

mod_backhand

use your resources

INTERNALS EXPLAINED

THEO SCHLOSSNAGLE

<jesus@cnds.jhu.edu>

April 5th, 2001

WHAT IS MOD_BACKHAND?

- AN APACHE MODULE THAT:
 - PROVIDES FACILITIES TO PROXY HTTP
 - PROVIDES FACILITIES TO REDIRECT HTTP
 - ANNOUNCES LOCAL RESOURCE INFORMATION
 - COLLECTS CLUSTER-WIDE RESOURCE INFORMATION
 - ALLOWS FLEXIBLE DECISIONS ON WHERE AND HOW TO REALLOCATE REQUESTS

IMPLEMENTATION OVERVIEW (1/3)

- A SEPARATE MODERATOR PROCESS THAT:
 - COLLECTS AND DISTRIBUTES RESOURCE INFORMATION
 - PASSES ACTIVE CONNECTIONS TO CHILDREN
 - RECEIVES CONNECTIONS FROM CHILDREN
 - MANAGES CONNECTION POOLING

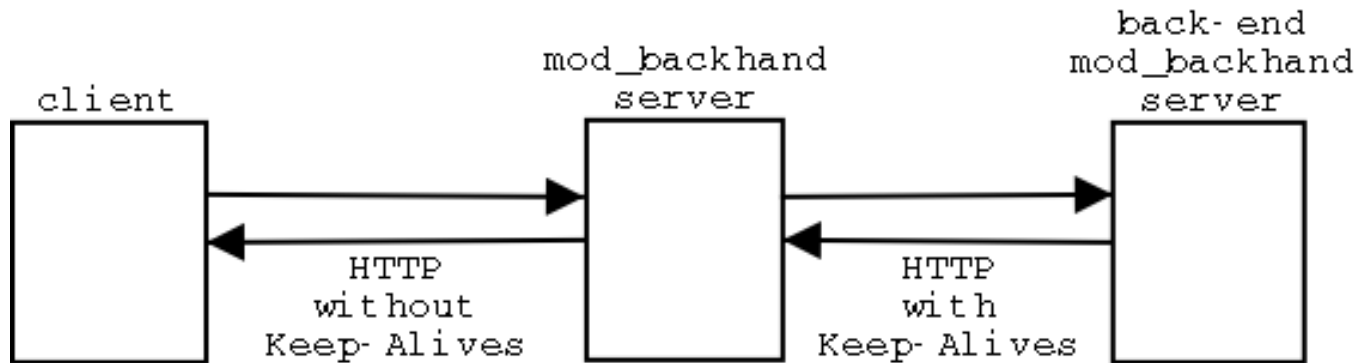
IMPLEMENTATION OVERVIEW (2/3)

- DECISION MAKING:
 - EXTENSIVE RESOURCE INFORMATION
 - POWERFUL BUILT-IN CANDIDACY FUNCTIONS
 - ARCHITECTURE FOR DYNAMICALLY-LOADED FUNCTIONS
 - SIMPLE METHOD TO CHOOSE PROXYING/REDIRECTION

IMPLEMENTATION OVERVIEW (3/3)

- PROXYING:

- AUTO-“MAGIC” UPGRADE OF HTTP CONNECTIONS



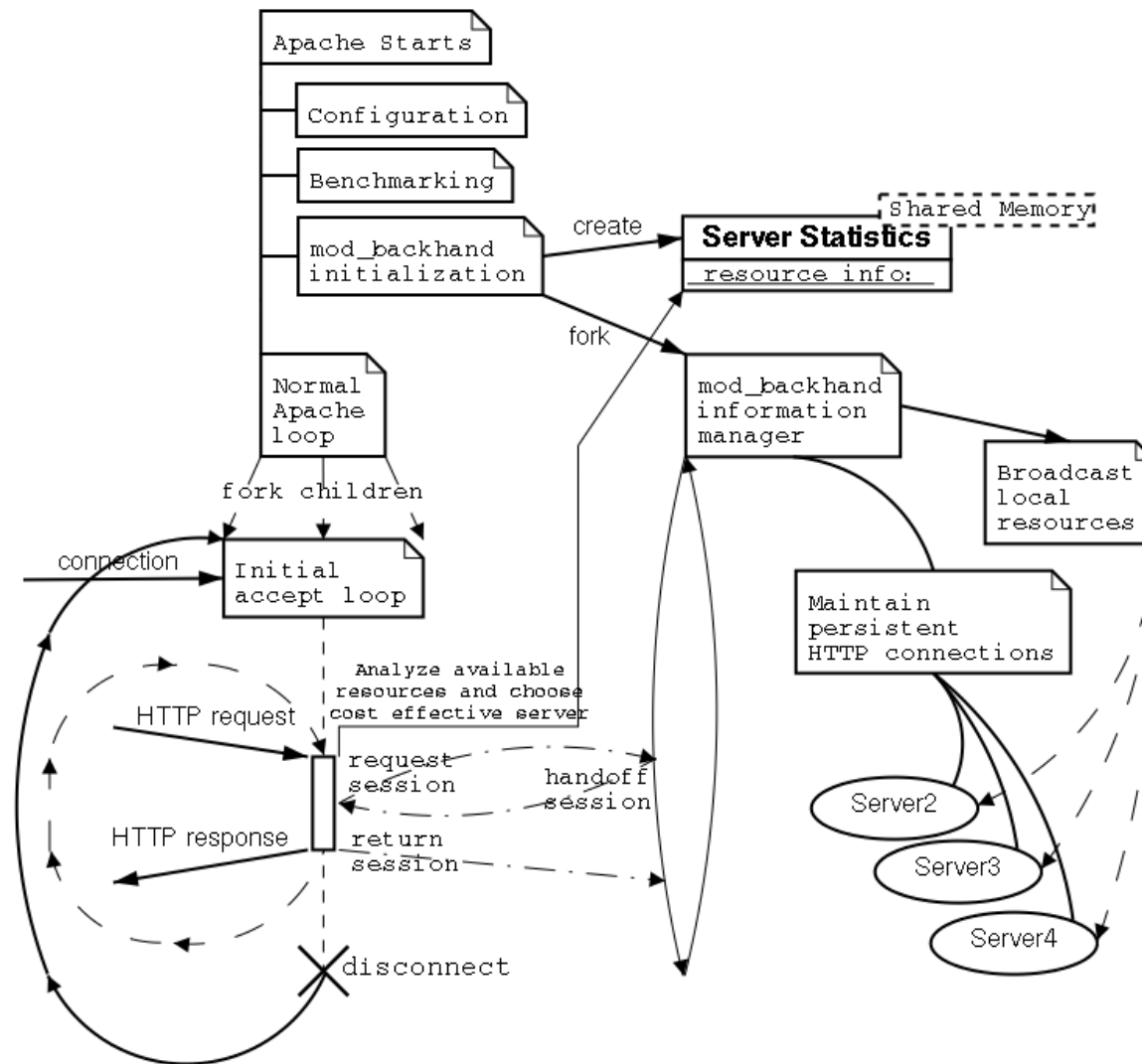
- CONNECTION POOLING MAKES THIS VIABLE

- KEEP-ALIVES ARE DANGEROUS!

(WITHOUT THE PARENTAL ACCEPT PATCH)

<http://www.omniti.com/~jesus/projects/>

IMPLEMENTATION OVERVIEW DIAGRAM



MAKING DECISIONS – BUILT-INS (1/2)

- **BUILT-INS: Backhand *function* [arg]**
 - **off** – CHOOSE NO SERVERS
 - **addSelf** – ADD THE LOCAL MACHINE
 - **removeSelf** – REMOVE THE LOCAL MACHINE
 - **byAge [timeout]** – REMOVE “OLD” MACHINES
 - **byLoad [bias]** – SORT BY SYSTEM LOAD
 - **byBusyChildren [bias]** –
SORT BY BUSY APACHE CHILDREN
 - **byCPU** – CHOOSE MACHINE WITH THE HIGHEST CPU IDLE
 - **byLogWindow** – KEEP ONLY THE FIRST LOG_2 MACHINES
 - **byRandom** – RANDOMIZE THE LIST
 - **byCost** – SORT BY “COSTLINESS”

MAKING DECISIONS – BUILT-INS (2/2)

- BUILT-INS: **Backhand *function* [arg]**
 - **bySession [identifier]** – STICKY SESSIONS
 - * LOOK FOR A COOKIE NAMED `identifier`
 - * IF NOT FOUND, RETURN THE LIST UNAUGMENTED
 - * DECODE ASSUMING IT IS A HEX-ENCODED IPV4 ADDRESS
 - * ATTEMPT TO FIND A SERVER WITH THAT MATCHES THIS IP
 - * IF FOUND, SELECT THAT SERVER
 - * IF NOT FOUND, RETURN THE LIST UNAUGMENTED

MAKING DECISIONS – CUSTOM

- **BackhandFromSO *filename function [arg]***
 - Dynamically load `filename` and execute `function` with the optional argument `arg`.
 - `function` should have the prototype:

```
int function(request_rec *r,  
            ServerSlot *candidates,  
            int *numcandidates,  
            char *arg);
```

MAKING DECISIONS – INPUTS

ServerSlot

int id	MACHINE ID IN THE SERVERSTATS TABLE
redirect	MB_HTTP_{PROXY,REDIRECT}
hosttype	MB_HOSTTYPE_{NAME,IP}

serverstat

char hostname[40]	MACHINE NAME
time_t mtime	LAST MODIFICATION TIME
int arriba	CPU BENCHMARK
int aservers	AVAILABLE APACHE CHILDREN
int nservers	TOTAL APACHE CHILDREN
int load	SYSTEM LOAD (TIMES 1000)
int cpu	PERCENT CPU IDLE (TIMES 1000)
int tmem	TOTAL MEMORY
int amem	AVAILABLE MEMORY

EXAMPLE FUNCTION

```
int byRandom(request_rec *r,
             ServerSlot *c, int *n, char *a) {
    int i;
    ServerSlot swap;
    static int rs=-1;
    if(rs==0) { srand(time(NULL)); rs=1; }
    for(i=0;i<*n;i++) {
        rs = rand()%(*n-i);
        swap = c[i];
        c[i] = c[i+rs];
        c[i+rs] = swap;
    }
    return *n;
}
```

EXAMPLE FUNCTION (REDIRECTION)

```
int byRandom(request_rec *r,
             ServerSlot *c, int *n, char *a) {
    int i;
    ServerSlot swap;
    static int rs=-1;
    if(rs==0) { srand(time(NULL)); rs=1; }
    for(i=0;i<*n;i++) {
        rs = rand()%(*n-i);
        swap = c[i];
        c[i] = c[i+rs];
        c[i+rs] = swap;
        c[i].redirect = MB_HTTP_REDIRECT;
        c[i].hosttype = MB_HOSTTYPE_NAME;
    }
    return *n;
}
```

CONFIGURING A TYPICAL CLUSTER

- **ENABLE THE MODULE**

```
LoadModule backhand_module libexec/mod_backhand.so
AddModule mod_backhand.c
```

- **SET OUR GLOBAL SYSTEM PARAMETERS**

```
MulticastStats 224.2.3.4:4445,1
AcceptStats 192.168.10.0/24
UnixSocketDir /opt/apache/backhand
```

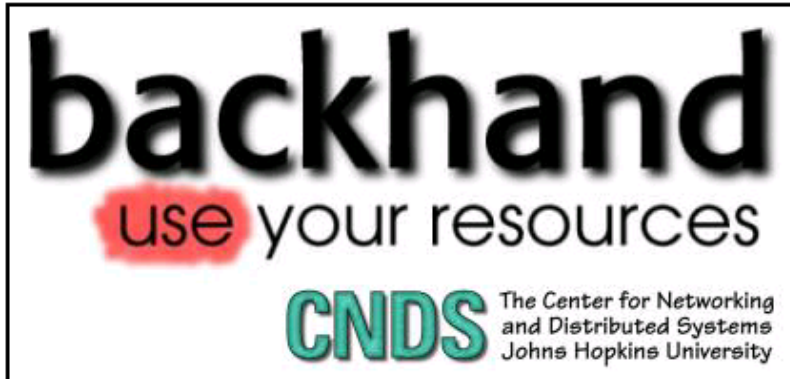
CONFIGURING A TYPICAL CLUSTER

- ENABLE THE RESOURCE STATUS PAGE

```
<Location /backhand/>  
  SetHandler backhand-handler  
</Location>
```

- NOW VISIT THE STATUS PAGE TO:
 - MAKE SURE THE CLUSTER IS VISIBLE
 - SEE HOW THE CLUSTER IS DOING

THE MOD_BACKHAND STATUS PAGE



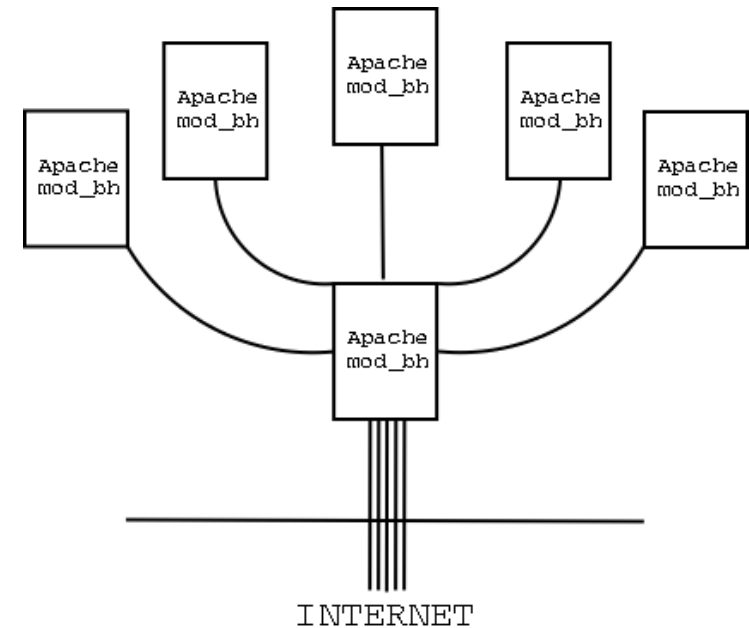
Local Machine Name: fog13.cnds.jhu.edu
Apache Version String:
Apache/1.3.12 (Unix) balanced_by_mod_backhand/1.1.1pre3
Server built: Feb 7 2001 17:00:33
REMOTE_ADDR: 216.0.51.145

Entry	Hostname	Age	Address	Total Mem	Avail Mem	# ready servers/ # total servers	~ms/req [#req]	Arriba	# CPUs	Load/HWM	CPU Idle
0	fog13.cnds.jhu.edu	0	128.220.221.113:80	263700480	161514688	150/150	0 [0]	859809	2	0.000000/256	0.996000
1	fog9.cnds.jhu.edu	0	128.220.221.109:80	263700480	166274240	148/150	0 [0]	858204	2	0.000000/256	0.996000
2	fog14.cnds.jhu.edu	0	128.220.221.114:80	263700480	159118528	150/150	0 [0]	858312	2	0.000000/256	0.996000
3	fog10.cnds.jhu.edu	0	128.220.221.110:80	263700480	165725376	150/150	0 [0]	858557	2	0.000000/256	0.996000
4	fog16.cnds.jhu.edu	0	128.220.221.116:80	261623808	193549504	150/150	0 [0]	859386	2	0.000000/256	0.996000
5	fog15.cnds.jhu.edu	0	128.220.221.115:80	263700480	168273088	150/150	0 [0]	854075	2	0.000000/256	0.996000
6	fog11.cnds.jhu.edu	0	128.220.221.111:80	263700480	172516544	150/150	0 [0]	859232	2	0.000000/256	0.996000

SINGLE-POINT CLUSTER

- BACKHAND DIRECTIVES

```
<Files ~ "(\\.asp|\\.php)$">  
  Backhand byAge  
  Backhand removeSelf  
  Backhand byRandom  
  Backhand byLogWindow  
  Backhand byLoad  
</Files>
```



- ADVANTAGES

- SIMPLE DESIGN
- NO COMPLICATED RESOURCE CONTENTION

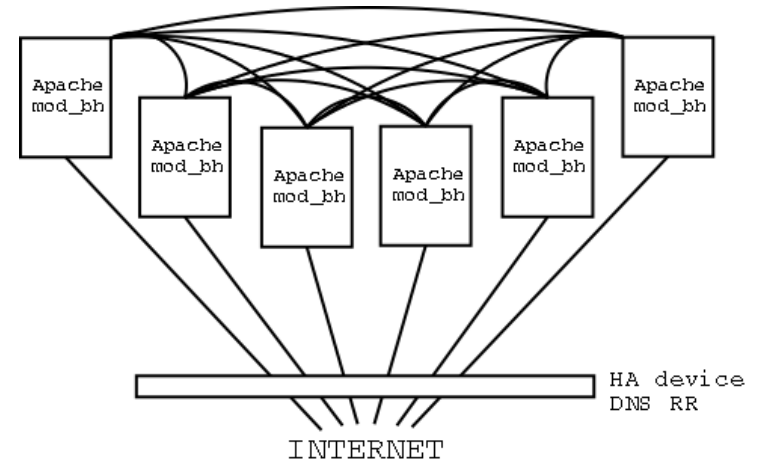
- DISADVANTAGES

- SINGLE POINT OF FAILURE
- OBVIOUS BOTTLENECK

MULTI-POINT CLUSTER

● BACKHAND DIRECTIVES

```
<Files ~ "(\\.asp|\\.php)$">  
  Backhand byAge  
  Backhand byRandom  
  Backhand byLogWindow  
  Backhand addSelf  
  Backhand byLoad 2.0  
</Files>
```



● ADVANTAGES

- MORE EGRESS POINTS
- NO SINGLE POINT OF FAILURE
- HIGHER PROBABILITY REQUESTS WILL NOT NEED TO BE REALLOCATED

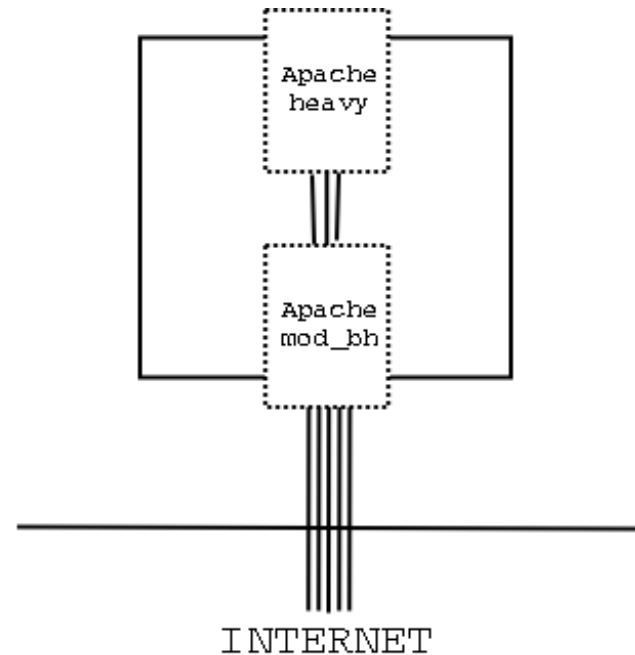
● DISADVANTAGES

- COMPLICATED CONTENTION ISSUES
- MORE RESOURCES USED FOR PROXYING
- HIGH-AVAILABILITY IS NOT PROVIDED

SIMPLE HTTP ACCELERATOR

- BACKHAND DIRECTIVES

```
MulticastStats 127.0.0.1 224.2.3.4:4445,1
BackhandSelfRedirect On
<Files ~ "(\\.asp|\\.php)$">
  Backhand byAge
</Files>
```



- ADVANTAGES

- CONNECTION POOLING!!!
- EASY STEPPING STONE TO A TWO-TIER MULTI-POINT CLUSTER

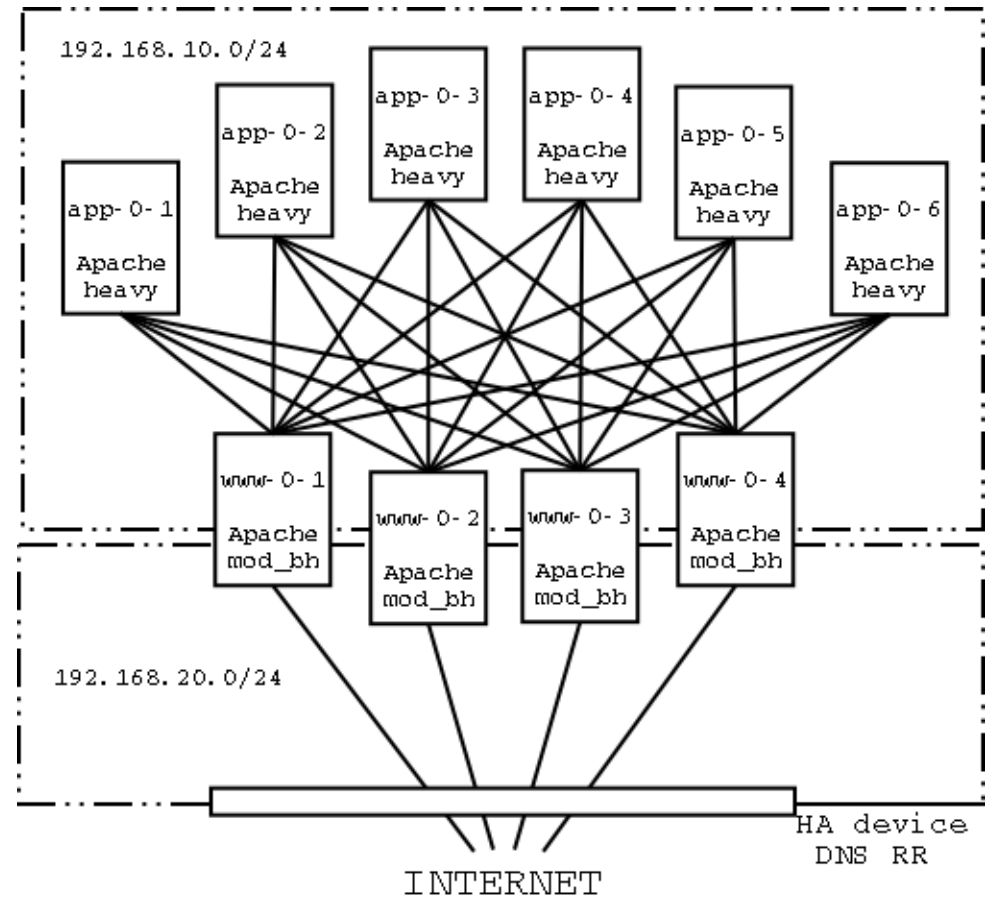
TWO-TIER MULTI-POINT CLUSTER

- **ADVANTAGES**

- MANY EGRESS POINTS
- NO SINGLE POINT OF FAILURE
- ACCELERATION
- SECOND TIER HIGH-AVAILABILITY
- REDUCED RESOURCE CONTENTION
- REUSED SESSIONS BETWEEN TIERS

- **DISADVANTAGES**

- MORE MACHINES
- HARD TO EFFICIENTLY ASSIGN MACHINES TO A TIER



TWO-TIER MULTI-POINT CONFIG (1/2)

- FIRST-TIER

```
UnixSocketDir /opt/apache/backhand
MulticastStats 192.168.10.255:4445
AcceptStats 192.168.10.0/24
<Location /backhand/>
    SetHandler backhand-handler
</Location>
<Files ~ "(\\.php)$">
    Backhand byAge
    BackhandFromSO libexec/byHostname.so byHostname app
    Backhand byRandom
    Backhand byLogWindow
    Backhand byBusyChildren
</Files>
```

TWO-TIER MULTI-POINT CONFIG (2/2)

- SECOND-TIER

```
UnixSocketDir /opt/apache/backhand
MulticastStats 192.168.10.255:4445
AcceptStats 192.168.10.0/24
<Location /backhand/>
    SetHandler backhand-handler
</Location>
```

SUMMARY

- mod_backhand is a powerful tool
- its various components can be used in several ways
- load-balancing is trivial to configure
- the built-in candidacy functions are generic and powerful
- powerful, customized candidacy functions can be loaded
- it can be used in combination with other LB/HA solutions
- the proxying is more efficient than mod_proxy's implementation (for reverse proxying)
- it is Open Source and actively maintained

CREDITS

- Johns Hopkins University
- The Center for Networking and Distributed Systems
(and all of its members)
- Yair Amir
- Sherry Schlossnagle
- Jonathan Stanton
- The mod_backhand user community