APACHE 2

Multi-process, multi-threaded, or both?

LinuxTag 2004

Erik Abele

The Apache Software Foundation
http://www.apache.org/
Apache History

- 1994 NCSA HTTPd
- 1995 A "patchy server" is born
  - April: Apache 0.6.2 - first public release
  - December: Apache 1.0
- 1997 Apache 1.2.0
- 1998 Apache 1.3.0
- 1999 Incorporation of the ASF
- 2000 Apache 2.0 Alpha 1
- 2004 Apache 2.1 Beta 1?
Apache 2: New Features

- Based on the Apache Portable Runtime
- MPMs (Multi-processing modules)
- Filtering, IPv6 and Multi-protocol support
- Built-in SSL and improved Authn/Authz mechanisms (e.g. mod_auth_ldap)
- Module improvements
  - New: mod_dav, mod_deflate, mod_logio, ...
  - Improved: mod_include, mod_negotiation, ...
- Out-of-the-box XHTML-compliant, multi-language error responses
- Drastically improved module API
- Active development
Apache Portable Runtime

- Used by Apache HTTPD, Subversion, Flood, Prothon and other projects
- Consistent interface to underlying platform-specific implementations
- Platforms are implemented in their native API instead of using the POSIX-emulation layers
- Solid foundation for Linux, Unix and non-Unix platforms such as BeOS, OS/2 and Windows
Multi-processing modules

- An MPM defines how the server will receive and manage incoming requests:
  - Different HTTP server process models (e.g. threaded, multi-process based or hybrid)
  - Platform- & OS-specific optimizations (e.g. Windows, BeOS, NetWare, OS/2)
  - OS-specific features (e.g. Sendfile, AcceptEx)
  - Admin can choose: Reliability vs. Scalability vs. Performance vs. Features
  - More efficient ways of controlling the server (resource limits, thread/process ratio)
  - Extendable with third-party MPMs
Prefork

- Each child handles one connection at a time: much traffic, many children :)
  - High memory requirements
  - Highly tolerant of faulty modules
  - Highly tolerant of crashing children
  - Fast
  - Well-suited for 1 and 2-CPU systems
  - Tried-and-tested model from Apache 1.3
  - "You'll run out of memory before CPU"
Prefork model

- Each child handles one connection at a time: many children are needed
Worker

- Multi-threaded within each child, each thread handles a single connection:
  - Low to moderate memory footprint
  - Moderately tolerant to faulty modules
  - Faulty threads can affect all threads in a child
  - Fast and highly scalable
  - Well-suited for multiple processors
  - Requires a mature threading library (Solaris, AIX, Linux 2.6 and others work well)
  - Memory is no longer the bottleneck
Worker model

- Multi-threaded within each child: only a few children are needed
Other MPMs

- OS-specific MPMs:
  - WinNT
  - OS/2
  - BeOS
  - NetWare
- Perchild (experimental)
- Leader-Follower (experimental)
- Threadpool (experimental)
- Third-party MPMs: Metux-MPM
Choosing an MPM

• Multi-process, multi-threaded, or both?
• Compile-time decision
• Depends on a variety of factors:
  – Does the OS support threads?
  – Scalability vs. Stability?
  – Are third-party modules with unknown, and possibly thread-unsafe extensions (e.g. PHP, mod_perl) used?
  – How much memory is available?
  – ...
Know your server

- Remove configuration defaults and examples
- Disable unused modules
  - saves memory
  - reduces some processing
  - 'enhances' security
- If something goes wrong, the first place to look is always the error_log
- Temporarily increase the LogLevel if needed
- mod_status shows what Apache is doing
- [http://httpd.apache.org/docs-2.0/](http://httpd.apache.org/docs-2.0/)
Keeping up to date

- Apache website and Announcement list
  - http://httpd.apache.org/
  - announce-subscribe@httpd.apache.org
- ApacheWeek
  - http://www.apacheweek.com/
- Vendor package updates
- CERT CC, BugTraq, Full Disclosure List
That's it!

Thanks for listening!

More info and the slides are available at
http://www.apache.org/~erikabele/

You can reach me at
erikabele@apache.org