

Apache 3.0 (a tall tale)

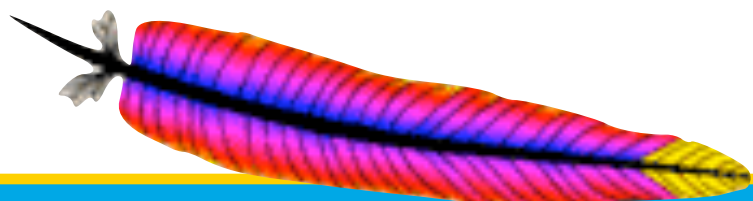
Roy T. Fielding, Ph.D.

Chief Scientist, Day Software

Cofounder, The Apache Software Foundation

Vice President, Apache HTTP Server

http://roy.gbiv.com/talks/200804_Apache3_ApacheCon.pdf





Vice President, Apache HTTP Server

NOT
APPROVED
APPROPRIATE
CONSENSUS

Apache 3.0

Roy T. Fielding, Ph.D.
Chief Scientist, Day Software
Cofounder, The Apache Software Foundation
Vice President, Apache HTTP Server

Outline

a bit of history

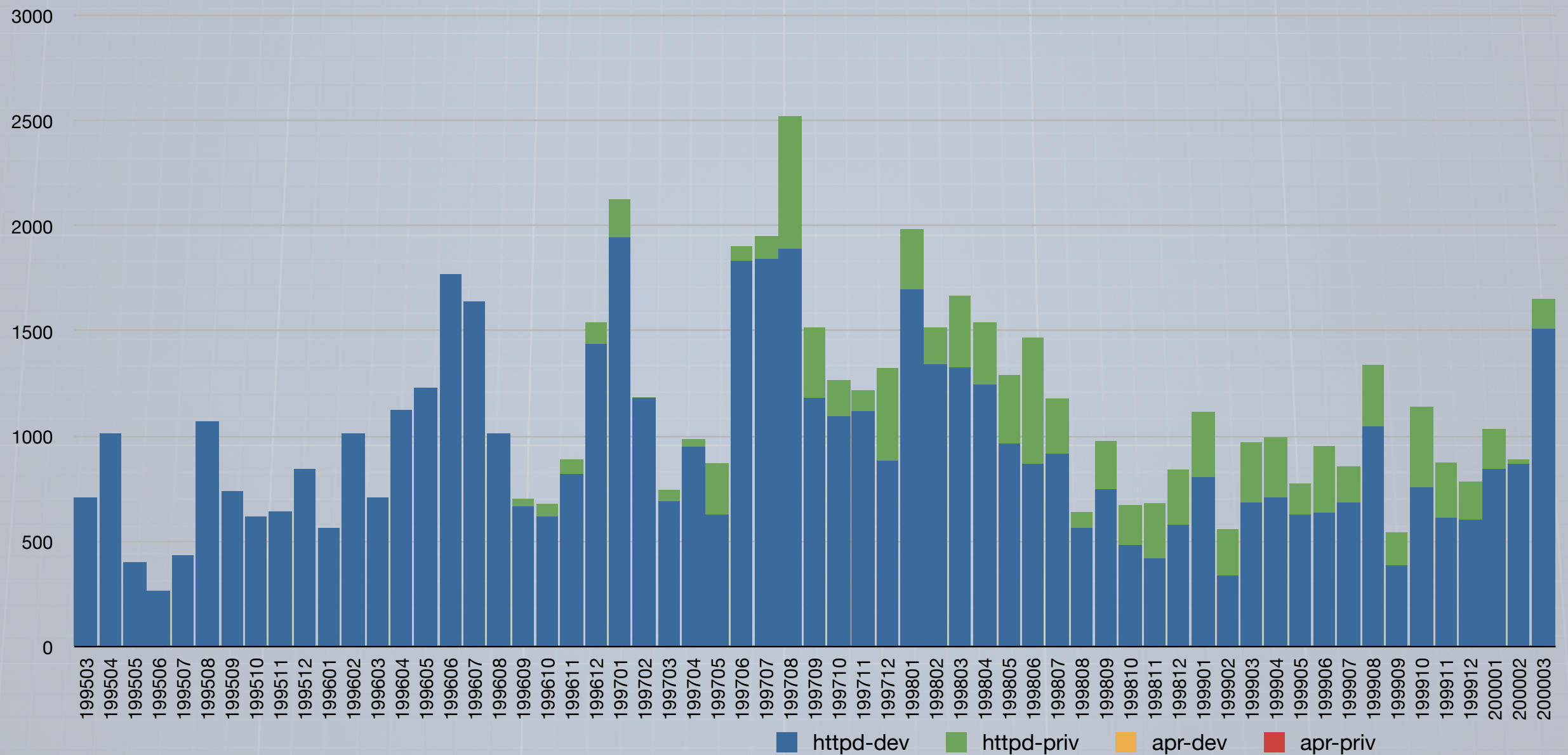
what does a web server do

introducing Apache 3.0

the nature of collaboration

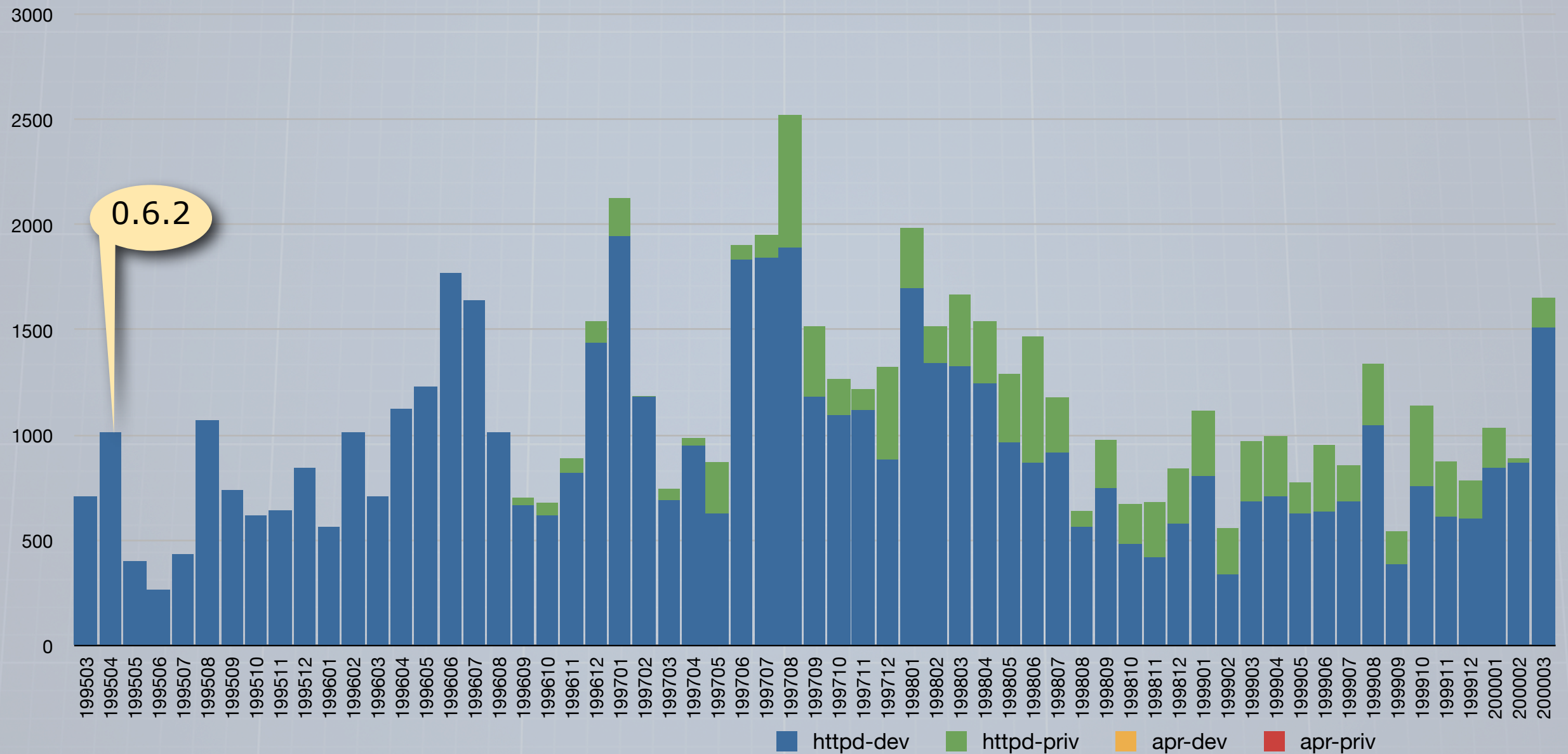
HTTP Server Project

First Five Years



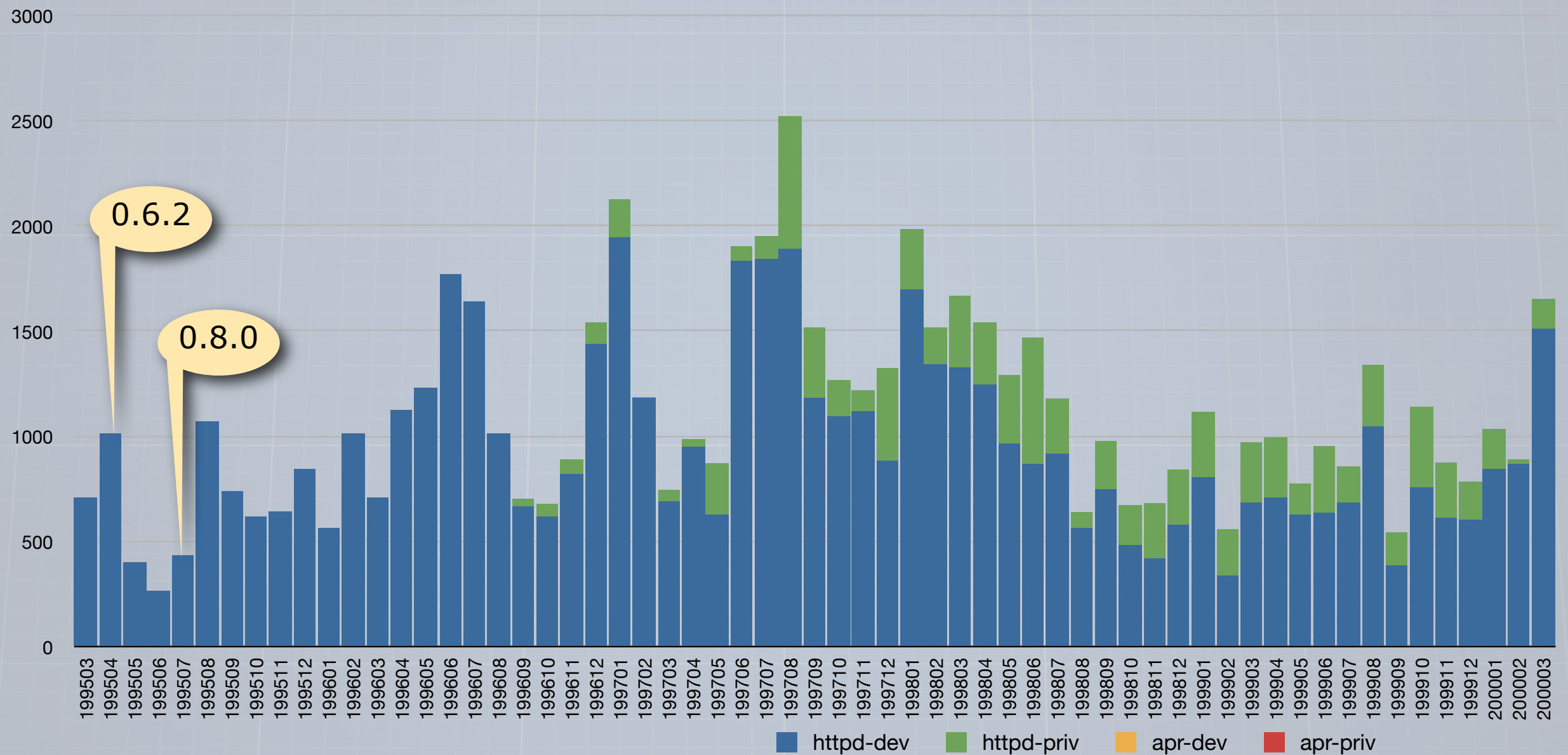
HTTP Server Project

First Five Years



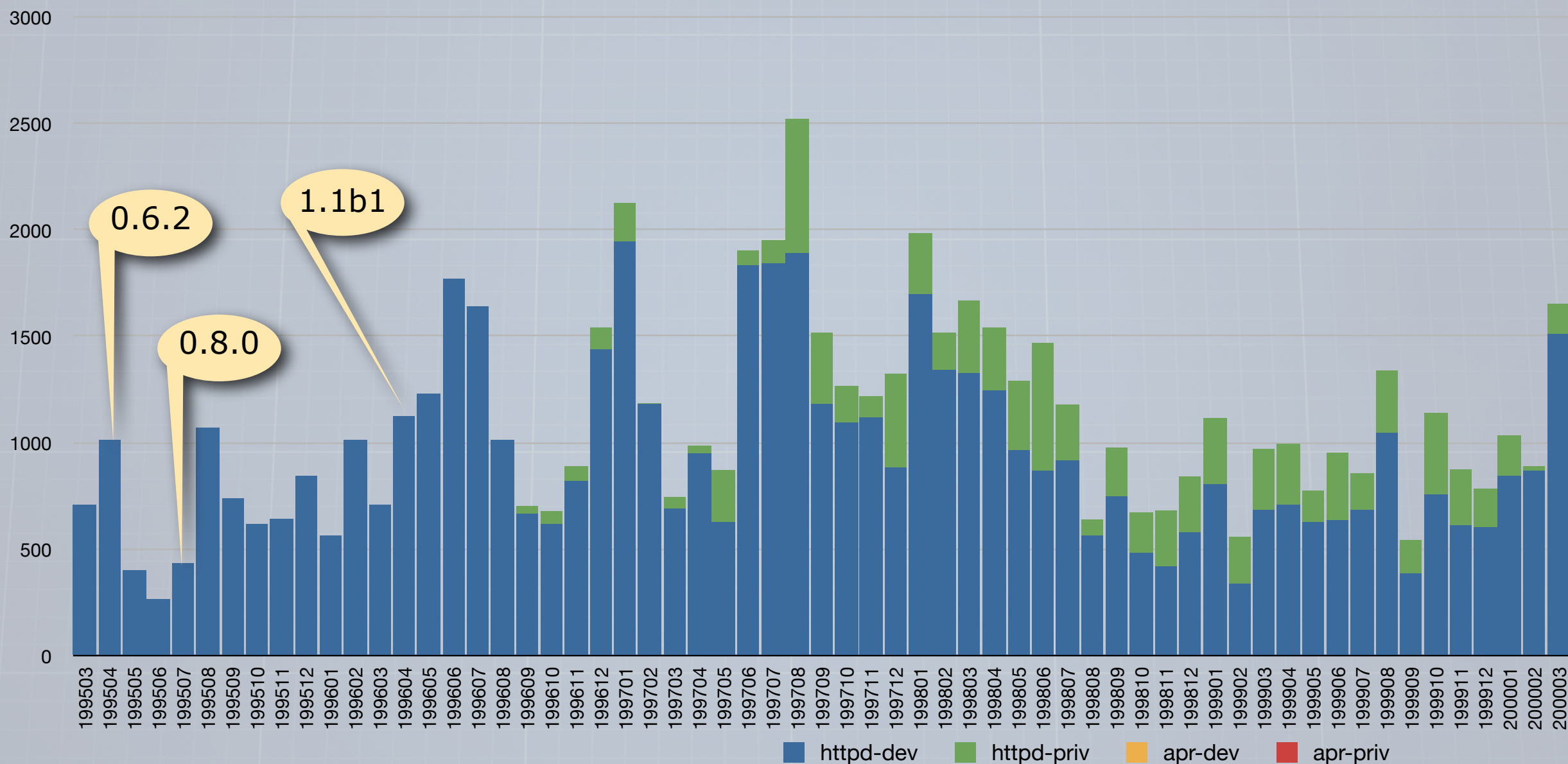
HTTP Server Project

First Five Years



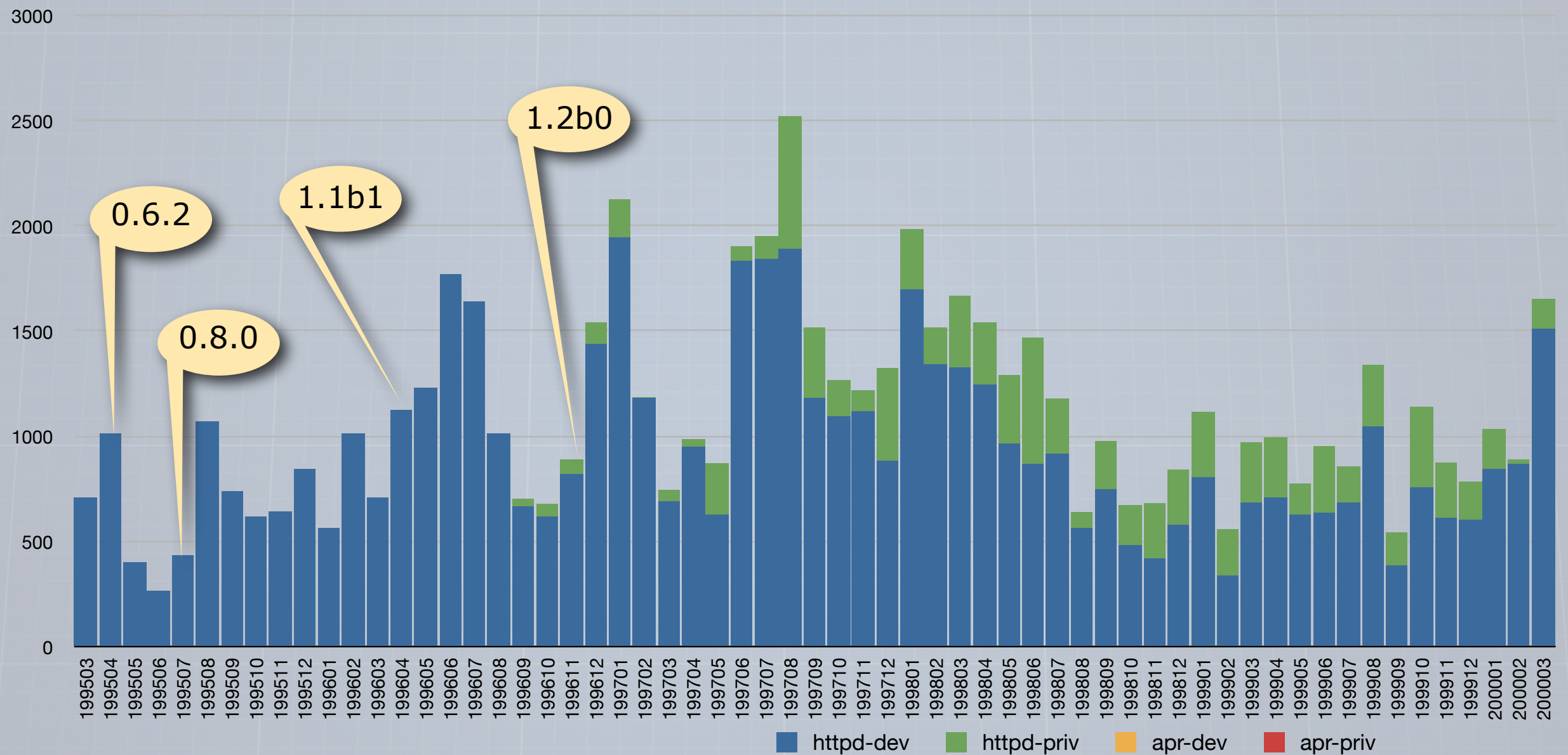
HTTP Server Project

First Five Years



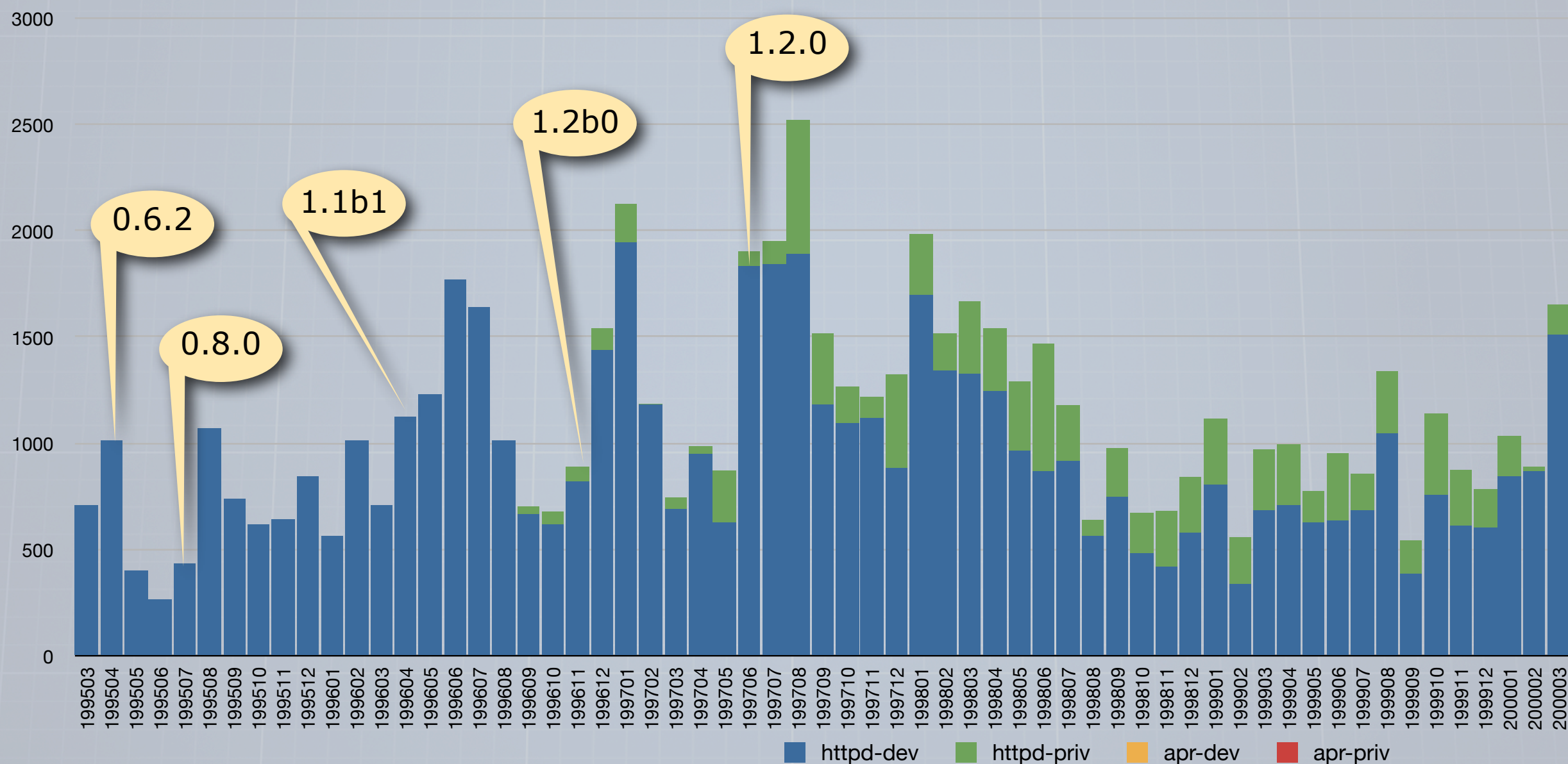
HTTP Server Project

First Five Years



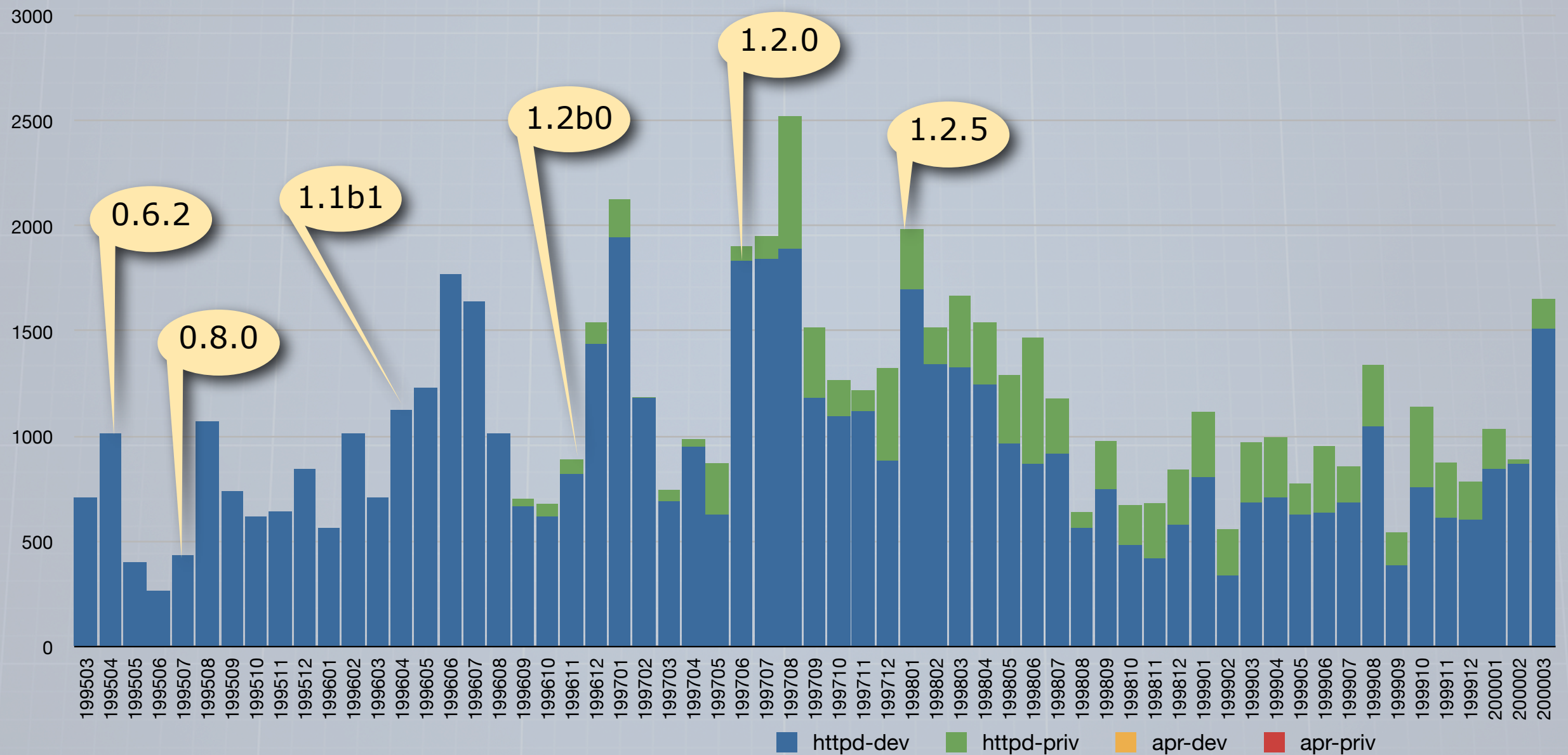
HTTP Server Project

First Five Years



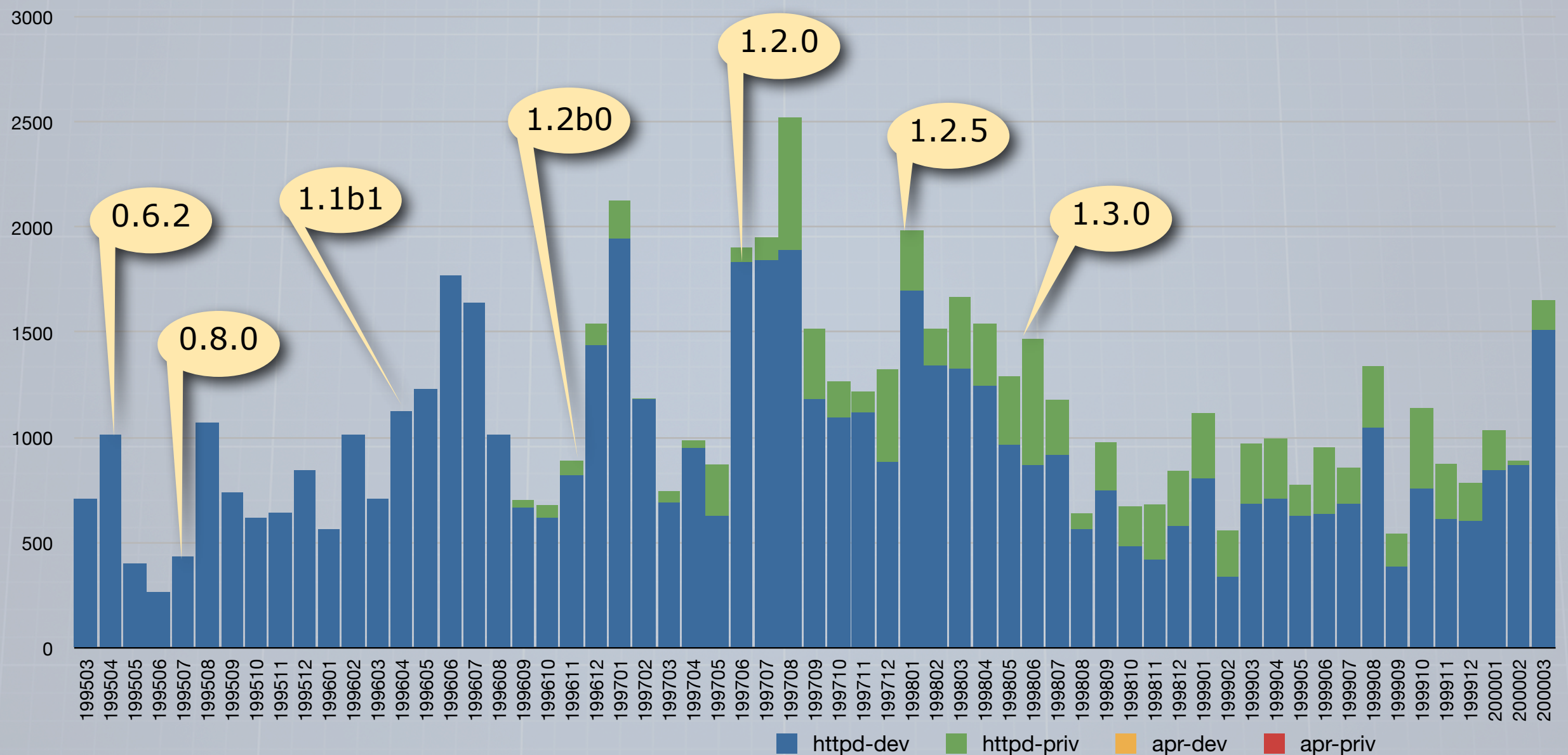
HTTP Server Project

First Five Years



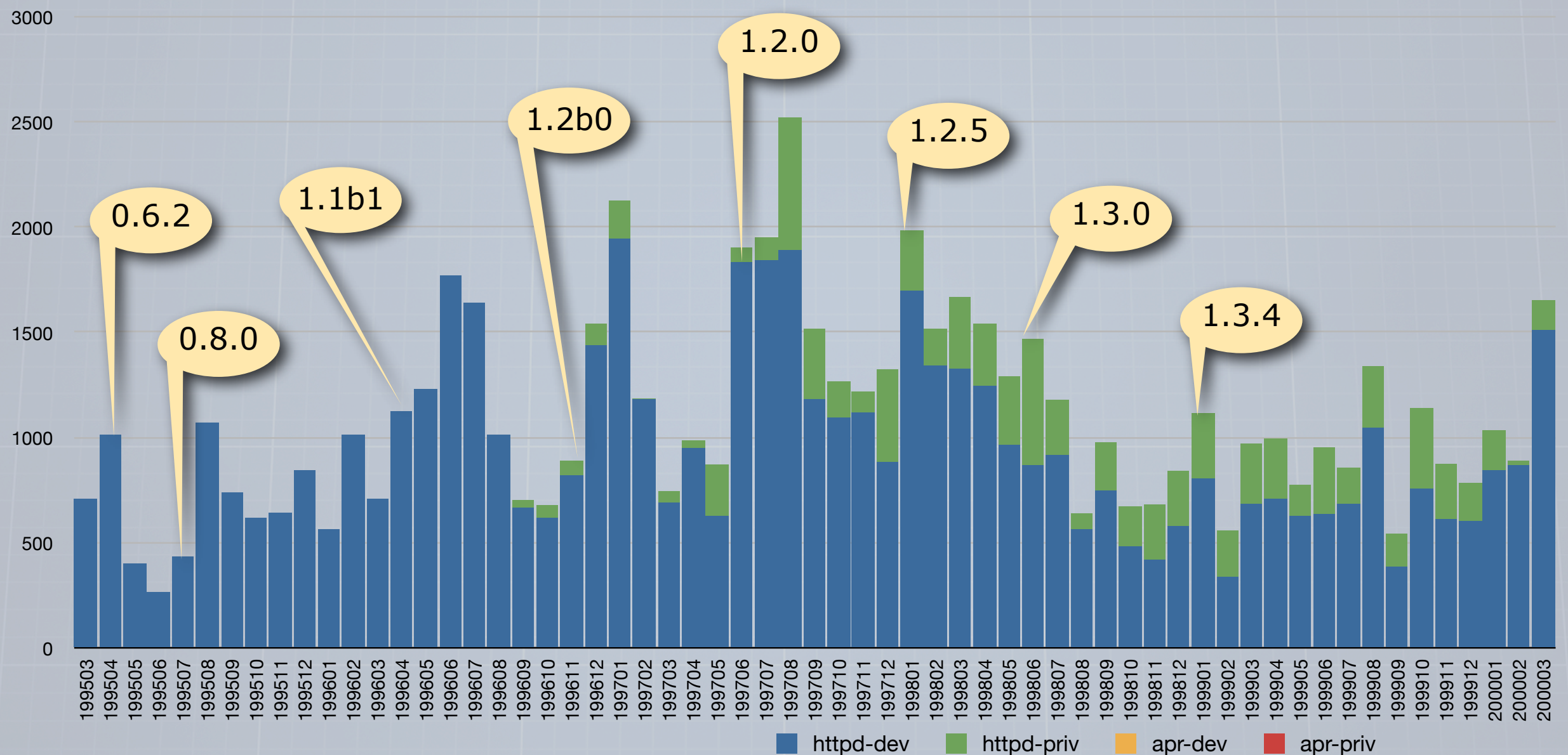
HTTP Server Project

First Five Years



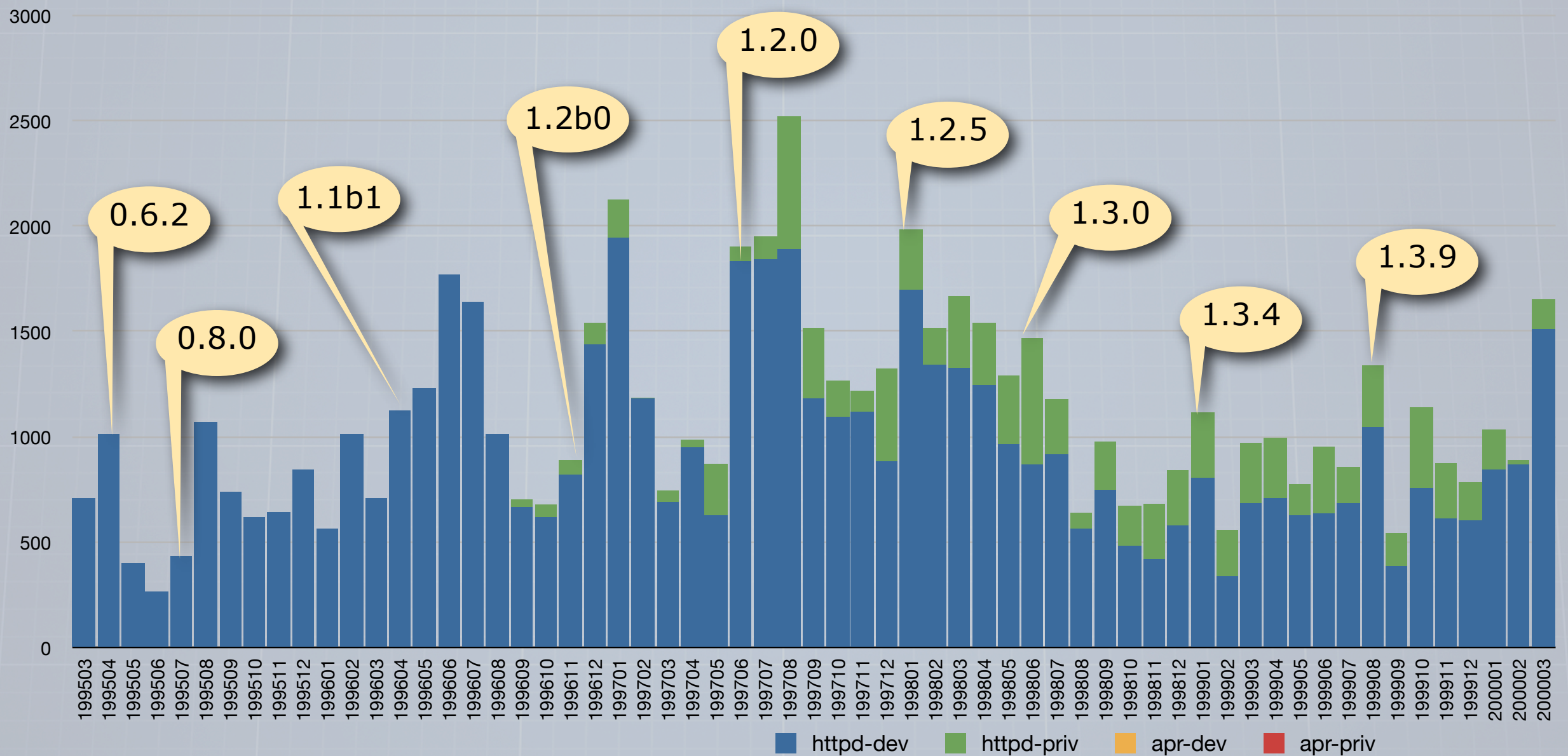
HTTP Server Project

First Five Years



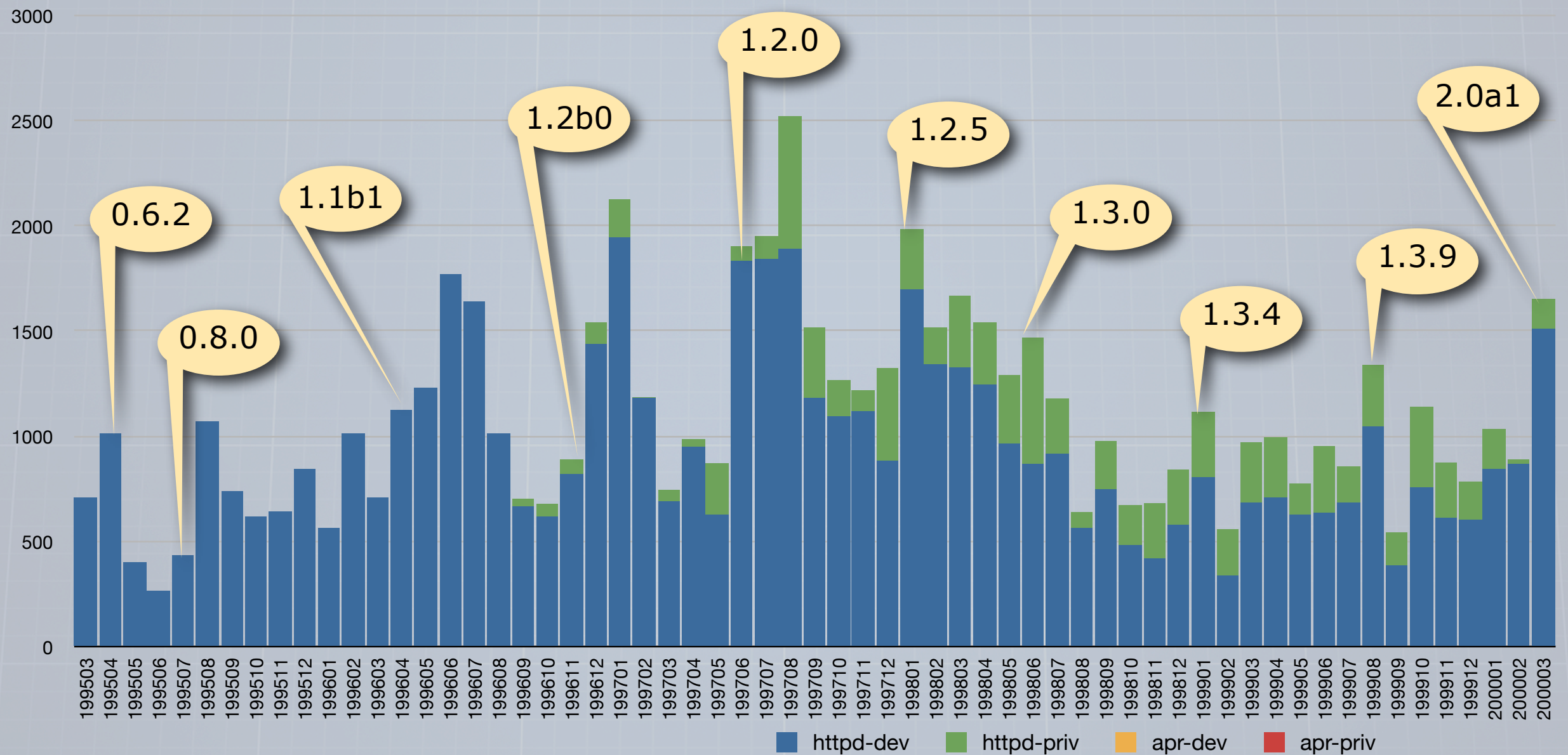
HTTP Server Project

First Five Years

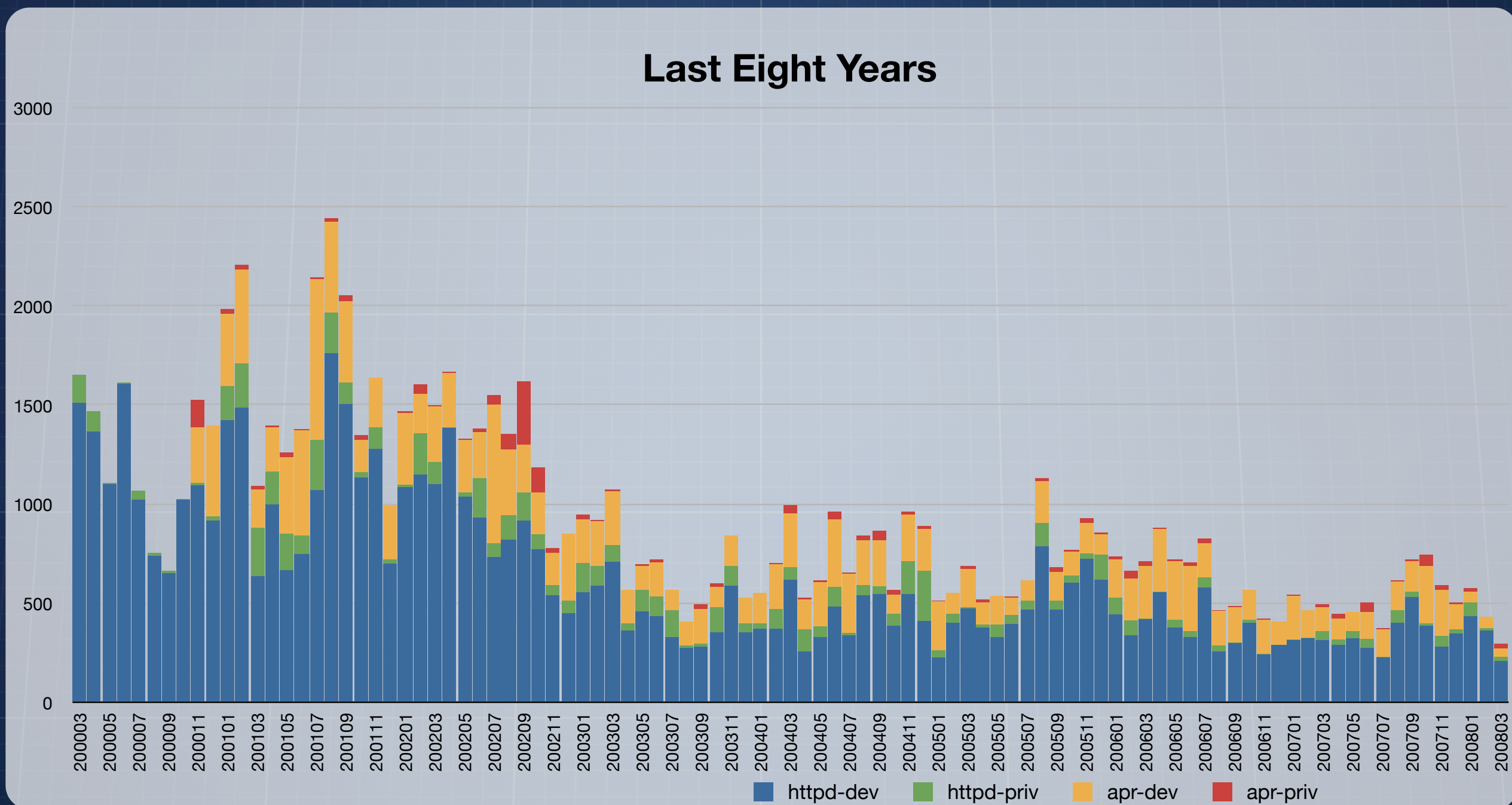


HTTP Server Project

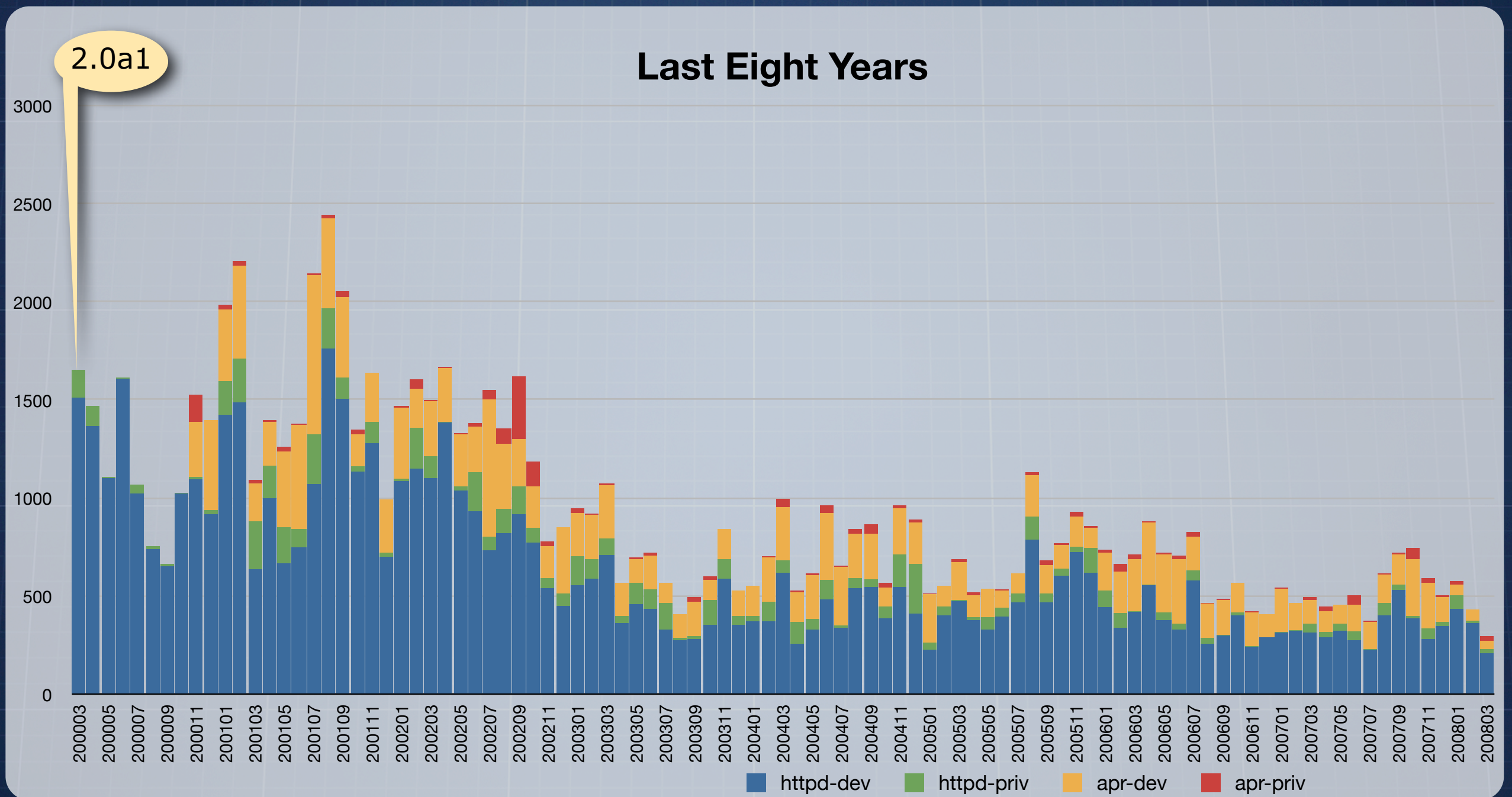
First Five Years



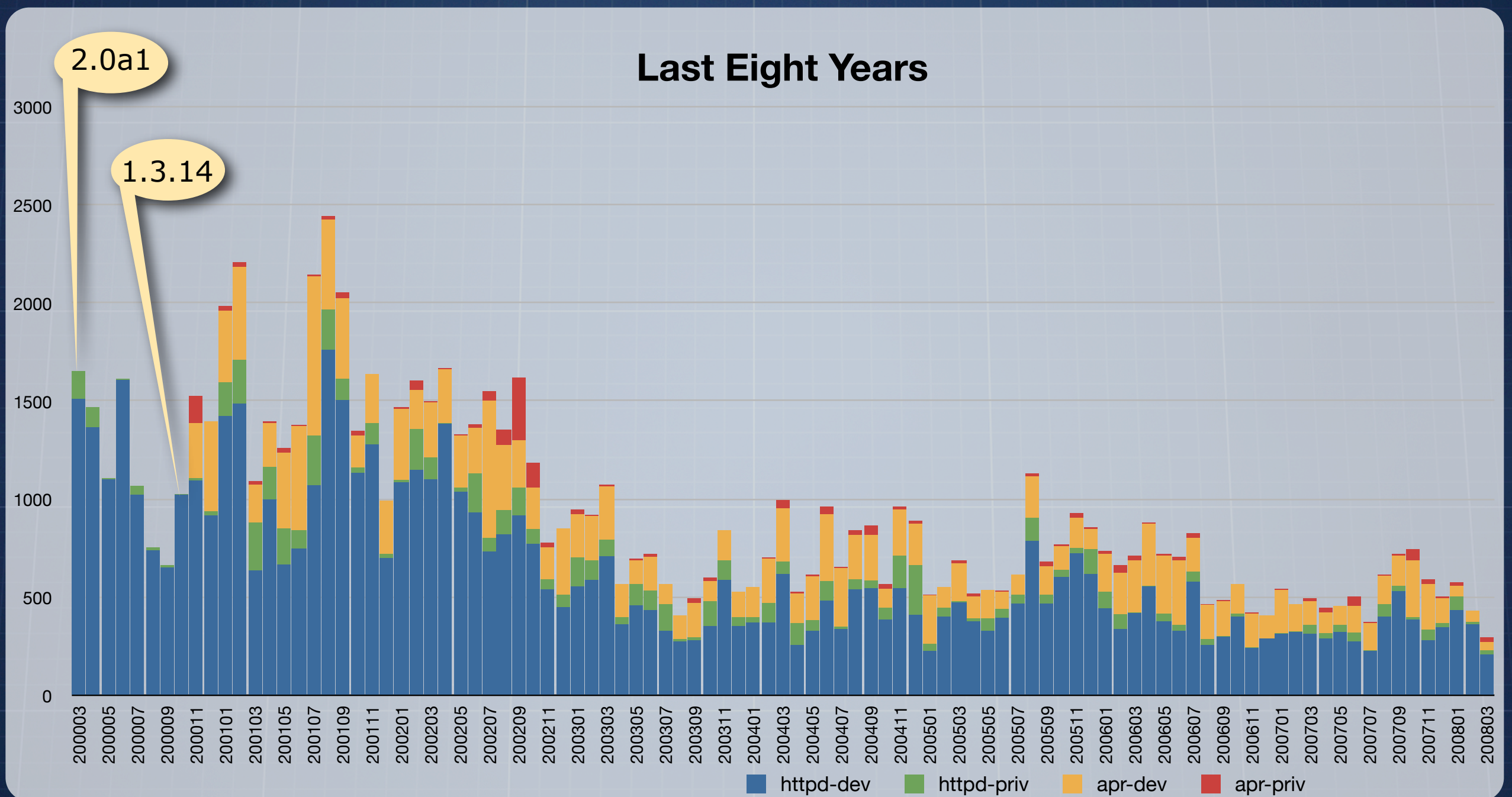
HTTP Server Project



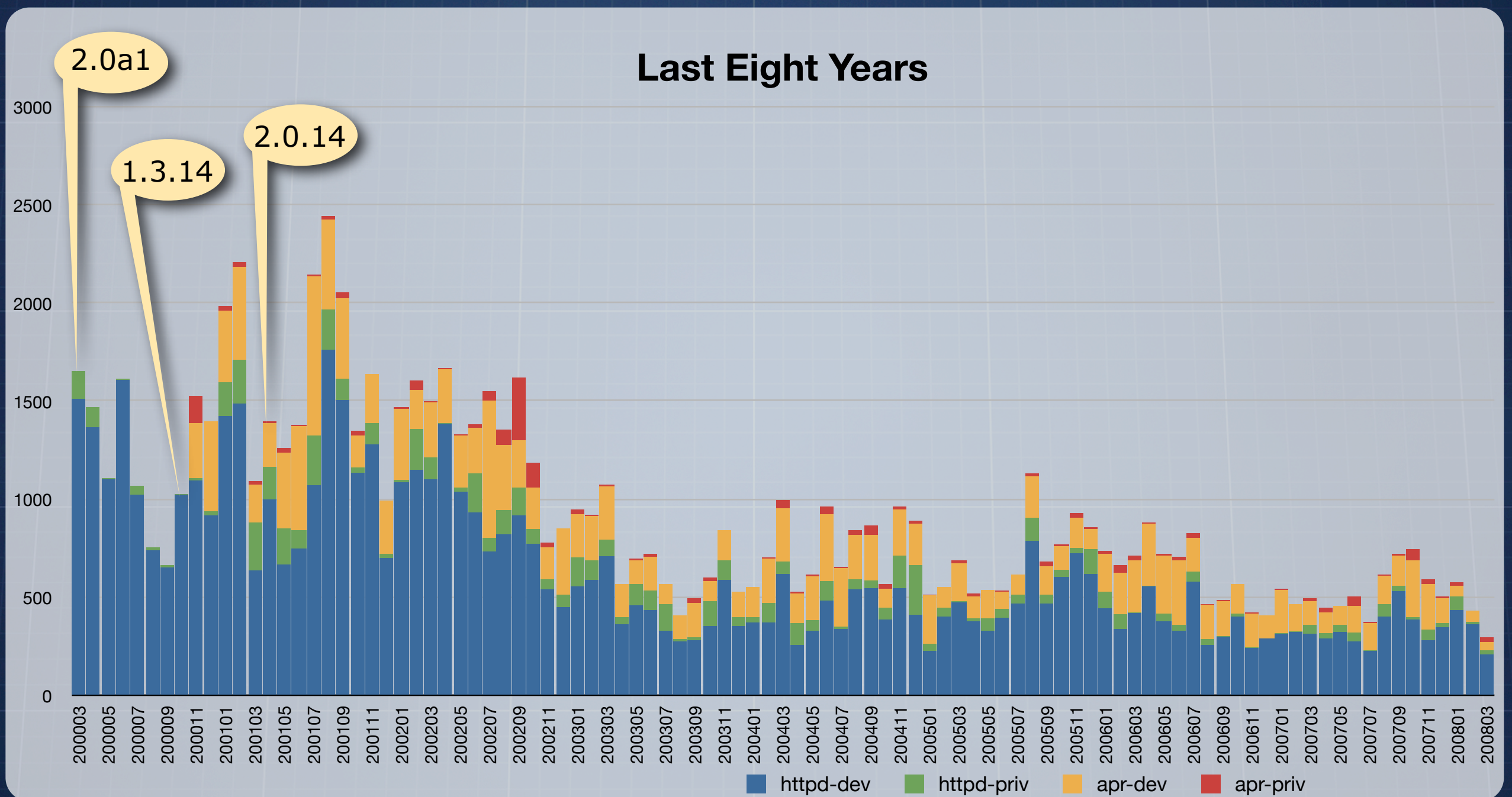
HTTP Server Project



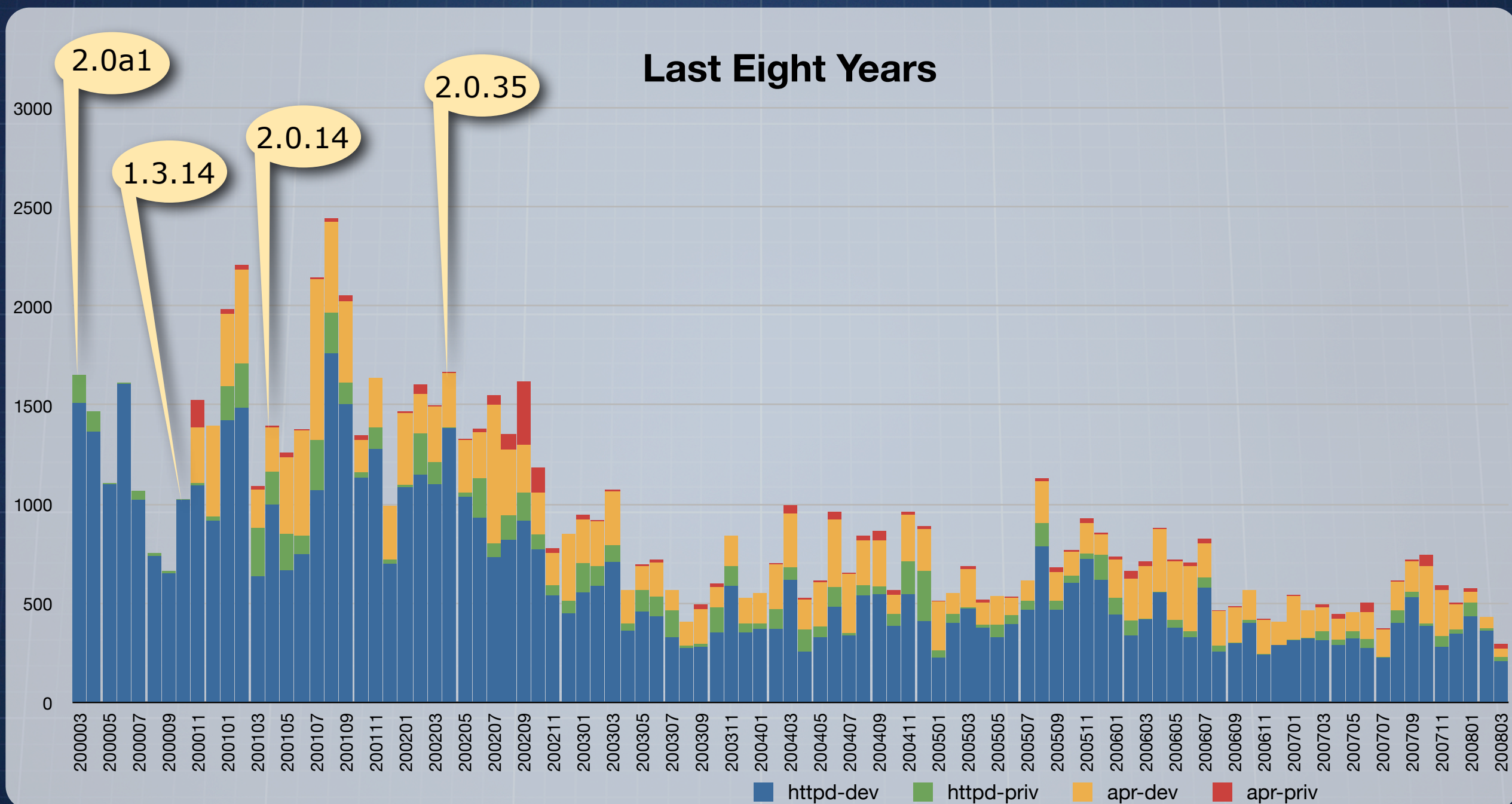
HTTP Server Project



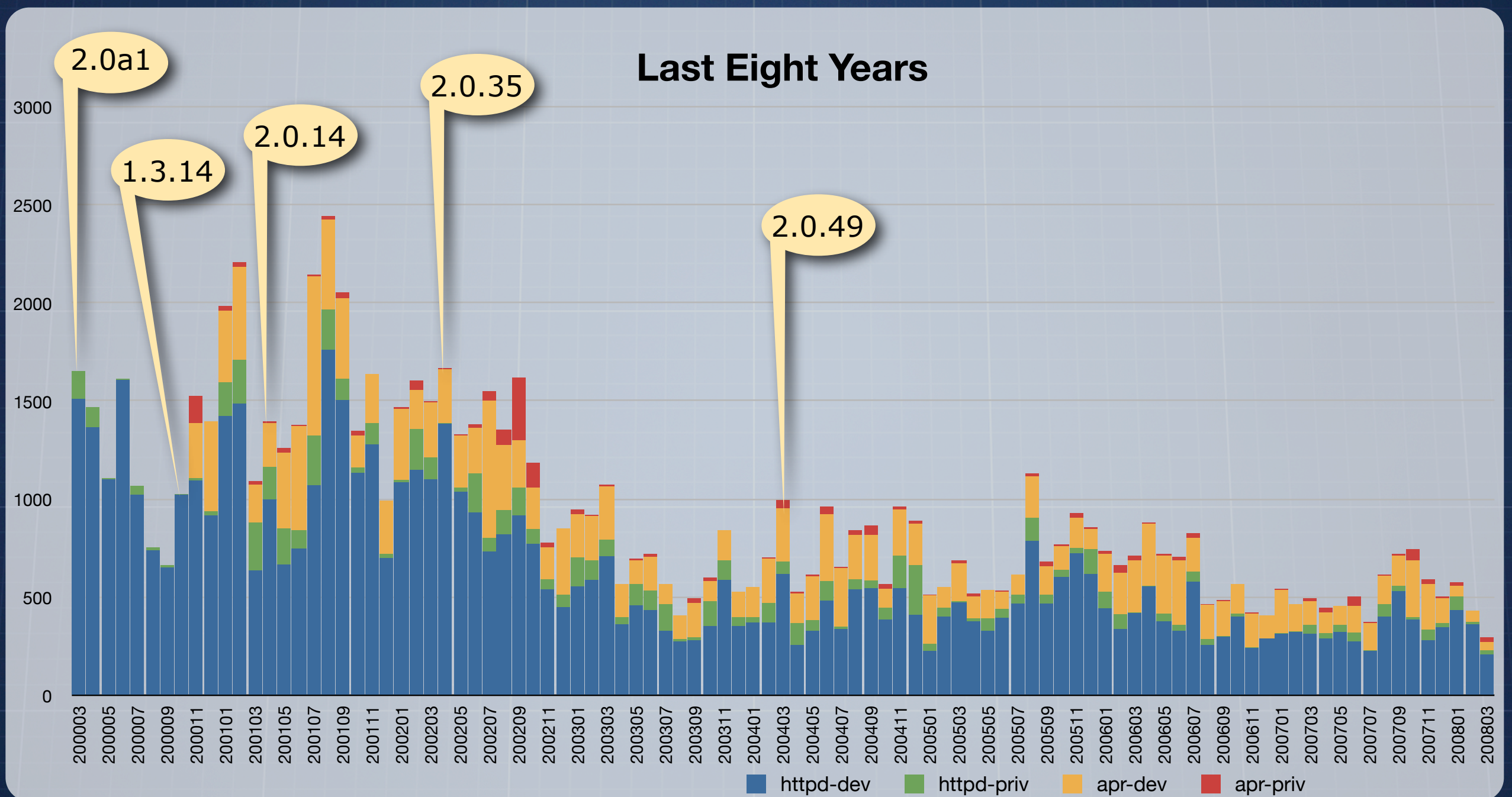
HTTP Server Project



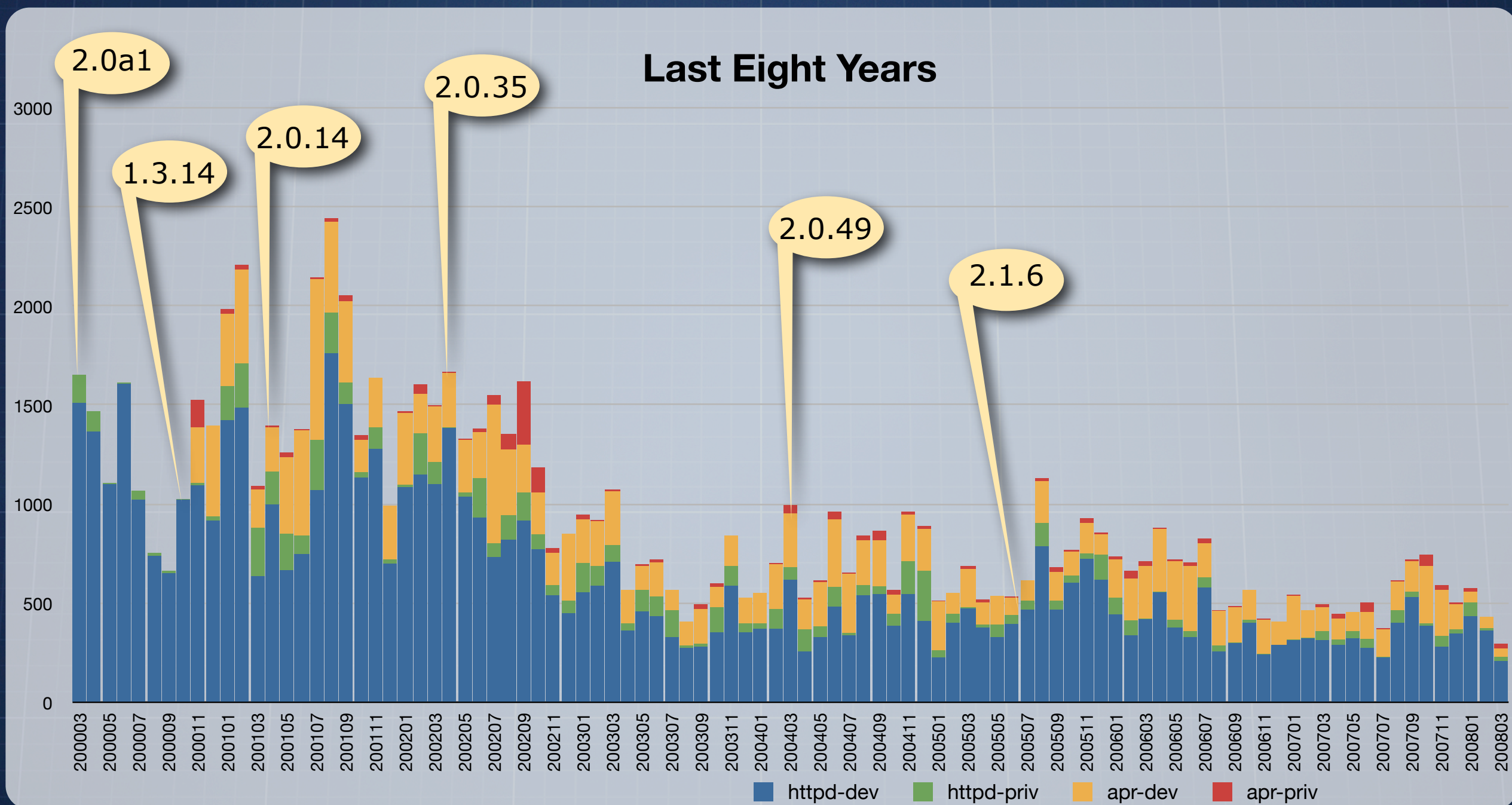
HTTP Server Project



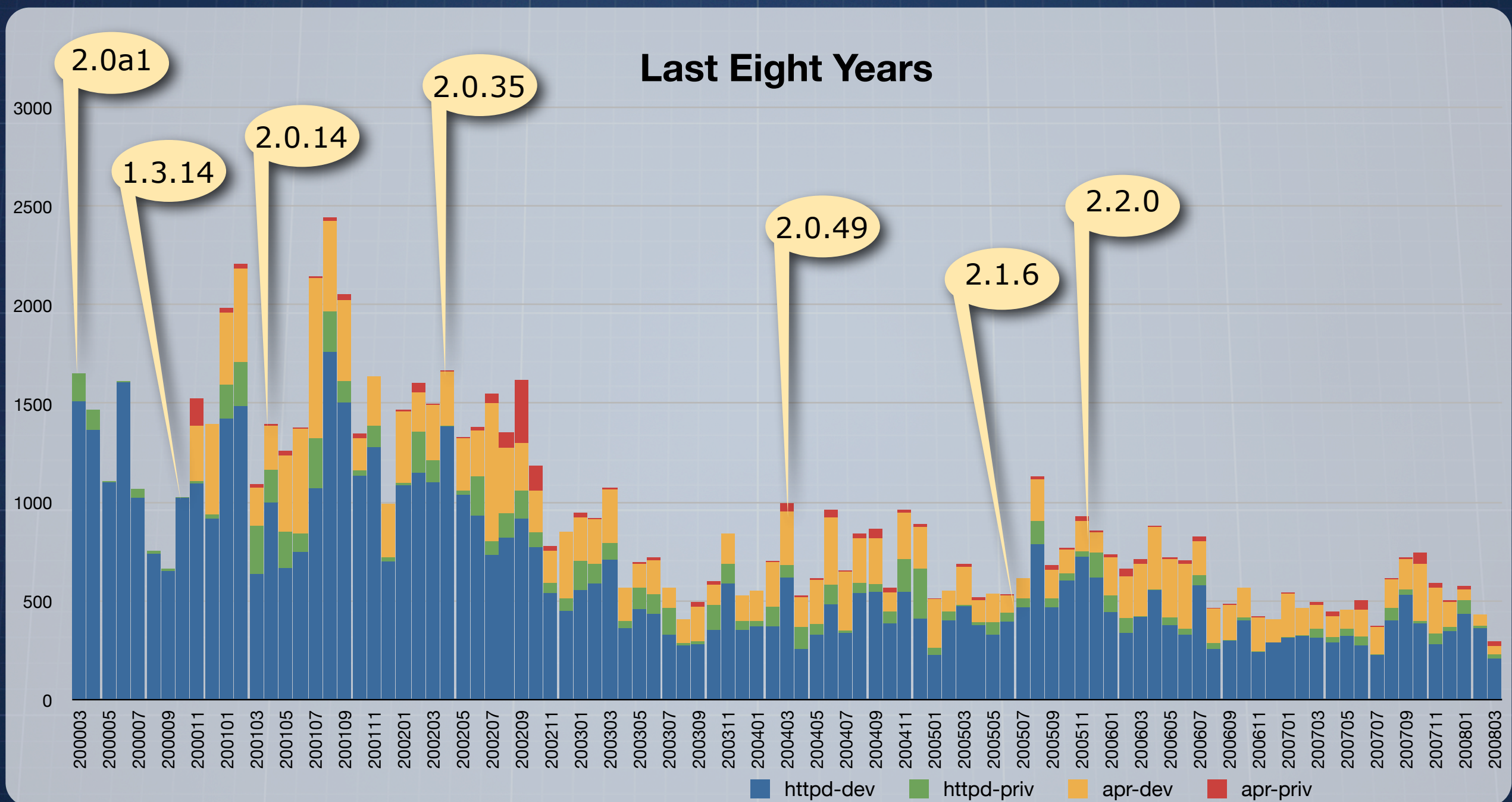
HTTP Server Project



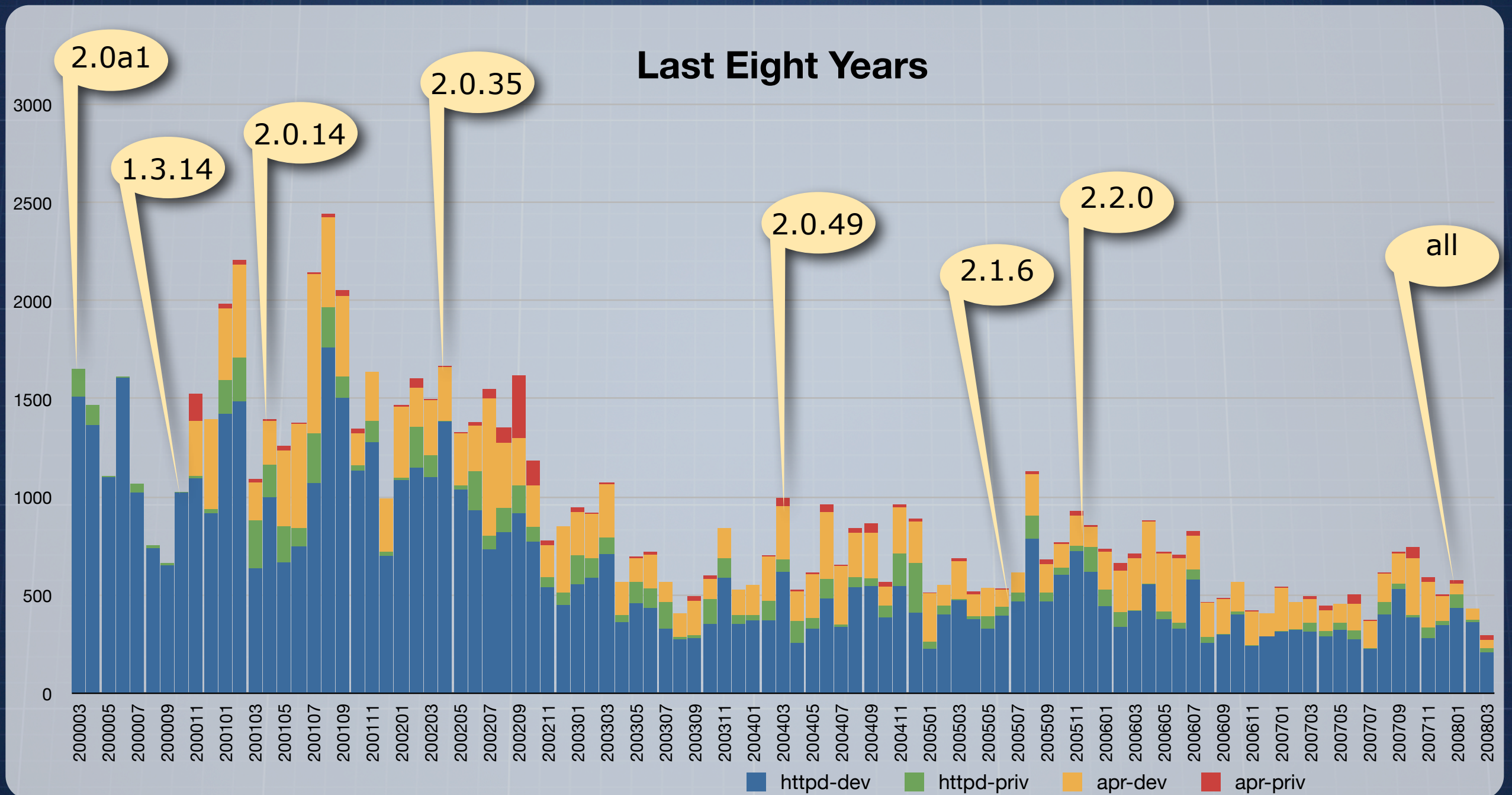
HTTP Server Project



HTTP Server Project



HTTP Server Project



Summary

Collaboration is best when people have work to do

When activity is low, people don't look for work

When activity is high, people look to help out

Creating activity is NOT a group decision!

**Infrequent releases cause lower expectations
and our expectations are at the lowest point in our history!**

Outline

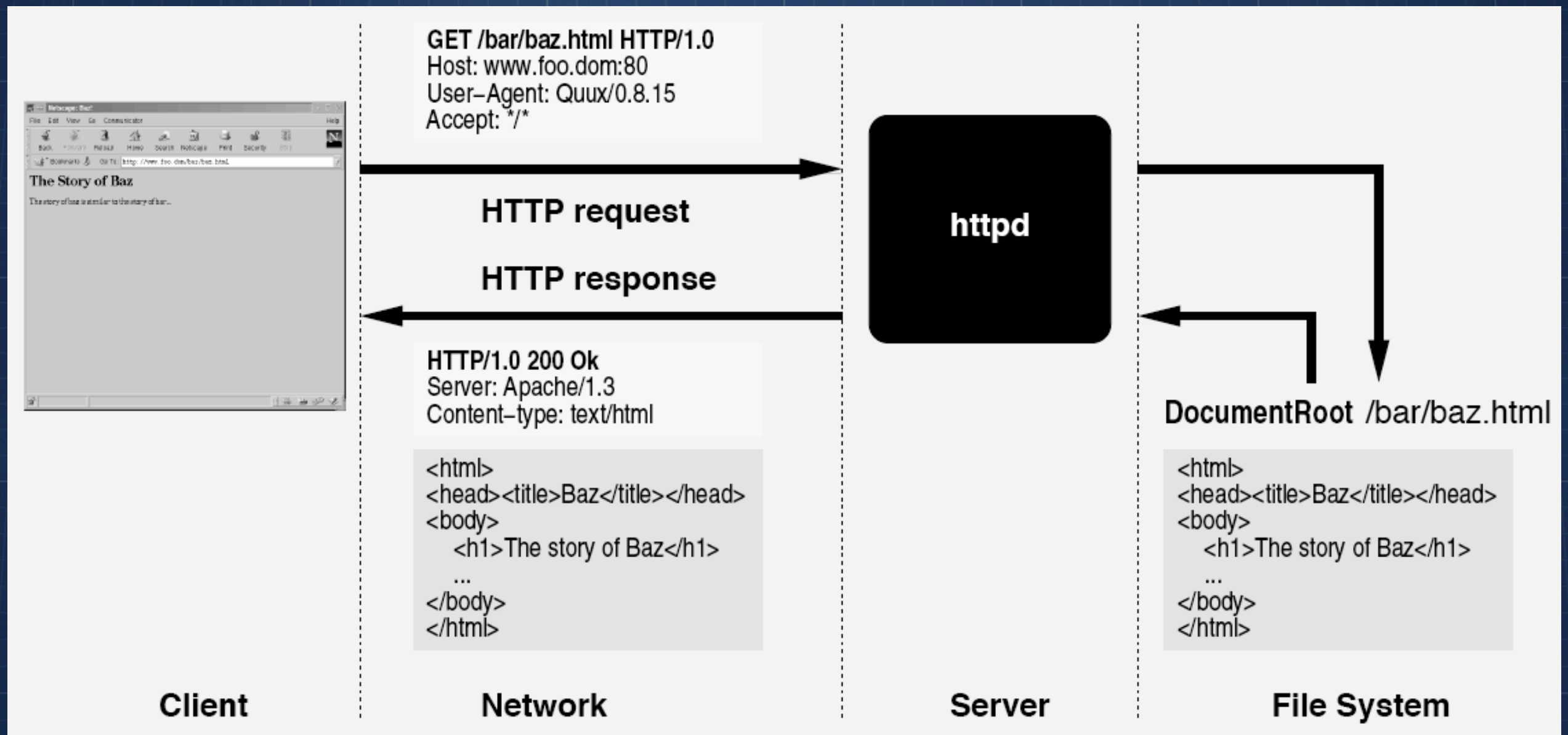
a bit of history

what does a web server do

introducing Apache 3.0

the nature of collaboration

Role of an HTTP Server



[Ralf S. Engelschall, 2000]

things a web server does

read configuration

listen for connections

parse requests

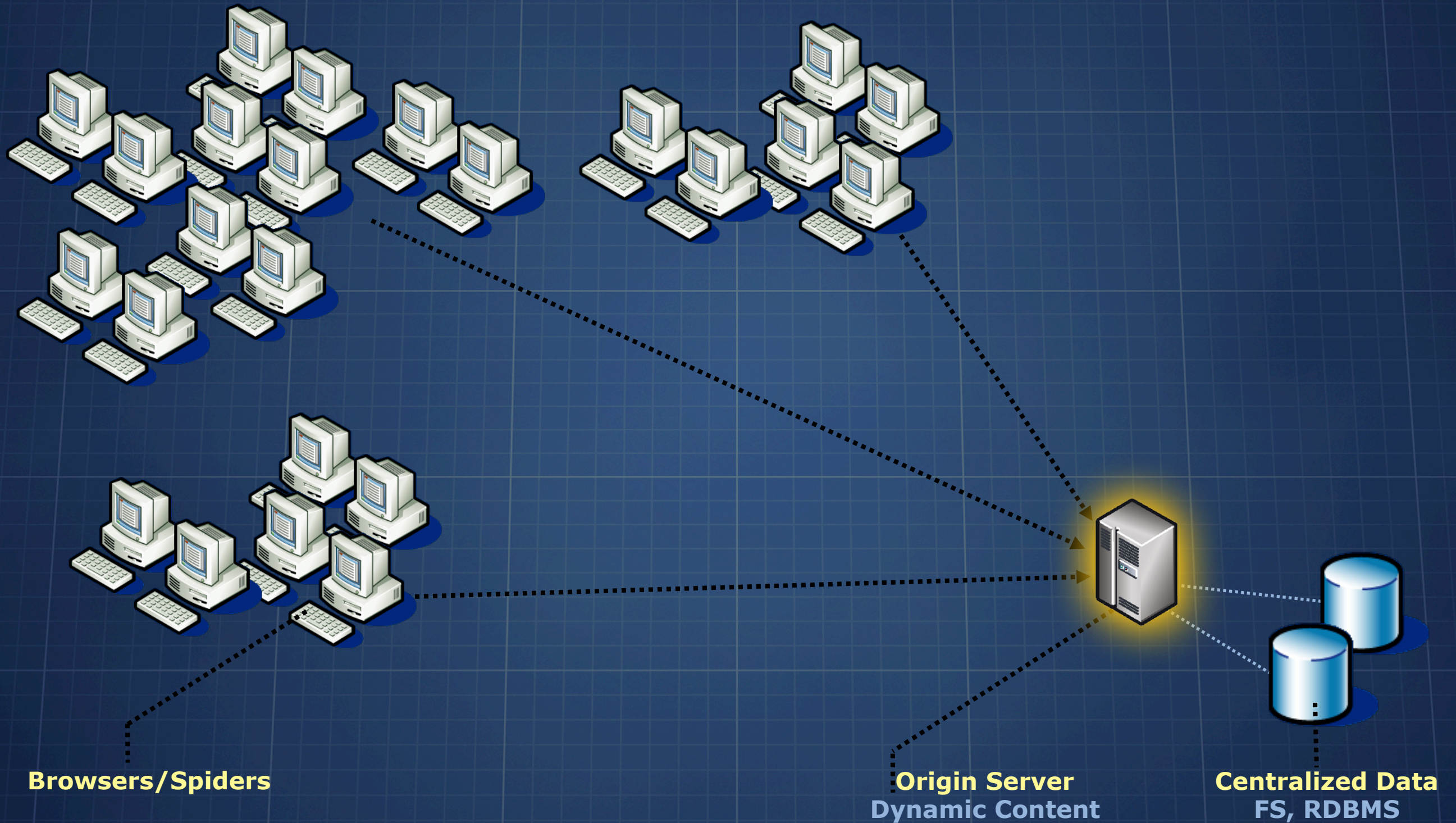
map URI and method to

- redirects and aliases
- cache
- access control, authenticate, authorize
- handler

determine response

send response

Role of an Apache 1.0

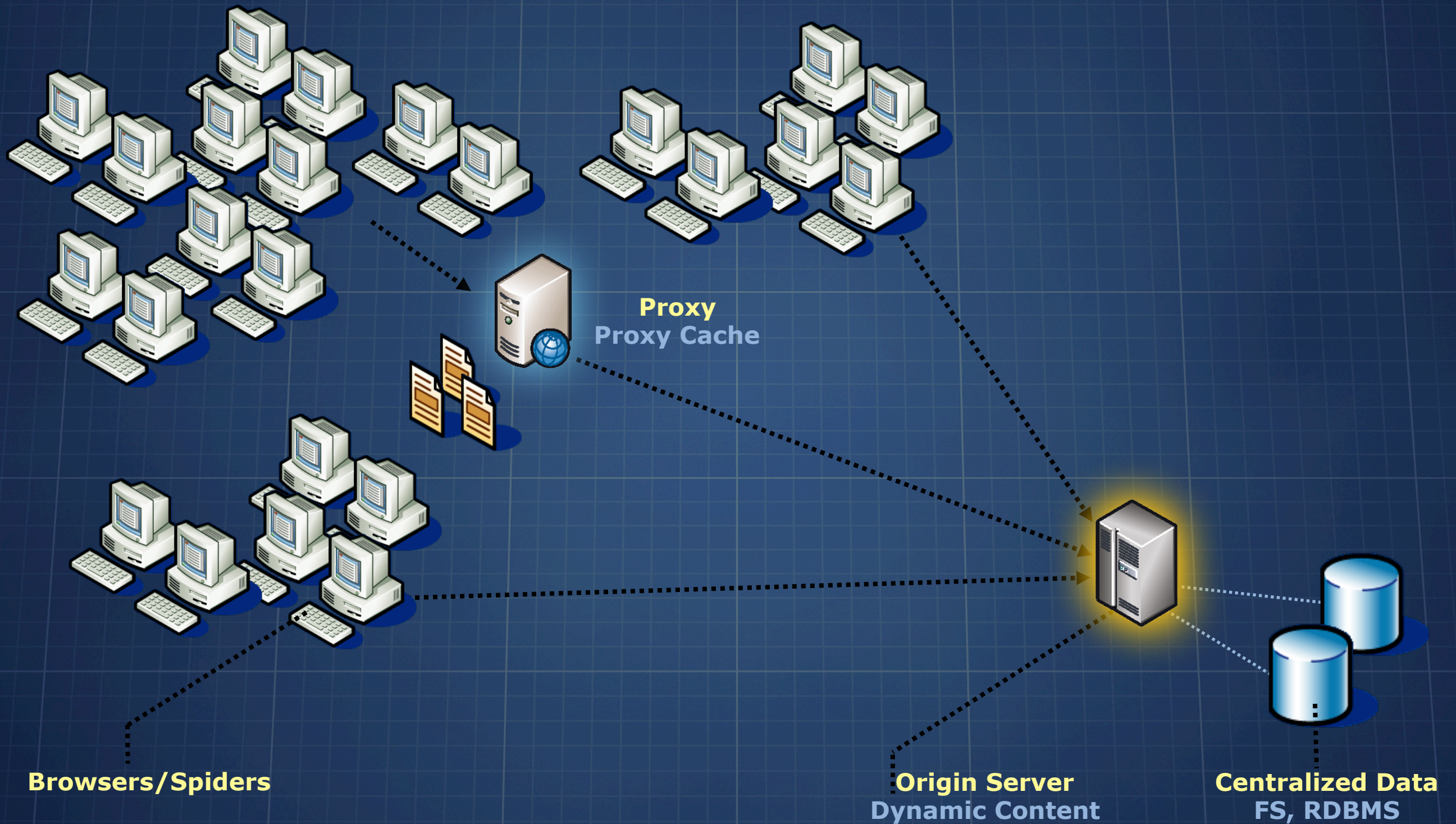


Browsers/Spiders

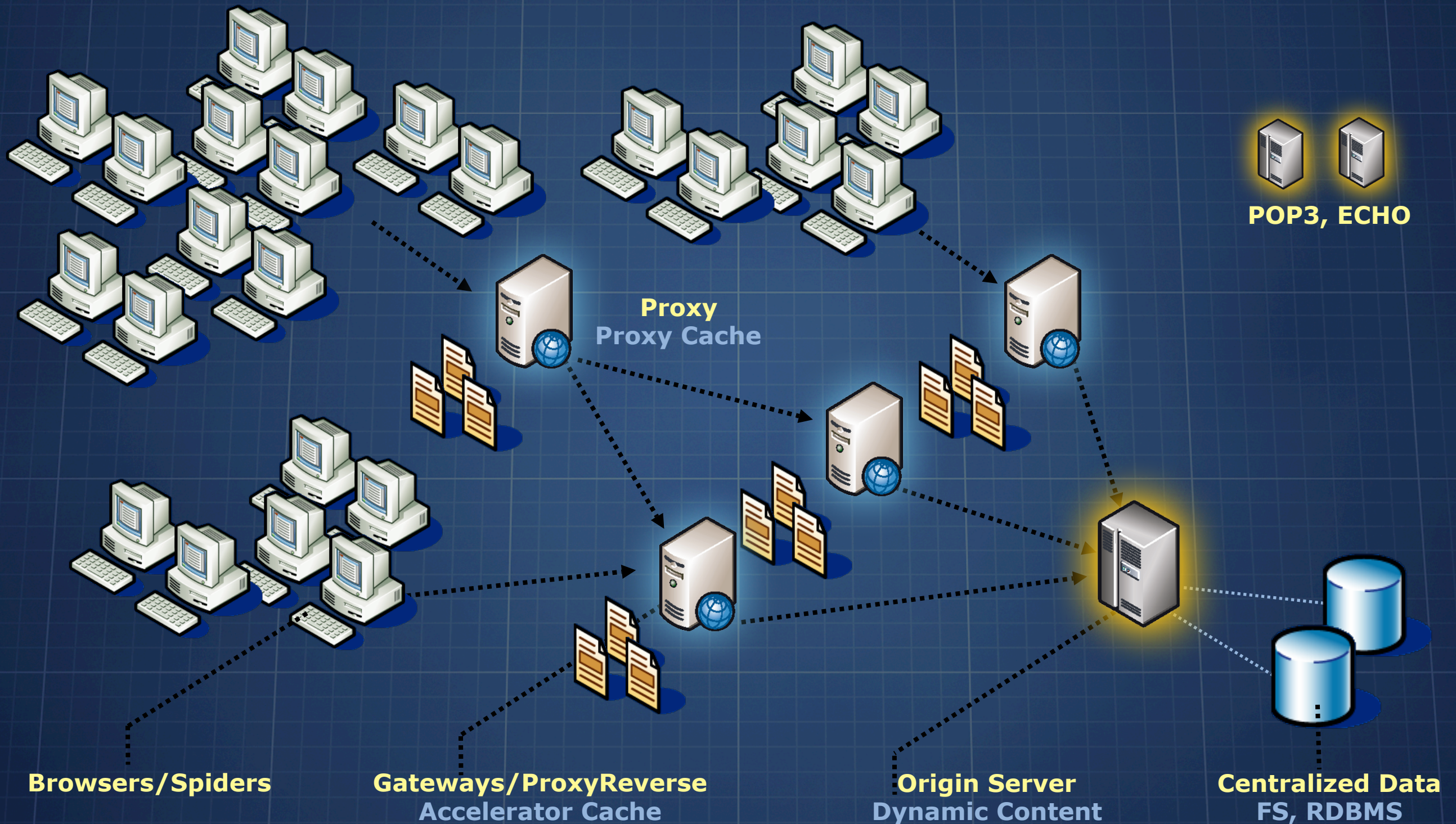
Origin Server
Dynamic Content

Centralized Data
FS, RDBMS

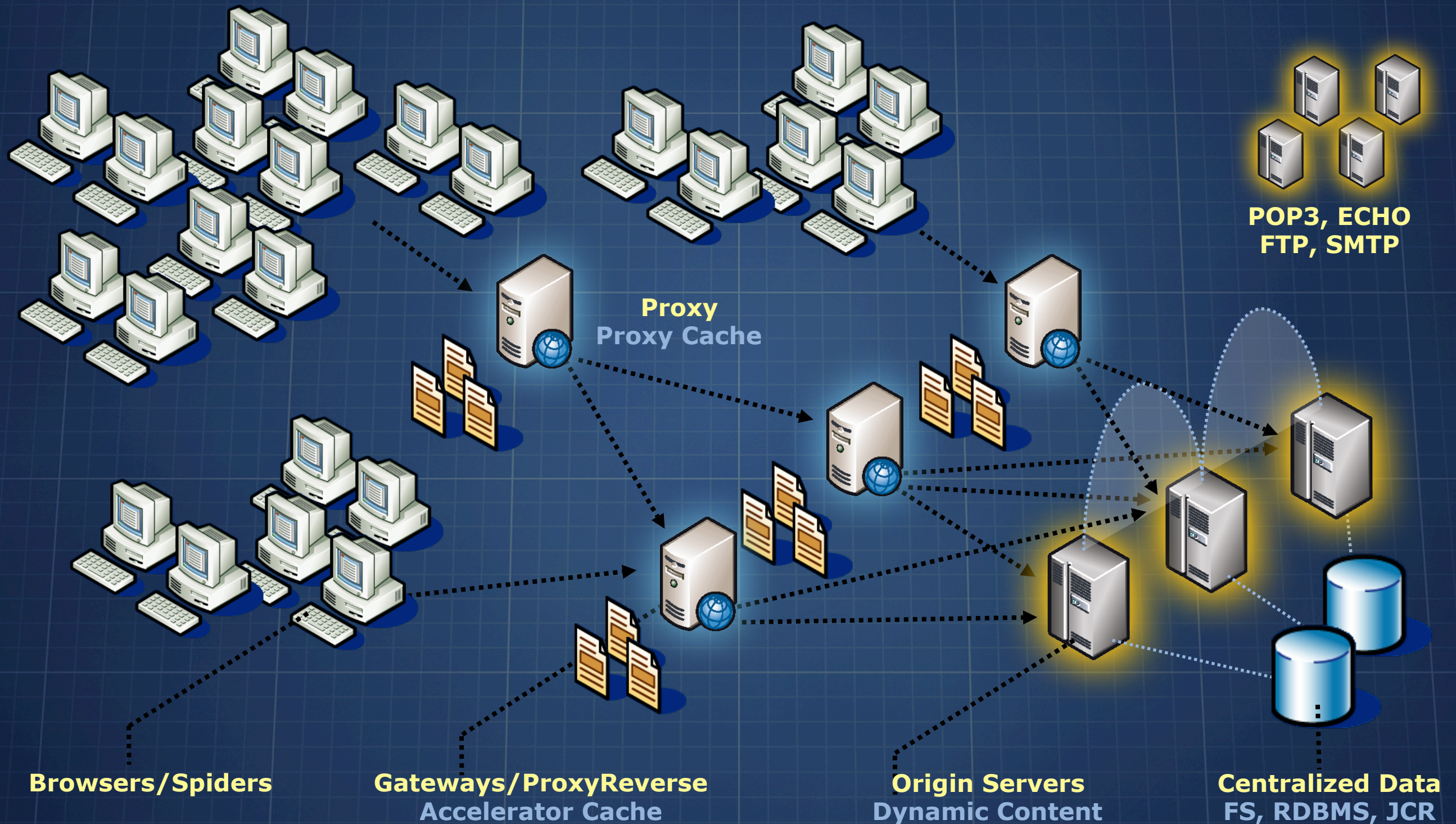
Role of an Apache 1.1



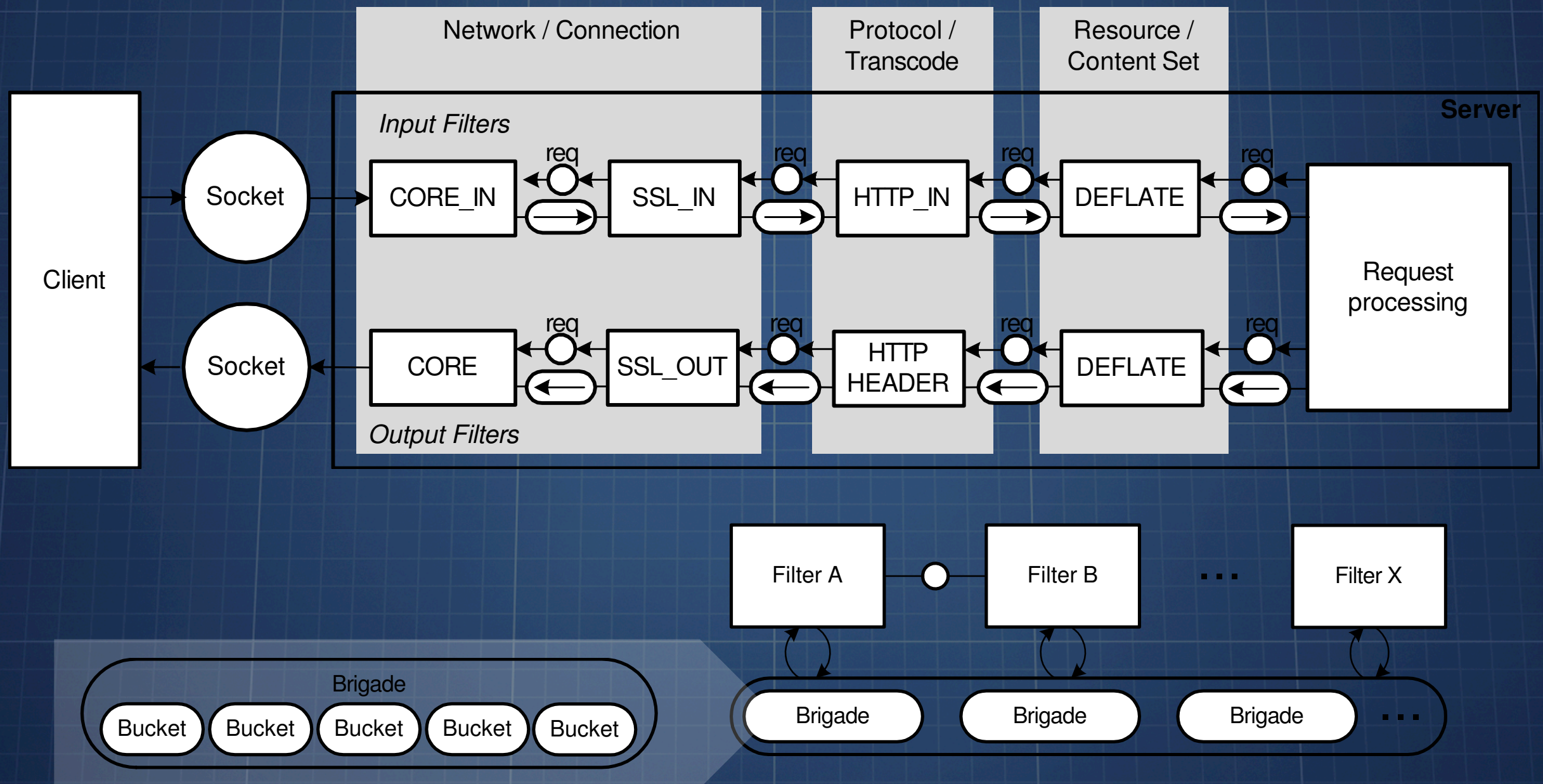
Role of an Apache 2.0



Role of an Apache 2.2



Apache httpd 2.x Filters



[HPI Apache Modeling Project, <http://apache.hpi.uni-potsdam.de/>]

Summary

Apache httpd does a lot more now than in 1.0

Yet the internal architecture is basically the same

In other words,

- **our architecture was not designed for this stuff (i.e., it is only adequately supported, not ideally)**
- **much of it has been slapped together in odd ways to avoid incompatible changes and configuration**
- **much of it isn't used**
- **some of it shouldn't be used**

and it is interfering with our ability to HAVE FUN

Outline

a bit of history

what does a web server do

introducing Apache 3.0

the nature of collaboration

It's time for Apache 3.0

It's time for Apache 3.0

One process model per common platform

- no more guessing who is running how on what
- no more support for platforms that suck (use 1.3.x)
- if you want to run on/with something else, you've got the source code

It's time for Apache 3.0

One process model per common platform

- no more guessing who is running how on what
- no more support for platforms that suck (use 1.3.x)
- if you want to run on/with something else, you've got the source code

All compile-time choices are GONE

- httpd and httpsd are DIFFERENT executables
- APR is replaced by Moccasin

It's time for Apache 3.0

One process model per common platform

- no more guessing who is running how on what
- no more support for platforms that suck (use 1.3.x)
- if you want to run on/with something else, you've got the source code

All compile-time choices are GONE

- httpd and httpsd are DIFFERENT executables
- APR is replaced by Moccasin

All APIs are based on Waka

- get those sticky hands off my data structures!

It's time for Apache 3.0

One process model per common platform

- no more guessing who is running how on what
- no more support for platforms that suck (use 1.3.x)
- if you want to run on/with something else, you've got the source code

All compile-time choices are GONE

- httpd and httpsd are DIFFERENT executables
- APR is replaced by Moccasin

All APIs are based on Waka

- get those sticky hands off my data structures!

“Kiss my ass” compatibility

Simplify Process Model

Simplify Process Model

Modular Process Models

- Apache 0.3
fork model
- Apache 1.0
prefork model
- Apache 1.3 introduced non-Unix models
prefork, win32
- Apache 2.x introduced MPMs
prefork, beos, mpmt, os2, netware, threaded, winnt, worker, event

Simplify Process Model

Modular Process Models

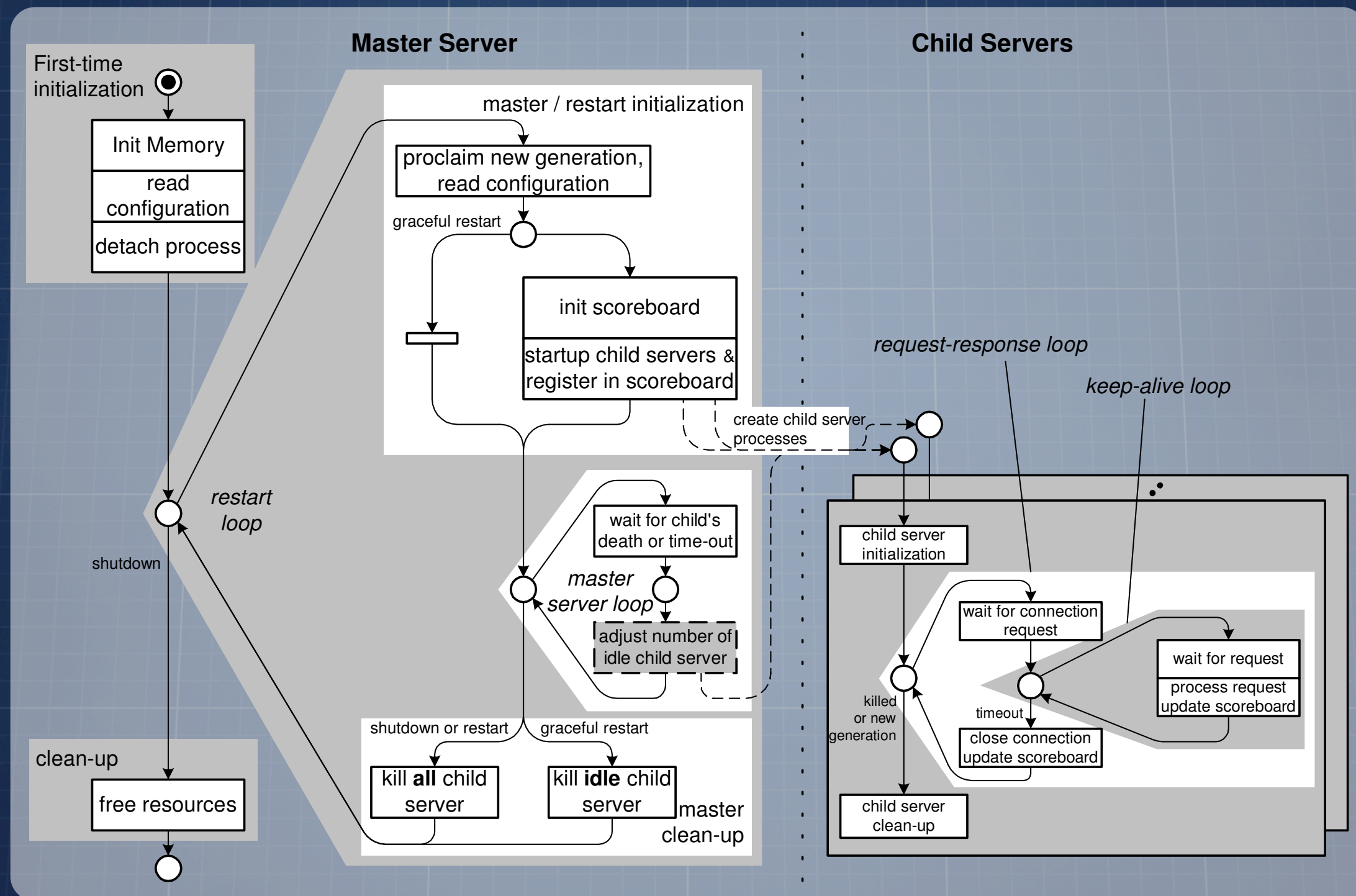
- Apache 0.3
fork model
- Apache 1.0
prefork model
- Apache 1.3 introduced non-Unix models
prefork, win32
- Apache 2.x introduced MPMs
prefork, beos, mpmt, os2, netware, threaded, winnt, worker, event

Apache 3.0 will just use the right MPM

based on whatever our own developers have tested to be
the best choice for a given platform

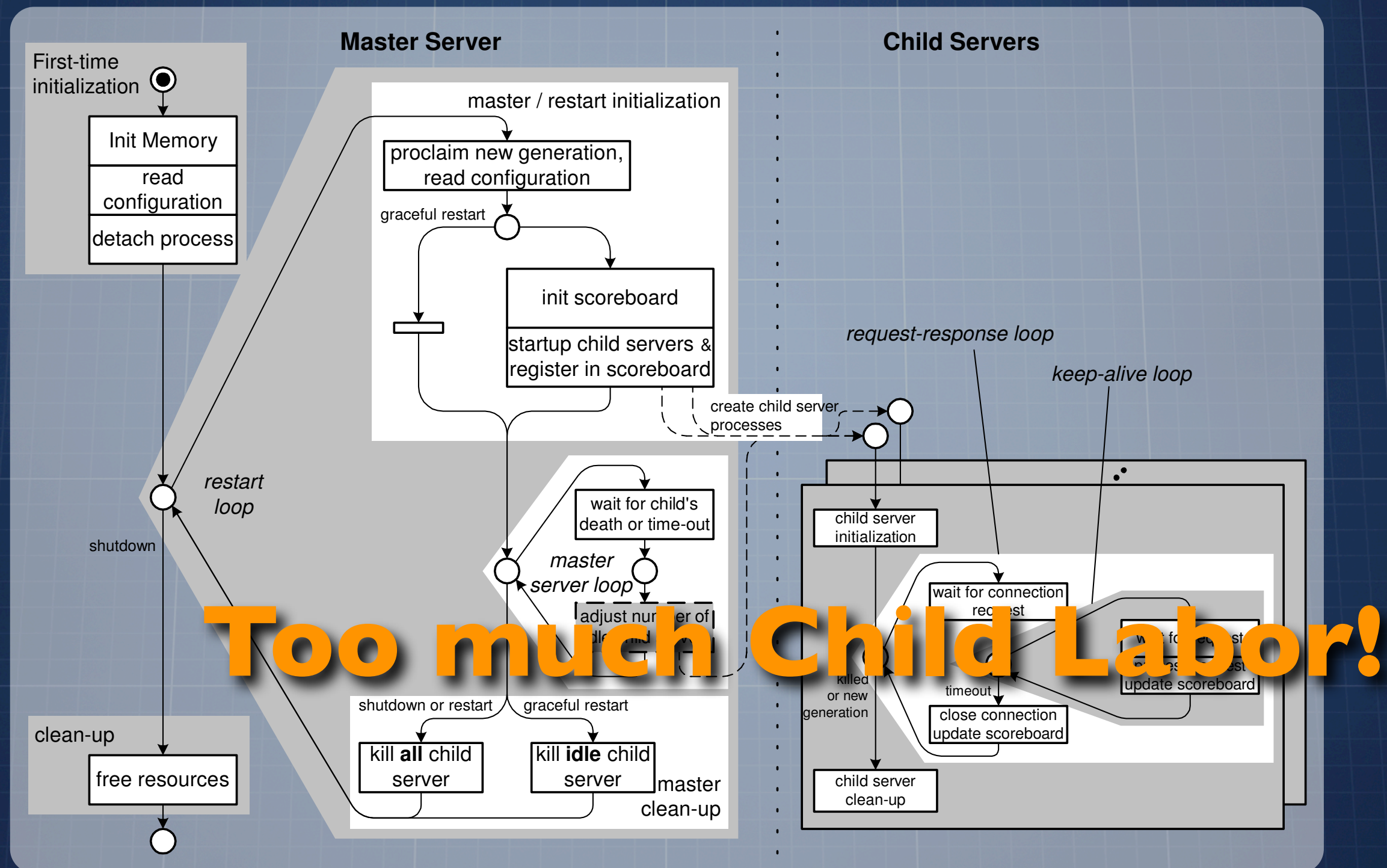
We call this the “spooning model”

Spooning Model



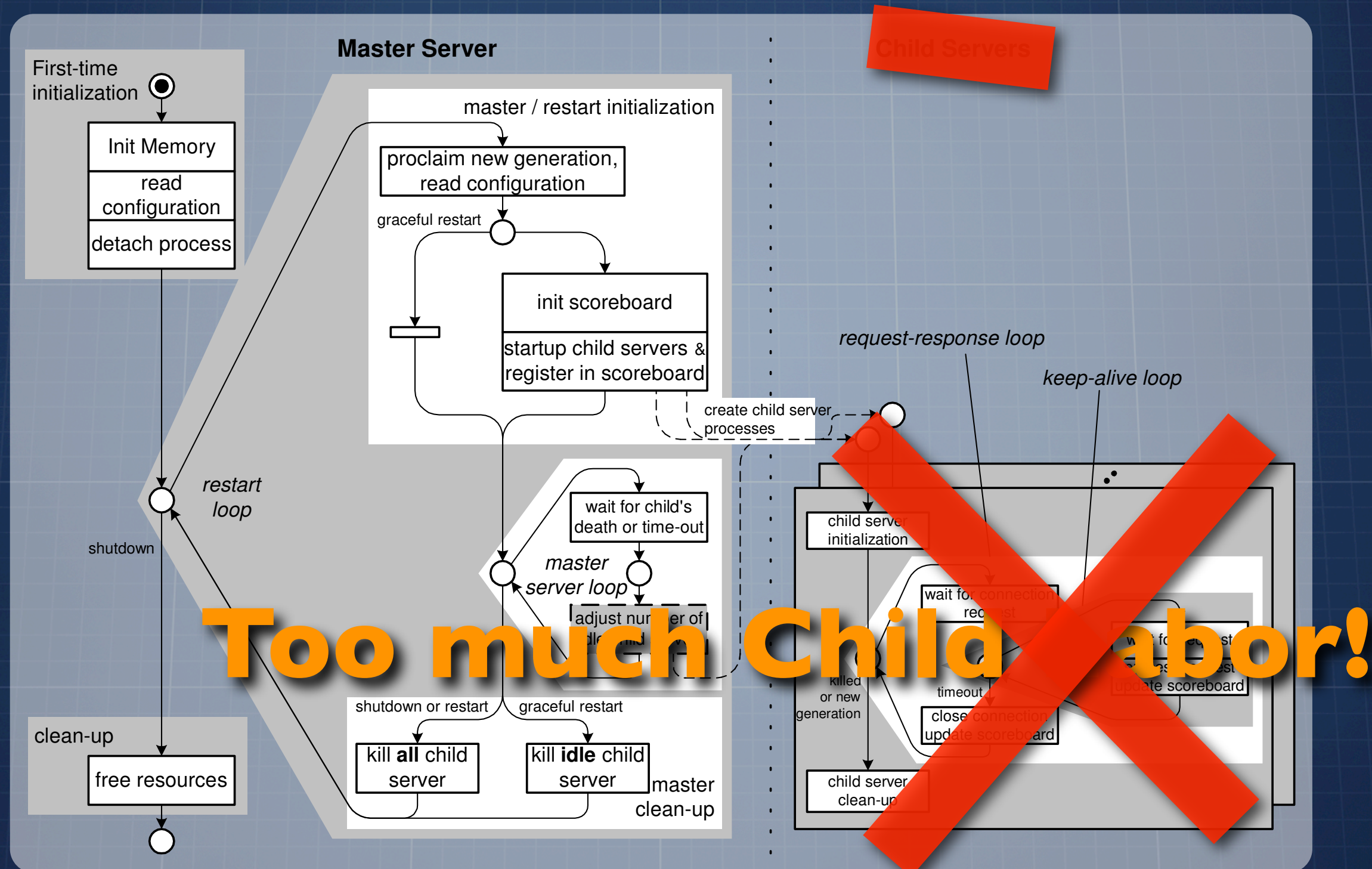
[HPI Apache Modeling Project, <http://apache.hpi.uni-potsdam.de/>]

Spooning Model



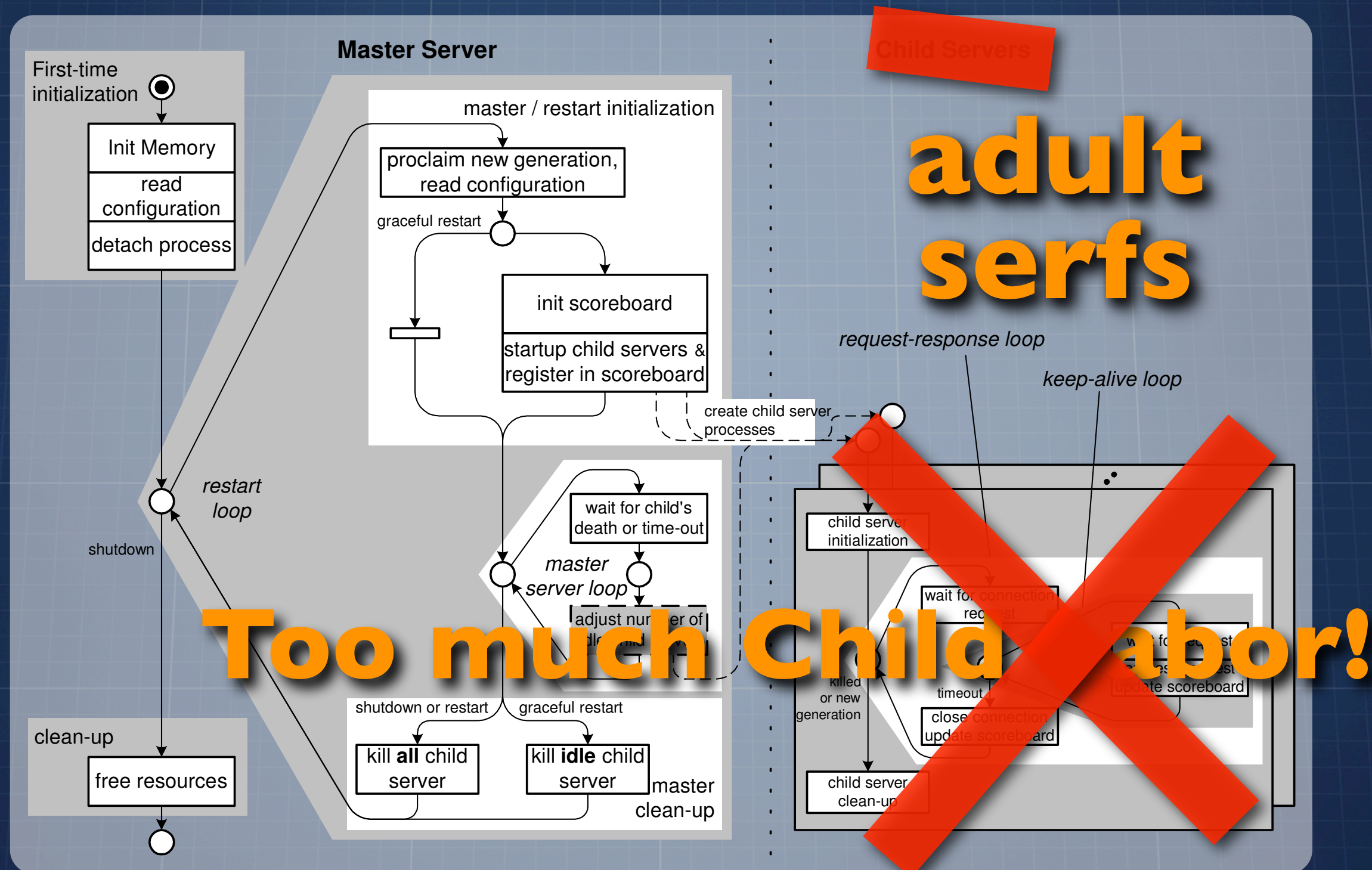
[HPI Apache Modeling Project, <http://apache.hpi.uni-potsdam.de/>]

Spooning Model



[HPI Apache Modeling Project, <http://apache.hpi.uni-potsdam.de/>]

Spooning Model



[HPI Apache Modeling Project, <http://apache.hpi.uni-potsdam.de/>]

Less Configuration

Default to SAFE and SIMPLE configuration

- no (default) configuration files
- no compile-time settings

All settings designed for PROGRAMMER TESTING

- start up using current owner and group
- listen to localhost:8080
- default to content/manual
- all other settings must be on command-line

httpd -c configuration.txt

httpsd -C directory

Waka

A new protocol designed to match the efficiency of well-designed Web Applications

Why “waka”?

- **Māori word (pronounced “wah-kah”) for the outrigger canoes used to travel safely on the Pacific Ocean, across hundreds of islands, to Aotearoa (New Zealand)**
- **Also, one of the few four-letter words suitable for a protocol name**

Deployable within an HTTP connection

- **via the HTTP/1.1 Upgrade header field**
- **defined mapping to HTTP/1.1 for proxies**

HTTP Syntax

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

...

HTTP Syntax

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

typical headers
325-400B +
cookies =
1079 Bytes

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

...

HTTP Syntax

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

typical headers
325-400B +
cookies =
1079 Bytes

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

typical headers
300-400B +
set-cookies =
3487 Bytes

...

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

...

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

Control Data

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

...

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

Control Data

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

Message Metadata

...

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

Control Data

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
```

Message Metadata

Representation
Metadata

```
4090\r\n
<HTML><HEAD>
```

...

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

Control Data

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
```

Message Metadata

Representation
Metadata

```
4090\r\n
<HTML><HEAD>
```

Resource
Metadata

...

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n

HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
...
```

Control Data

Message Metadata

Representation Metadata

Resource Metadata

Data

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

Control Data

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

Message Metadata

Representation Metadata

Resource Metadata

Data

...

Extensions cannot indicate scope or mandate

HTTP not self-descriptive

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
```

Control Data

\r\n Messages associated by order sent/received

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
```

Content-Type: text/html\r\n

Message Metadata

Content-Language: en\r\n

Transfer-Encoding: chunked\r\n

Etag: "a797cd-465af"\r\n

Representation

Cache-control: max-age=3600\r\n

Metadata

Vary: Accept-Language\r\n

\r\n

Resource

4090\r\n

<HTML><HEAD> Data

Metadata

...

Extensions cannot indicate scope or mandate

HTTP Syntax Wasted

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

...

HTTP Syntax Wasted

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

content
negotiation is a
waste of bits

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

...

HTTP Syntax Wasted

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

content
negotiation is a
waste of bits

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
```

Dates need 8
bytes max

Useless product
advertising

Header names
need 1-2 bytes

...

HTTP Syntax Wasted

```
GET /Test/hello.html HTTP/1.1\r\n
Host: kiwi.ics.uci.edu:8080\r\n
Accept: text/html, text/*, */*\r\n
User-Agent: GET/7 libwww-perl/5.40\r\n
\r\n
```

**content
negotiation is a
waste of bits**

```
HTTP/1.1 200 OK\r\n
Date: Thu, 09 Mar 2000 15:40:09 GMT\r\n
Server: Apache/1.3.12\r\n
Content-Type: text/html\r\n
Content-Language: en\r\n
Transfer-Encoding: chunked\r\n
Etag: "a797cd-465af"\r\n
Cache-control: max-age=3600\r\n
Vary: Accept-Language\r\n
\r\n
4090\r\n
<HTML><HEAD>
...
```

**Dates need 8
bytes max**

**Useless product
advertising**

**Header names
need 1-2 bytes**

**Mostly impacts low-power
and bandwidth-limited devices**

HTTP/2.x Syntax?

```
G http://kiwi.ics.uci.edu:8080/Test/hello.html
HTTP20\r\n
\r\n
```

```
HTTP20 200\r\n
DT:47FC71DD\r\n
CT:t/html\r\n
CL:en\r\n
TE:c\r\n
ET:"a797cd-465af"\r\n
CC:ma=3600\r\n
VY:AL\r\n
\r\n
4090\r\n
<HTML><HEAD>
...
```

Waka Syntax

```
waka22\r\n
QAG100#http://kiwi.ics.uci.edu:8080/Test/hello.html
SSakaw\r\n
```

```
waka22\r\n
RAG120#http://kiwi.ics.uci.edu:8080/Test/hello.html
D847FC71DDCT6t/htm1CL2enTEL1c0ETCa797cd-465afCCLmav36VYL2AL0
D2h4<HTML><HEAD>...
```

Unified Protocol and API

Unified Protocol and API

Protocols are the heart of a server

- Apache 1.0: HTTP/1.0
- Apache 1.2: HTTP/1.1
- Apache 2.0: HTTP/1.1, POP3, ECHO
- Apache 2.2: HTTP/1.1, POP2, ECHO, SMTP, FTP
- Apache 3.0: **waka**, HTTP/1.1

Unified Protocol and API

Protocols are the heart of a server

- Apache 1.0: HTTP/1.0
- Apache 1.2: HTTP/1.1
- Apache 2.0: HTTP/1.1, POP3, ECHO
- Apache 2.2: HTTP/1.1, POP2, ECHO, SMTP, FTP
- Apache 3.0: **waka**, HTTP/1.1

The request_rec is replaced with **waka_rec**

- waka messages are a tokenized data structure
- memcopy, validate, pass by reference
- coordinate using blackboard style, semaphores
- no more DIY request/response construction

Canned Error Responses

Canned Error Responses

Status codes and HTML aren't strong enough

- frequently overridden by stupid browsers
- often ignored by complacent users

This is the YouTube generation

- it's time for some fresh thinking!

Canned Error Responses

Status codes and HTML aren't strong enough

- frequently overridden by stupid browsers
- often ignored by complacent users

This is the YouTube generation

- it's time for some fresh thinking!

Samuel L. Jackson

- highest grossing actor
- over 100 movies
- translated worldwide



Canned Error Responses

Status codes and HTML aren't strong enough

- frequently overridden by stupid browsers
- often ignored by complacent users

This is the YouTube generation

- it's time for some fresh thinking!

Samuel L. Jackson

- highest grossing actor
- over 100 movies
- translated worldwide

401 Oh, Hell No!



Canned Error Responses

Status codes and HTML aren't strong enough

- frequently overridden by stupid browsers
- often ignored by complacent users

This is the YouTube generation

- it's time for some fresh thinking!

Samuel L. Jackson

- highest grossing actor
- over 100 movies
- translated worldwide

401 Oh, Hell No!

404 Where's... my... super suit!



Canned Error Responses

Status codes and HTML aren't strong enough

- frequently overridden by stupid browsers
- often ignored by complacent users

This is the YouTube generation

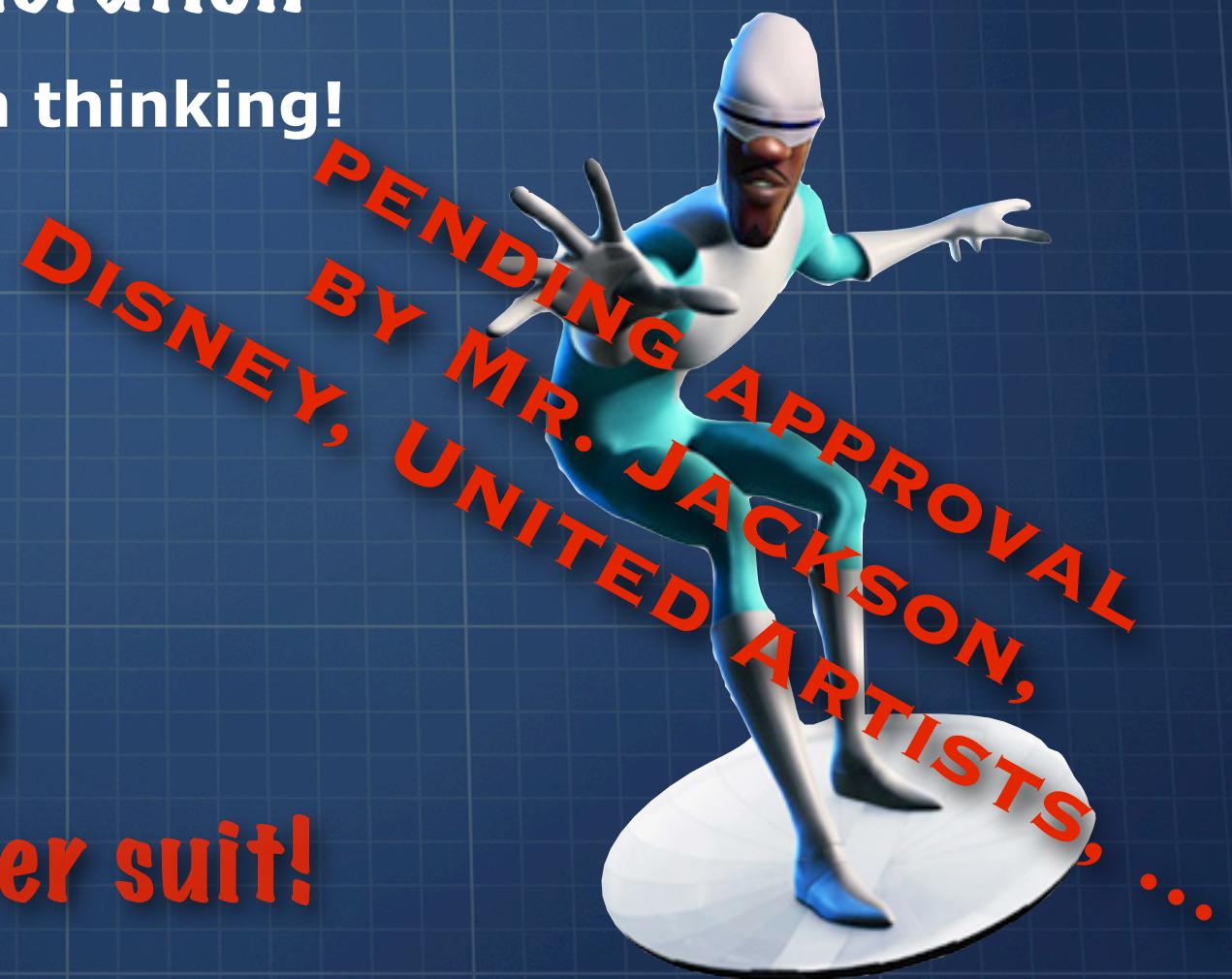
- it's time for some fresh thinking!

Samuel L. Jackson

- highest grossing actor
- over 100 movies
- translated worldwide

401 Oh, Hell No!

404 Where's... my... super suit!



Summary

Summary

**Apache 3.0
is our best product
ever!**

[aside from some minor implementation details]

Outline

a bit of history

what does a web server do

introducing Apache 3.0

the nature of collaboration

Collaboration

Collaboration

Somebody always leads

- we can take turns leading
only if we also take turns following**

Collaboration

Somebody always leads

- we can take turns leading
only if we also take turns following

Apache doesn't select the leaders

- leaders are defined by what they do
and how enticing they are to follow

Collaboration

Somebody always leads

- we can take turns leading
only if we also take turns following

Apache doesn't select the leaders

- leaders are defined by what they do
and how enticing they are to follow

**If you don't like where we're heading ...
maybe it's time for you to lead!**

Collaboration

Somebody always leads

- we can take turns leading
only if we also take turns following

Apache doesn't select the leaders

- leaders are defined by what they do
and how enticing they are to follow

**If you don't like where we're heading ...
maybe it's time for you to lead!**

See you on <dev@httpd.apache.org>