nginxの特徴

1.スピード
2.使いやすさ
3.モジュール性

nginxの特徴

1.スピード

-非同期ソケット
①リクエストがある度にプロセスを作成せず、
ひとつのプロセスで処理を行う。
②CPUの負荷とメモリ使用量を
おさえることができる。

nginxの特徴

2.使いやすさ

設定ファイル
①読みやすく、操作しやすい。

```
#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;
    #tcp_nopush     on;

    #keepalive_timeout  0;
    keepalive_timeout  65;

    #gzip  on;

    server {
        listen       80;
        server_name  localhost;

        #charset koi8-r;

        #access_log  logs/host.access.log  main;

        location / {
            root   html;
            index  index.html index.htm;
        }

        #error_page  404              /404.html;

        # redirect server error pages to the static page /50x.html
```

nginxの特徴

3.モジュール性

プラグインシステム
①必要に応じて、必要な機能を「モジュール」として組み込むことで、自由に拡張することができる。

nginxのプロセスアーキテクチャ

1.nginx起動時にはメモリ内には一つのプロセスが作られる → マスタプロセス

2.マスタプロセスはクライアントからの要求を処理せず、処理を行うプロセスを起動する。
→ ワーカプロセス

nginxとapacheの比較
(リクエストの処理方法)

nginx
非同期ソケットでリクエストを処理する。
別々のスレッドで処理は行わない。
→ メモリとCPUのオーバヘッドを軽減する。

Apache
同期ソケットでリクエストを処理する。
別々のスレッドまたはプロセスで
リクエストを処理する
→ リクエストが増える度に、メモリとCPUの
オーバーヘッドに繋がる。

nginxとapacheの比較
(プログラミング言語)

nginx
C言語

Apache
C言語。モジュール類はC++

nginxとapacheの比較
(対応OS)

nginx
Windows, GNU/Linux, Unix, BSD, Mac OS X, Solaris

Apache
Windows, GNU/Linux, Unix, BSD, Mac OS X, Solaris, Novell NetWare, OS/2, PTF, OPEN VMS, eCS, AIX, z/OS, HP-UX

nginxとapacheの比較
(誕生年)

nginx
2002年

Apache
1994年

nginxとapacheの比較 (その他)

nginx
仮想ホスト - サポートしている。
CGI – FastCGIのみ
モジュールシステム – 静的モジュールシステム

Apache
仮想ホスト – サポートしている。ディレクトリ毎に.htacessを置ける(個別に設定できる)
CGI – FastCGI, CGI共に対応している
モジュールシステム - 動的モジュールシステム

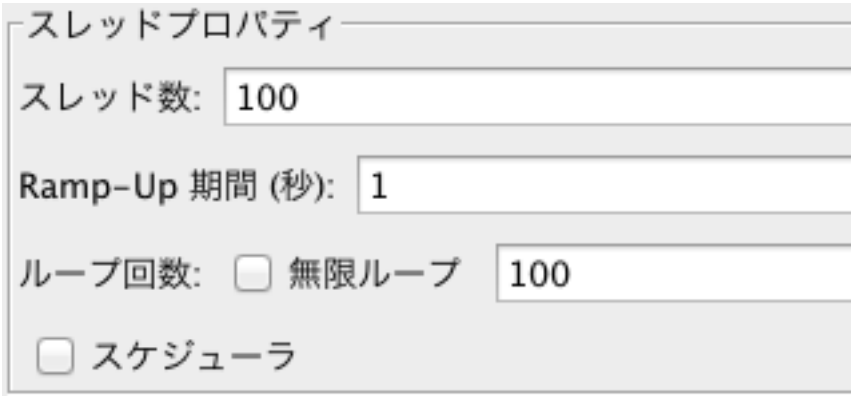
nginxとapacheの比較
(パフォーマンス)

RPS (Request Per Second)
nginxはapacheの約2倍

応答時間
nginxの方が短い。リクエストが増えるほど、
apacheは処理速度が遅くなる。

nginxとapacheの比較
(ベンチマーク apache)

100コネクション、100リクエスト




```
<IfModule prefork.c>
StartServers       1
MinSpareServers    1
MaxSpareServers    1
ServerLimit      100
MaxClients       100
MaxRequestsPerChild  100
</IfModule>
```

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	10000	173	123	165	3	5987	0.00%	527.0/sec	214.6
合計	10000	173	123	165	3	5987	0.00%	527.0/sec	214.6

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	20000	163	119	161	3	5987	0.00%	405.9/sec	165.3
合計	20000	163	119	161	3	5987	0.00%	405.9/sec	165.3

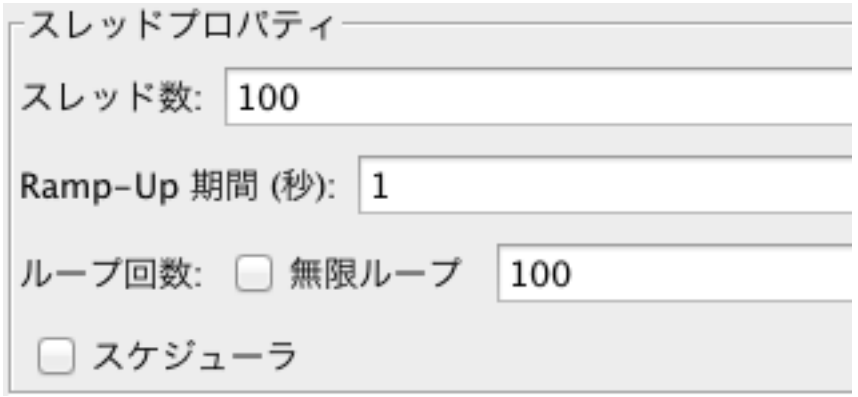
Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	30000	177	114	175	3	5987	0.01%	315.8/sec	128.6
合計	30000	177	114	175	3	5987	0.01%	315.8/sec	128.6

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	40000	193	118	253	3	7610	0.00%	216.2/sec	88.0
合計	40000	193	118	253	3	7610	0.00%	216.2/sec	88.0

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	50000	182	114	227	3	7610	0.01%	236.0/sec	96.1
合計	50000	182	114	227	3	7610	0.01%	236.0/sec	96.1

nginxとapacheの比較
(ベンチマーク nginx)

100コネクション、100リクエスト




```
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  100;
}
```

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	10000	113	124	183	2	1368	0.00%	785.6/sec	281.6
合計	10000	113	124	183	2	1368	0.00%	785.6/sec	281.6

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	20000	115	125	186	1	1368	0.00%	204.9/sec	73.4
合計	20000	115	125	186	1	1368	0.00%	204.9/sec	73.4

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	30000	117	127	188	1	1459	0.01%	250.9/sec	89.9
合計	30000	117	127	188	1	1459	0.01%	250.9/sec	89.9

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	40000	117	127	186	1	1459	0.01%	285.9/sec	102.5
合計	40000	117	127	186	1	1459	0.01%	285.9/sec	102.5

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Throughput	KB/sec
HTTP リク...	50000	117	126	185	1	1459	0.01%	311.3/sec	111.6
合計	50000	117	126	185	1	1459	0.01%	311.3/sec	111.6

nginxとapacheの比較
(証言①)

-apacheは100ものオプションがあるが、必要なオプションは6個だけだ。
nginxはその6個のことだけを行う。そのうち5つは、apacheよりも50倍高速だ。

By Chris Lea (Media Temple engineer)

http://maisonbisson.com/blog/post/12249/chris-lea-on-nginx-and-wordpress/

nginxとapacheの比較 (証言②)

-1日数千万を超えるHTTP要求をnginxのリバースプロキシで捌いているが、使っているサーバは1台だけだ。約15MBのRAMとCPUの処理能力の10%ほどを使用している。同じ負荷がかかれば、apacheは1000個くらいのプロセスを作り、RAMに至ってはどれだけ使ったかわからないまま落ちるだろう。スレッドスタック全体で400MB+ものRAMを使って。

By Bob Ippolito (Mochi Media engineer)
http://highscalability.com/product-nginx

結論①

apacheは、処理速度を犠牲にして機能を重視している構成となっている。apacheはリクエストの度にメモリにモジュール、その他コンポーネントをロードするので、リクエストが増えるとサーバにかかる負担は大きくなる。
nginxは、軽量で安定しており大量のリクエストがきても処理できる。apacheと比べて消費するRAM容量やCPU待ち時間も少ない。

結論②

その代わり、apacheは動的コンテンツを処理するのに向いている。
nginxは静的ファイルを処理するのに向いている。
両者の特性を活かして、nginxをリバースプロキシとして使用する事例が多い。

結論③

nginxをリバースプロキシとして使用し、バックエンドサーバはその他のサーバを使用する、という構成が多いようだ。今後もその傾向は強まるのではないか。