fastcgi_script_name;
    fastcgi_param ZHOU1 $fastcgi_path_info;
    fastcgi_param ZHOU2 $document_uri;
    fastcgi_param ZHOU3 $name;
    include fastcgi_params;
}
```

配置语法

变量原则：与请求绑定

推荐春哥的新浪博客：<http://blog.sina.com.cn/openresty>

配置语法

调试方法: `ngx_echo`、`ngx_lua`

推荐使用 `ngx_lua` 方便, 快捷

Debug 日志

编译选项

`--with-debug`

开启 `debug` 相关选项支持

一般操作

`error_log /path/to/log debug;`

循环内存缓冲区

`error_log memory:32m debug;`

Debug链接设置

`debug_connection 192.168.1.1;`

Debug 日志

编译选项

--with-debug

一般操作

error_log /path/to/log debug;

log level

循环内存缓冲区

error_log memory:32m debug;

Debug链接设置

debug_connection 192.168.1.1;

Debug 日志

编译选项

--with-debug

一般操作

error_log /path/to/log debug;

stderr

内存中编写调试日志

循环内存缓冲区

error_log memory:32m debug;

memory:

Debug链接设置

debug_connection 192.168.1.1;

syslog:

Debug 日志

编译选项	<code>--with-debug</code>
一般操作	<code>error_log /path/to/log debug;</code>
循环内存缓冲区	<code>error_log memory:32m debug;</code>
Debug链接设置	<code>debug_connection 192.168.1.1;</code>

指定允许记录Debug信息的 client

Nginx设计结构

分层、别类的模块化设计

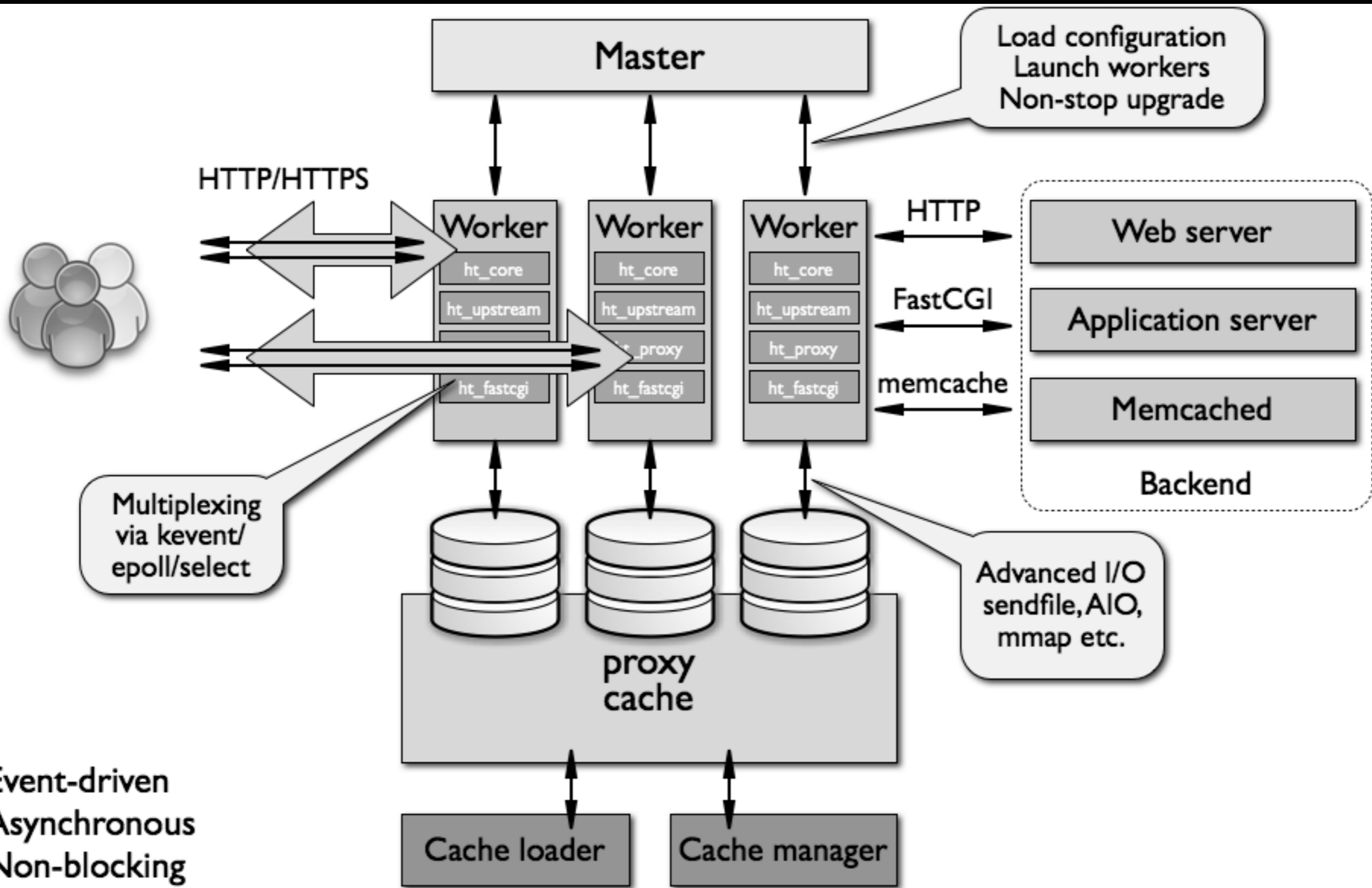
(五大类型：核心、配置、事件、HTTP、Mail)

事件驱动、请求异步处理

(处理过程分阶段，由事件收集、分发器触发)

Master、Worker进程设计

(发挥多核优势)

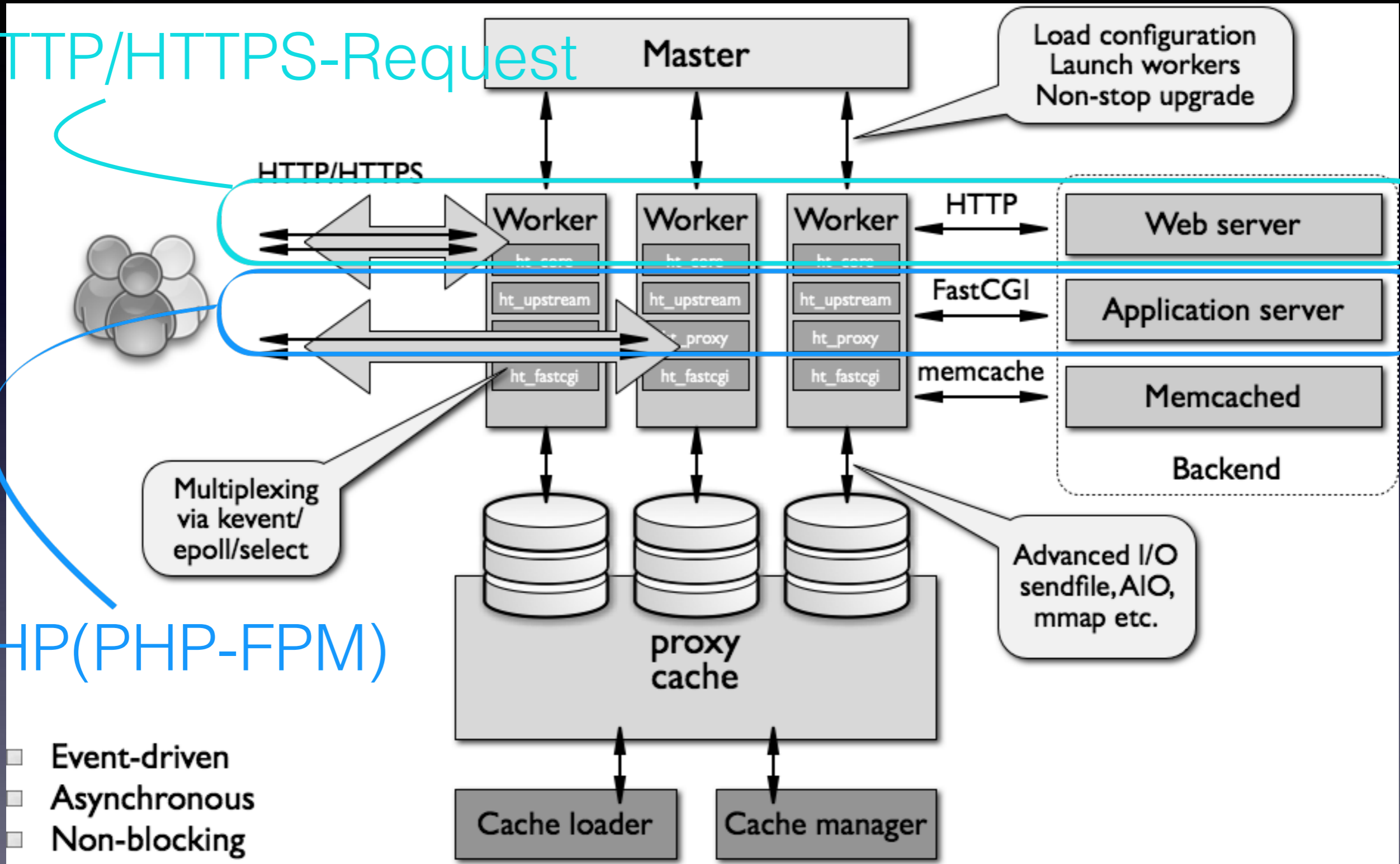


- Event-driven
- Asynchronous
- Non-blocking

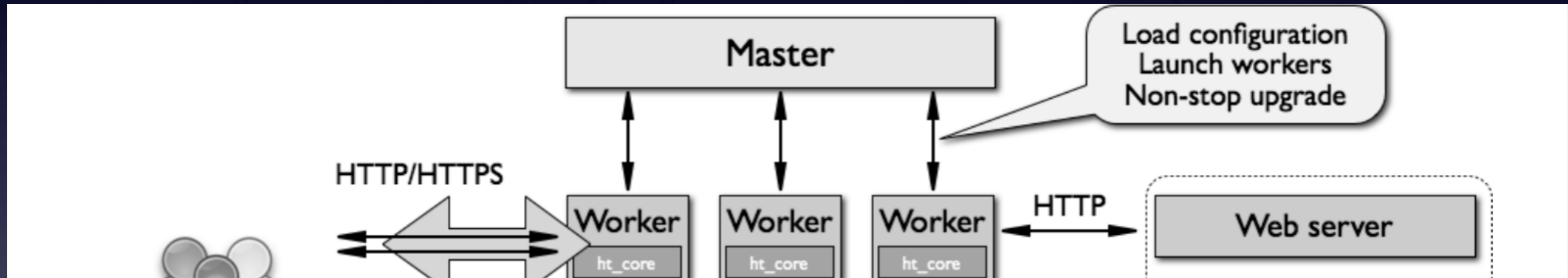
站在PHP业务开发的角
度看Nginx

HTTP/HTTPS-Request

PHP(PHP-FPM)



HTTP/HTTPS-Request



LNMP实践



LNMP 实践

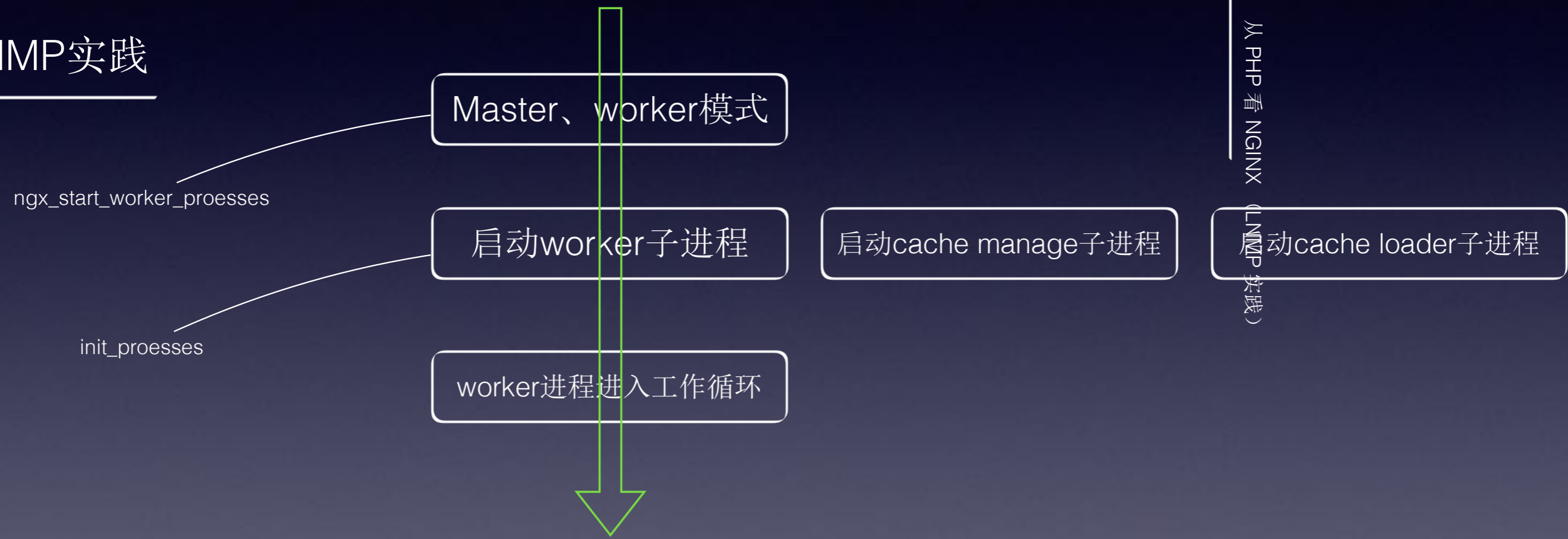
Master、worker 模式

启动 worker 子进程

启动 cache manage 子进程

启动 cache loader 子进程

LNMP实践



LNMP 实践

Master 仅负责：相当于你的BOSS

配置初始化

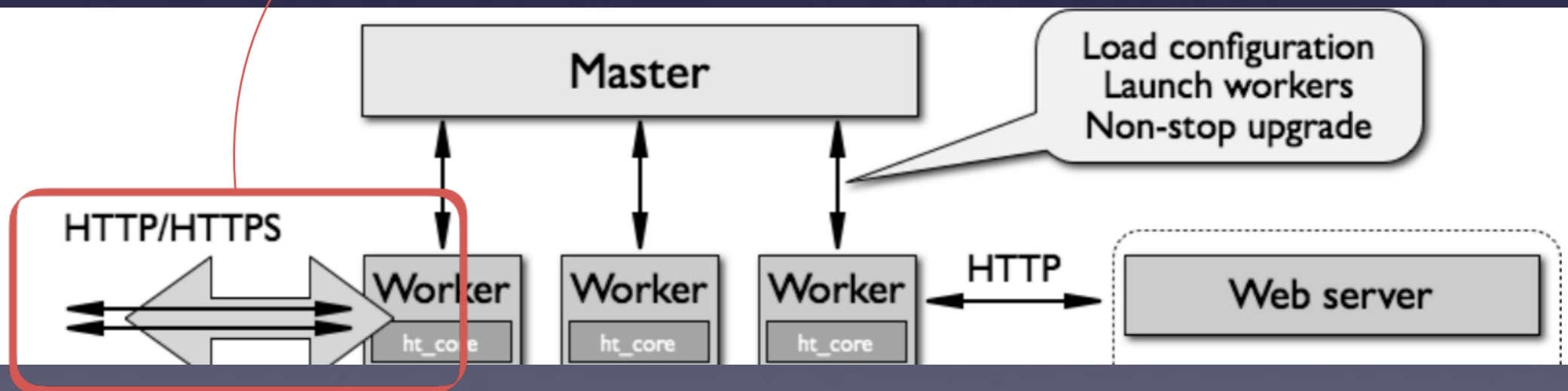
子进程管理

平滑升级、重启 等等...

就是不负责请求转发

`ngx_event_accept`

监听端口上读事件的处理方法



那具体 NGINX 又是如何处理请求的呢？

LNMP 实践

根据请求头中 Host 字段确定提供服务的 Server 段

不匹配则调用此端口的 default_server

如果 request 头中 Host 字段为空

```
server {  
    listen 80;  
    server_name "";  
    return 444;  
}
```

基于 uri 检测最具体的前缀 location 段

前缀 location 段

```
location / {  
}  
location /a {  
}
```

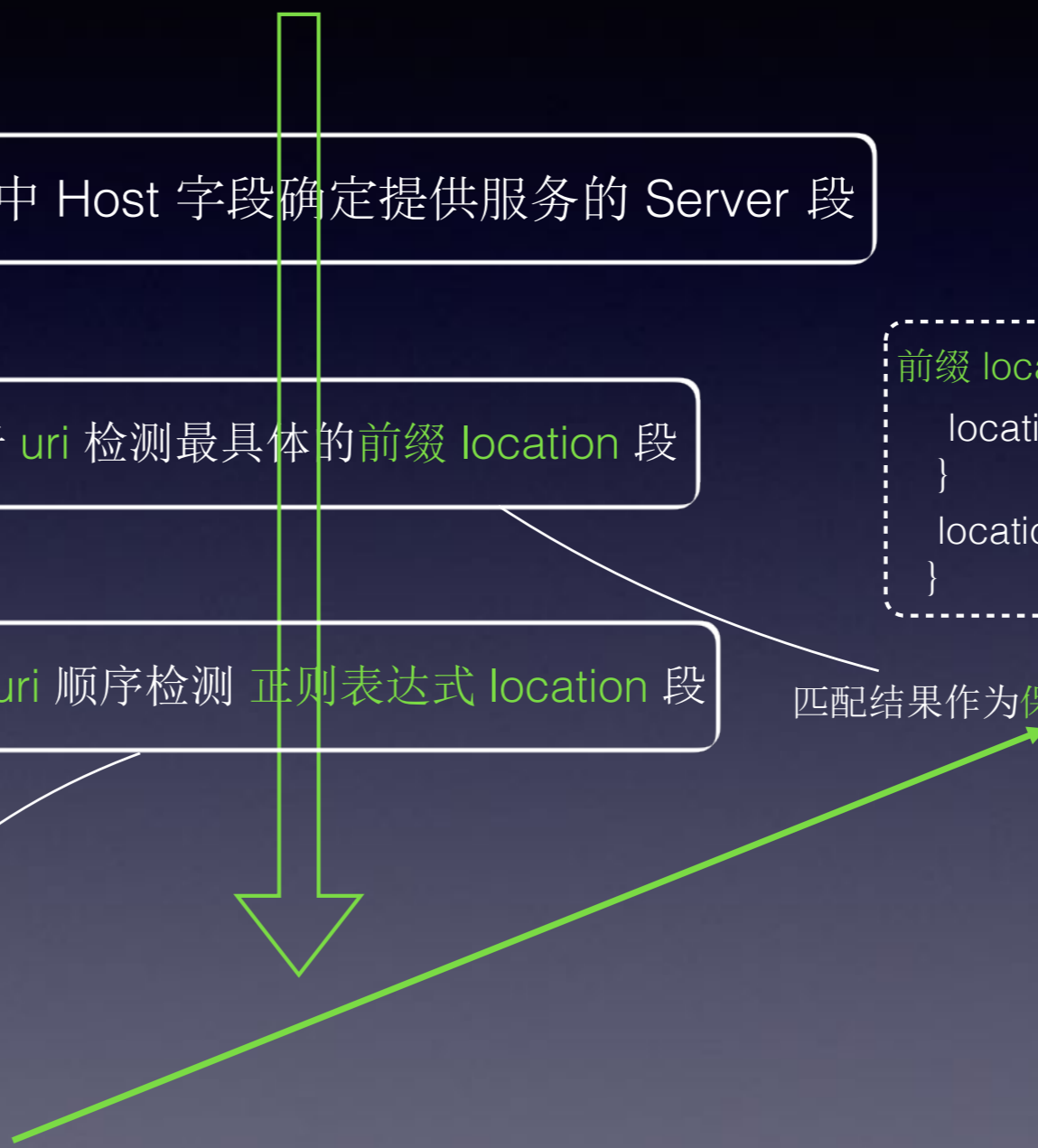
基于 uri 顺序检测 正则表达式 location 段

匹配结果作为保底处理手段

正则表达式 location 段

```
location ~* \.(gif|jpg|png)$ {  
}  
location ~ \.php$ {  
}
```

匹配成功则直接处理，否则



LNMP 实践

```
location / {
    index    index.html index.php;
}

location /a {
    index    a.html;
}

location ~* \.(gif|jpg|png)$ {
    expires 30d;
}

location ~ \.php$ {
    fastcgi_pass    localhost:9000;
    fastcgi_param   SCRIPT_FILENAME
                    $document_root$fastcgi_script_name;
    include         fastcgi_params;
}

location = /favicon.ico {
    access_log    off;
    root /media/psf/code/www/vanilla/stuq/pub/;
    expires      max;
}
```

```
curl http://z.shared/a.gifx?a=b
```

```
test location: "/"
test location: "favicon.ico"
test location: "a"
test location: ~ "\.(gif|jpg|png)$"
test location: ~ "\.php$"
using configuration "/a"
```

```
curl http://z.shared/x.gifx?a=b
```

```
test location: "/"
test location: "favicon.ico"
test location: ~ "\.(gif|jpg|png)$"
test location: ~ "\.php$"
using configuration "/"
```

LNMP 实践

PHP-FPM (FastCGI Process Manager)

FastCGI (Common Gateway Interface)

`ngx_http_fastcgi_module`

开启debug

LNMP实践

```
location / {
    index    index.html index.php;
}

location /a {
    index    a.html;
}

location ~* \.(gif|jpg|png)$ {
    expires 30d;
}

location ~ \.php$ {
    fastcgi_pass    localhost:9000;
    fastcgi_param   SCRIPT_FILENAME
                    $document_root$fastcgi_script_name;
    include         fastcgi_params;
}

location = /favicon.ico {
    access_log off;
    root /media/psf/code/www/vanilla/stuq/pub/;
    expires        max;
}
```

```
curl http://z.shared/a.php?a=b
```

```
test location: "/"
test location: "favicon.ico"
test location: "a"
test location: ~ "\.(gif|jpg|png)$"
test location: ~ "\.php$"
using configuration "\.php$"
```

LNMP 实践

curl http://z.shared/a.php?a=b

```
location ~ /\.php$ {
    fastcgi_pass localhost:9000;
    fastcgi_param SCRIPT_FILENAME
        $document_root$fastcgi_script_name;
    fastcgi_param ZHOU1 $fastcgi_path_info;
    fastcgi_param ZHOU2 $document_uri;
    include fastcgi_params;
}
```

```
[REQUEST_METHOD] => GET
[QUERY_STRING] => a=b
[ZHOU2] => /a.php
[ZHOU1] =>
[SCRIPT_FILENAME] => /media/psf/code/www/vanilla/stuq/a.php
```

```
test location: "/"
test location: "favicon.ico"
test location: "a"
test location: ~ "\.(gif|jpg|png)$"
test location: ~ /\.php$"
using configuration /\.php$"
```

```
post access phase: 10
http init upstream, client timer: 0
epoll add event: fd:3 op:3 ev:80002005
http script copy: "SCRIPT_FILENAME"
```

```
get rr peer, try: 2
get rr peer, current: 00007FD86DC6B9B0 0
stream socket 12
epoll add connection: fd:12 ev:80002005
connect to 127.0.0.1:9000, fd:12 #9
http upstream connect: -2
```

```
http run request: "/a.php?a=b"
http upstream check client, write event:1, "/a.php"
http upstream request: "/a.php?a=b"
http upstream send request handler
http upstream send request
http upstream send request body
```

```
write new buf t:1 f:0 00007FD86DC36C08
http write filter: l:0 f:0 s:225
http cacheable: 0
http upstream process upstream
pipe read upstream: 1
pipe preread: 988
readv: eof:1, avail:0
readv: 1, last:3032
pipe recv chain: 0
```

这才刚刚开始

从 LNMP 到 LORMP

OpenResty做了什么

L · OR · M · P

OpenResty

OpenResty = Nginx + ngx_http_lua_module
+ lua_resty*

OpenResty 为 PHP Web 服务插上翅 膀

你只需要一次平滑重启

即可从Nginx升级到
OpenResty

More

- Openresty <https://github.com/openresty/lua-nginx-module>
- ngx_php https://github.com/rryqszq4/nginx_php
- ngx_python https://github.com/rryqszq4/nginx_python
- ngx_ruby https://github.com/rryqszq4/nginx_ruby
- more

QA
Thanks