

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATHEMATISCH-
NATURWISSENSCHAFTLICHE FAKULTÄT
WILHELM-SCHICKARD-INSTITUT FÜR INFORMATIK
INFORMATIONSDIENSTE

Grundlagen der Web-Entwicklung

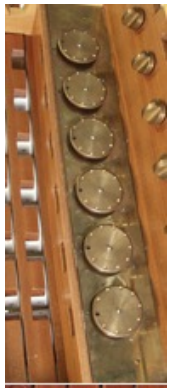
INF3172

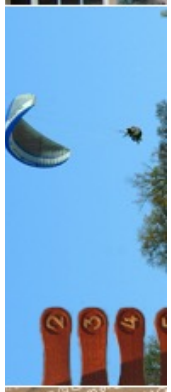
Der Apache Web-Server Teil II

Thomas Walter

05.11.2026

Version 1.0

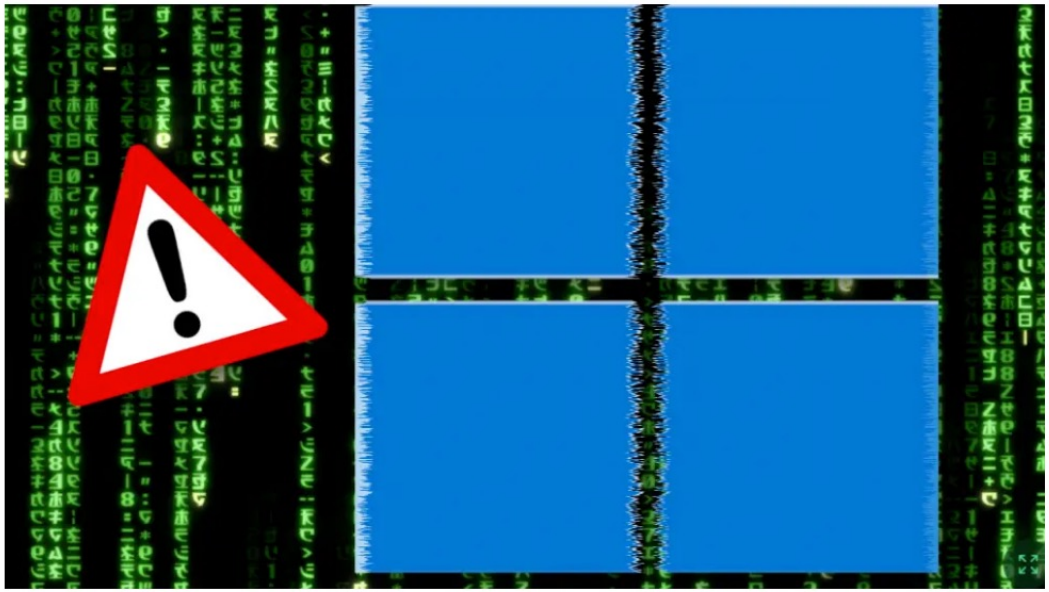





Microsoft gibt Tipps für erweiterten Support für kommerzielles Windows 10

Windows 10 erhält seit dem 14. Oktober keinen Support mehr. Microsoft gibt Admins Tipps für kommerziellen, erweiterten Support.

32



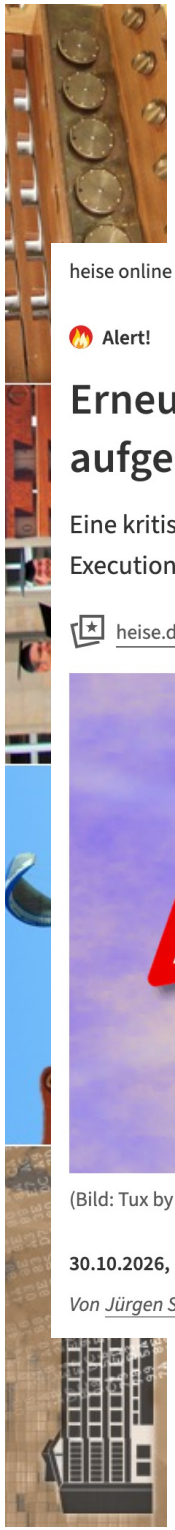
(Bild: heise online / dmk)

09:06 Uhr Lesezeit: 3 Min. | Security

Von Dirk Knop

Inzwischen sollte es sattsam bekannt sein: Microsoft hat den Support für Windows 10 offiziell zum 14. Oktober 2025 eingestellt. Privatnutzer in der EU bekommen nach langem Hin und Her ein Jahr kostenlos erweiterten Support (Extended Security Updates, ESU), wenn sie sich dafür anmelden. Auch in Organisationen müssen IT-Verantwortliche aktiv werden, damit die Windows-10-Geräte weiterhin Sicherheitsupdates erhalten. Microsoft gibt dafür nun Tipps und Hinweise.

© 2026 Universität Tübingen



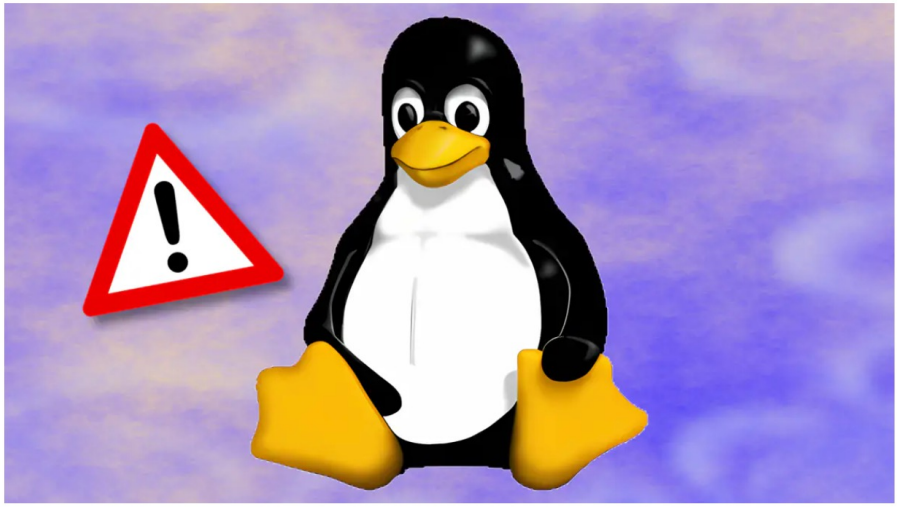
heise online > Betriebssystem > Linux > Erneut kritischer Bug in Linux TCP/IP aufgedeckt – Updates ir

Alert!

Erneut kritischer Bug in Linux TCP/IP aufgedeckt – Updates in Arbeit

Eine kritische Lücke im TCP/IP-Stack des Linux-Kerns erlaubt Remote Code Execution. Anthropics Claude entdeckte den elf Jahre alten Fehler.

heise.de bei Google folgen 164



(Bild: Tux by Larry Ewing/GIMP)

30.10.2026, 13:12 Uhr **Lesezeit: 4 Min.** | **Security**

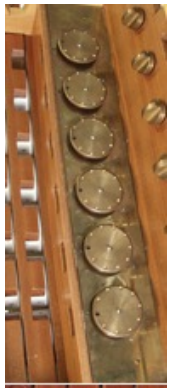
Von Jürgen Schmidt

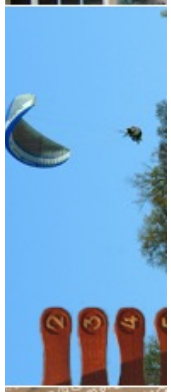
Eine kritische Lücke im Netzwerk-Code des Linux-Kerns betrifft nahezu alle Linux-Distributionen, darunter Debian, Red Hat, Suse und Ubuntu. Es handelt sich um ein Problem im TCP/IP-Stack, das Angreifer übers Netz ausnutzen können, um eigenen Code einzuschleusen und auszuführen (Remote Code Execution, CVSS 9,8 und somit allerhöchste Gefahrenstufe). Es gibt jedoch bereits einen Fix des Kernel-Teams; erste Distributoren haben bereits Updates veröffentlicht.

Anthropics Claude deckte diese kritische Sicherheitslücke erst kürzlich bei systematischen Tests wichtiger Projekte auf. Sie schlummerte seit ziemlich genau elf Jahren unbemerkt im Code des Linux-Kerns, den Millionen von Systemen nutzen. Es handelt sich dabei um eine Race-Condition, bei der ein Angreifer den Linux-Kernel seines Opfers dazu bringen kann, auf einer CPU Speicher zu verwenden, den kurz zuvor eine andere CPU bereits freigegeben hatte. Dies führt zu nicht vorhergesehenen Speicherzugriffen, die der Angreifer so koordinieren kann, dass von ihm eingeschleuster Code ausgeführt wird. Das konnte Claude mit einem konkreten Exploit beweisen und Anthropic meldete den Fehler dem Linux-Kernel-Team.

Der Fix nach elf Jahren

Der Commit des fehlerhaften Codes erfolgte am 2. Oktober 2015 in Kernel-Version 4.4 und wurde von mehreren Experten kontrolliert und freigegeben. Erst die systematische Suche einer KI förderte ihn zutage. Am 15. Oktober, also elf Jahre nach dem Commit, bestätigte Greg Kroah-Hartman, dass der Fehler CVE-2026-98432 in den Kernel-Versionen 6.12.113, 6.18.56, 7.2.10 und 7.3-rc7 gefixt wurde.



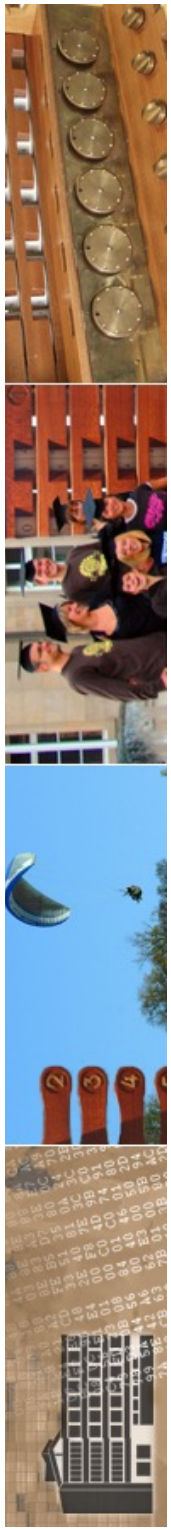



das Weihnachtsrätsel

- am **1. Advent 29.12.2024 ab 11.00h:**
  das Weihnachtsrätsel!



- Preise für schnellste Lösung und Verlosung weiterer
- Auflösung und Verlosung der Gewinne in der Vorlesung vor Weihnachten

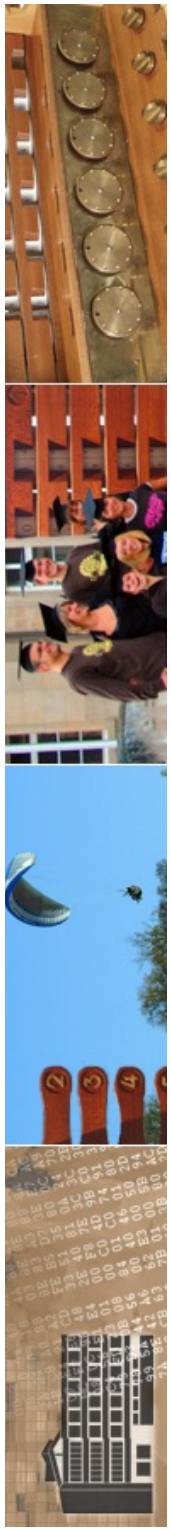


bisher

- wir haben in Teil I den Apache-Webserver grundlegend kennen gelernt
  - Installation
  - Konfiguration

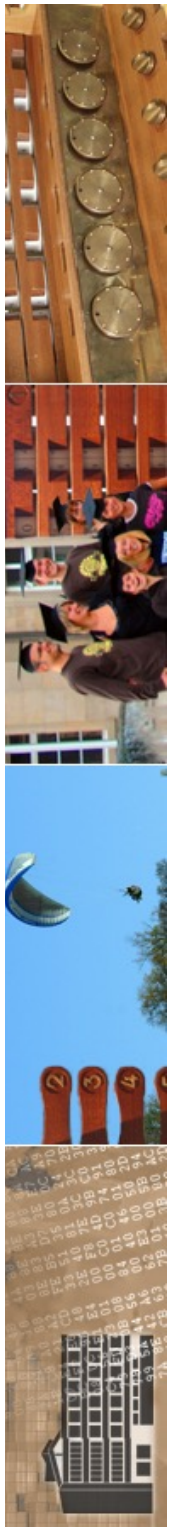
- nun: wir wollen Apache weiter an unsere Anforderungen anpassen:

  Apache-Module



Apache-Module

- Apache ist *modular* aufgebaut
  - Funktionalität ist in Modulen gebündelt
  - Module erweitern den Apache Webserver
- je mehr Module desto
  - langsamer
  - unsicherer
- wird der Web-Server
  - **"weniger ist mehr"**

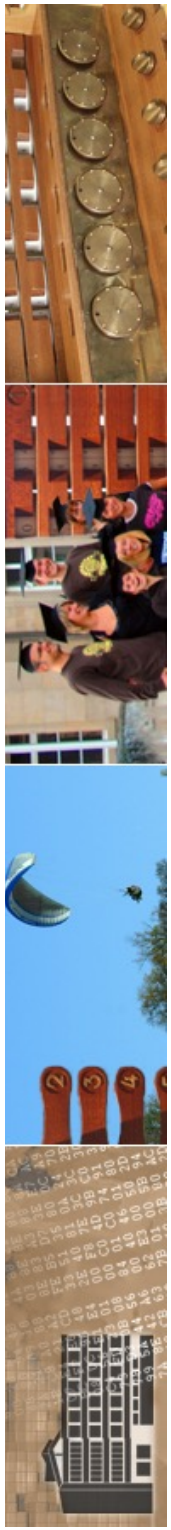


Arten von Modulen

- Module werden unterschieden
  - Module der ASF
    - werden mit dem Apache ausgeliefert
    - davon werden nur die "Standardmudule" per default installiert
  - Drittmodule
    - z.B. PHP



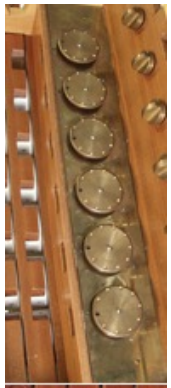
- statische und dynamische Module



statische und dynamische Module

- statische und dynamische Module
  - statische werden in den Apache Kern hineincompiliert und sind immer verfügbar
  - dynamische (so/dll) werden zur Laufzeit (beim Starten des Apache) gebunden

- statische sind schneller (insbesondere beim Start)
- dynamische sind flexibler
  - bei Update des Moduls muss *nur dieses* compiliert werden und nicht der ganze Web-Server
  - Drittmodule sind dynamisch





welche Module sind aktiv?

- Modulübersicht der **statischen** Module über

```
httpd -l
apachectl -l
```

134.2.2.38 - Internet-Prog SS 09 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

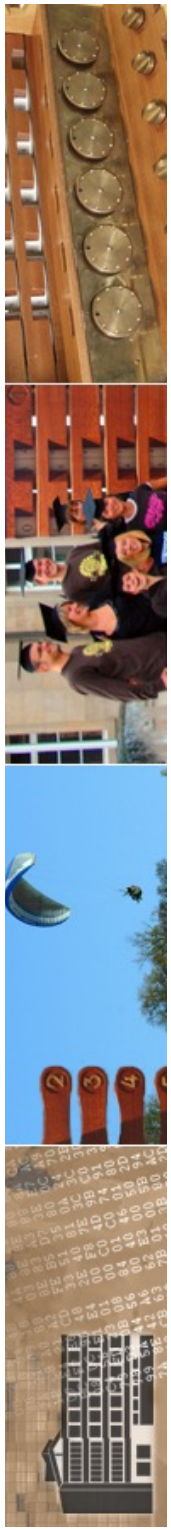
```
zrvwa01@infodienste:~/apache_test/bin$
zrvwa01@infodienste:~/apache_test/bin$ ./httpd -l
Compiled in modules:
  core.c
  mod_authn_file.c
  mod_authn_default.c
```

```
[thomas@Vaux =>
[thomas@Vaux => ./httpd -l
 Compiled in modules:
   core.c
   mod_so.c
   http_core.c
   prefork.c
[thomas@Vaux =>
```

DOS Shell

```
C:\Program Files\Apache Software Foundation\Apache24\bin>
C:\Program Files\Apache Software Foundation\Apache24\bin>
C:\Program Files\Apache Software Foundation\Apache24\bin>.\httpd.exe -l
Compiled in modules:
  core.c
  mod_win32.c
  mpm_winnt.c
  http_core.c
  mod_so.c
```

- aes128-cbc - hmac-m

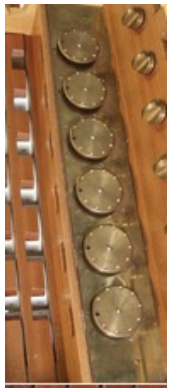


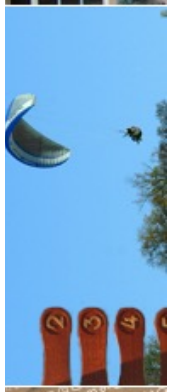
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN



thomas — ssh ◂ -bash — 68×10

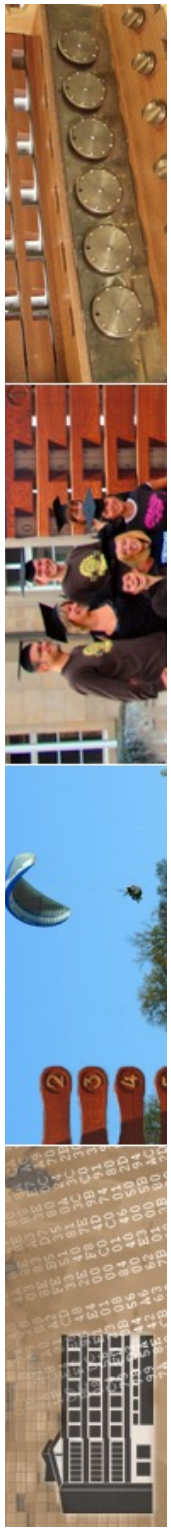
```
zrvwa01@infodienste =>
zrvwa01@infodienste =>
zrvwa01@infodienste => ./apachectl -l
Compiled in modules:
  core.c
  mod_so.c
  http_core.c
  event.c
zrvwa01@infodienste =>
zrvwa01@infodienste =>
```





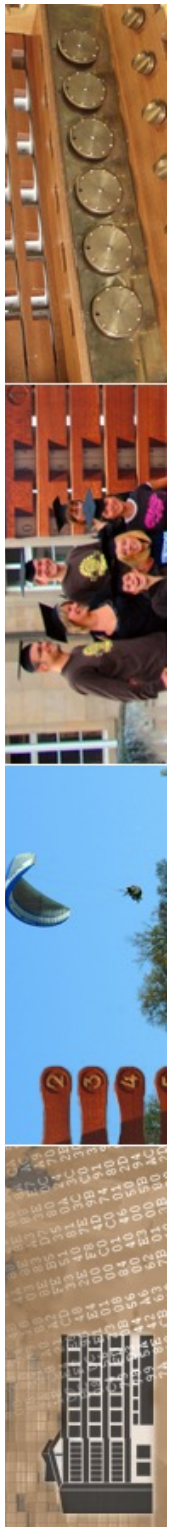

laden statischer Module

- dies wird bereits vor dem Compilieren (make) durch das Skript **`configure`** festgelegt

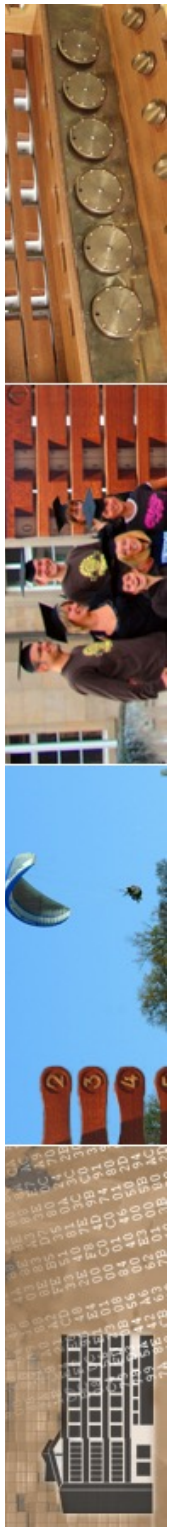


Laden dynamischer Module

- die dynamischen Module werden durch die *zentrale Konfiguration http.conf* geladen
- Direktive **`LoadModule`**

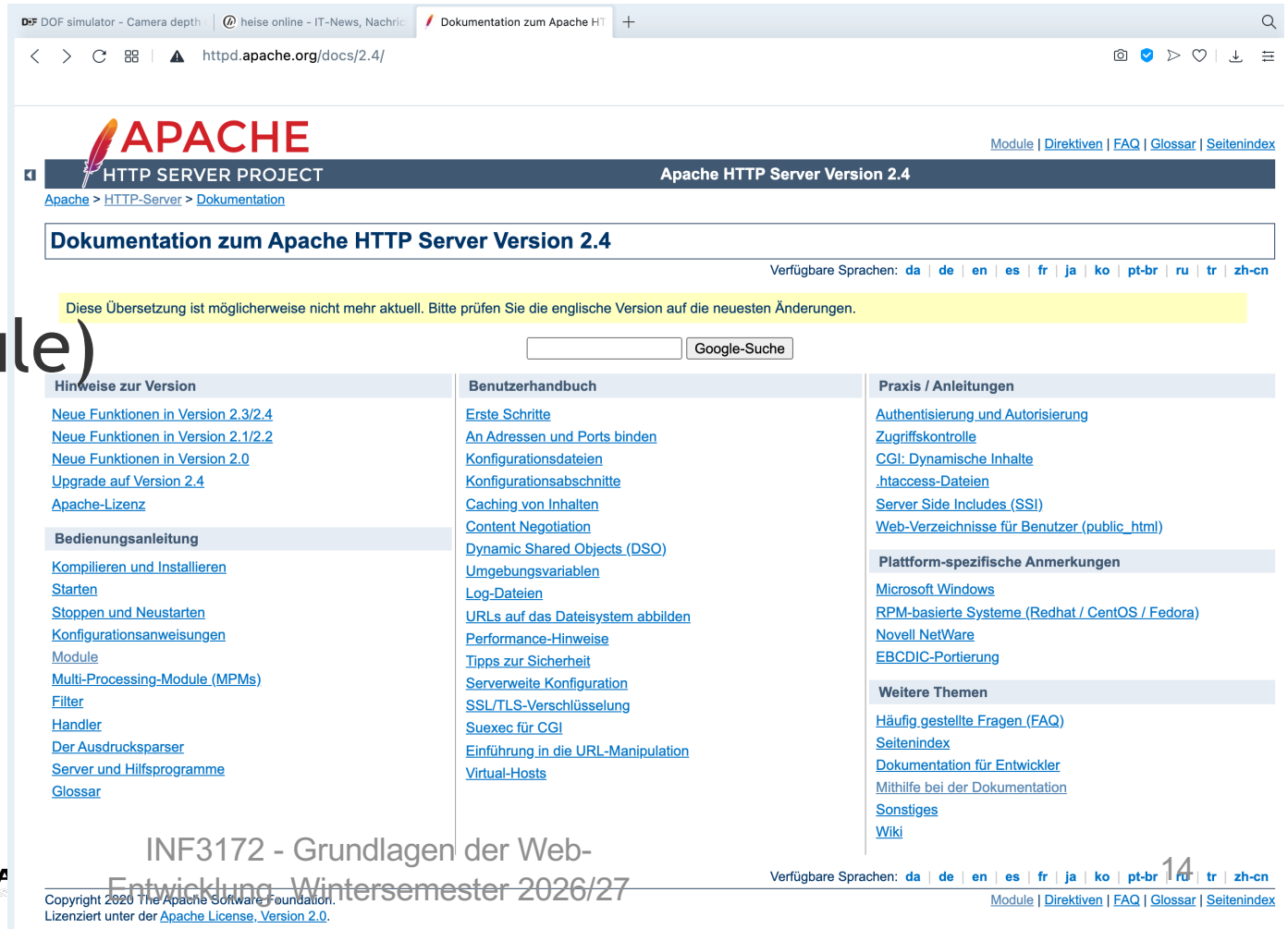


```
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule authn_file_module modules/mod_authn_file.so
#LoadModule authn_dbm_module modules/mod_authn_dbm.so
#LoadModule authn_anon_module modules/mod_authn_anon.so
#LoadModule authn_dbd_module modules/mod_authn_dbd.so
#LoadModule authn_socache_module modules/mod_authn_socache.so
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_user_module modules/mod_authz_user.so
#LoadModule authz_dbm_module modules/mod_authz_dbm.so
#LoadModule authz_owner_module modules/mod_authz_owner.so
#LoadModule authz_dbd_module modules/mod_authz_dbd.so
LoadModule authz_core_module modules/mod_authz_core.so
#LoadModule authnz_ldap_module modules/mod_authnz_ldap.so
LoadModule access_compat_module modules/mod_access_compat.so
LoadModule auth_basic_module modules/mod_auth_basic.so
#LoadModule auth_form_module modules/mod_auth_form.so
#LoadModule auth_digest_module modules/mod_auth_digest.so
#LoadModule allowmethods_module modules/mod_allowmethods.so
#LoadModule file_cache_module modules/mod_file_cache.so
#LoadModule cache_module modules/mod_cache.so
#LoadModule cache_disk_module modules/mod_cache_disk.so
#LoadModule cache_socache_module modules/mod_cache_socache.so
#LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
#LoadModule socache_dbm_module modules/mod_socache_dbm.so
#LoadModule socache_memcache_module modules/mod_socache_memcache.so
#LoadModule watchdog_module modules/mod_watchdog.so
```



Übersicht über Module

- die Apache-Online-Dokumentation gibt eine Übersicht über die Apache Module (ASF-Module)



EBERHARD KARLS

Modul-Index

Verfügbare Sprachen: de | en | es | fr | ja | ko | tr | zh-cn

Nachfolgend ist eine Liste aller Module angegeben, die als Bestandteil der Distribution des Apache HTTP Servers mitgeliefert werden. Bitte beachten Sie auch die vollständige alphabetische Liste aller Apache-HTTP-Server-Direktiven.

- Kernfunktionen und Multi-Processing-Module
- Andere Module

Siehe auch

- Multi-Processing-Module (MPMs)
- Kurzreferenz der Direktiven

Kernfunktionen und Multi-Processing-Module

core
Ständig verfügbare Kernfunktionen des Apache HTTP Servers

mpm_common
Eine Sammlung von Direktiven, die in mehr als einem Multi-Processing-Modul (MPM) implementiert sind.

event
A variant of the `worker` MPM with the goal of consuming threads only for connections with active processing

mpm_netware
Multi-Processing Module implementing an exclusively threaded web server optimized for Novell NetWare

mpmt_os2
Hybrid multi-process, multi-threaded MPM for OS/2

prefork
Implementiert einen im Voraus forkenden Webserver ohne Thread-Unterstützung

mpm_winnt
Das Multi-Processing-Modul ist optimiert für Windows NT.

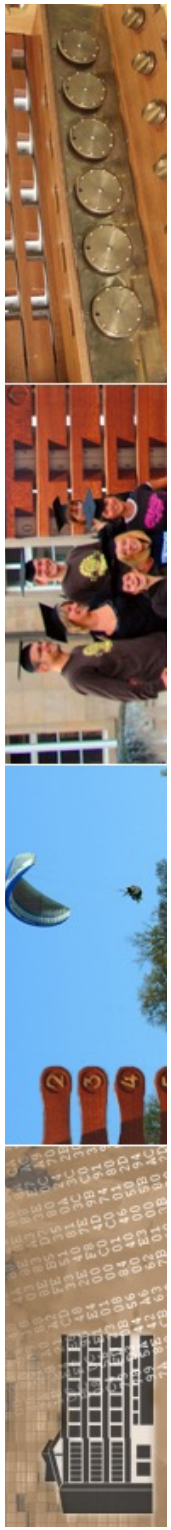
worker
Multi-Processing-Modul, das einen Hybrid-Webserver mit Multi-Thread und Multi-Prozess-Unterstützung implementiert

Andere Module

A | B | C | D | E | F | H | I | L | M | N | P | R | S | U | V | W | X

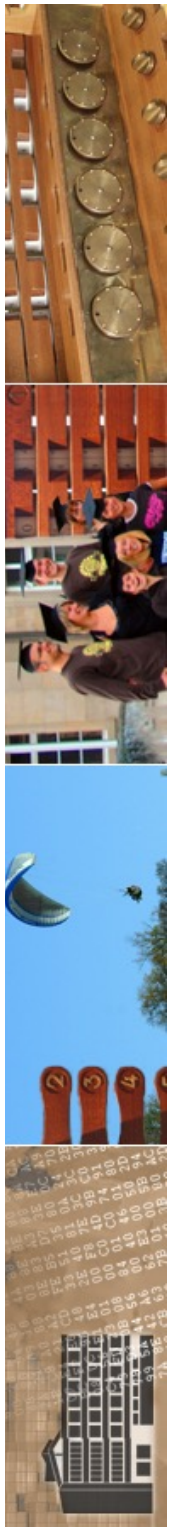
mod_access_compat
Group authorizations based on host (name or IP address)

mod_actions
Dieses Modul ermöglicht die Ausführung von CGI-Skripten in Abhängigkeit von Medientypen und Anfragemethoden.



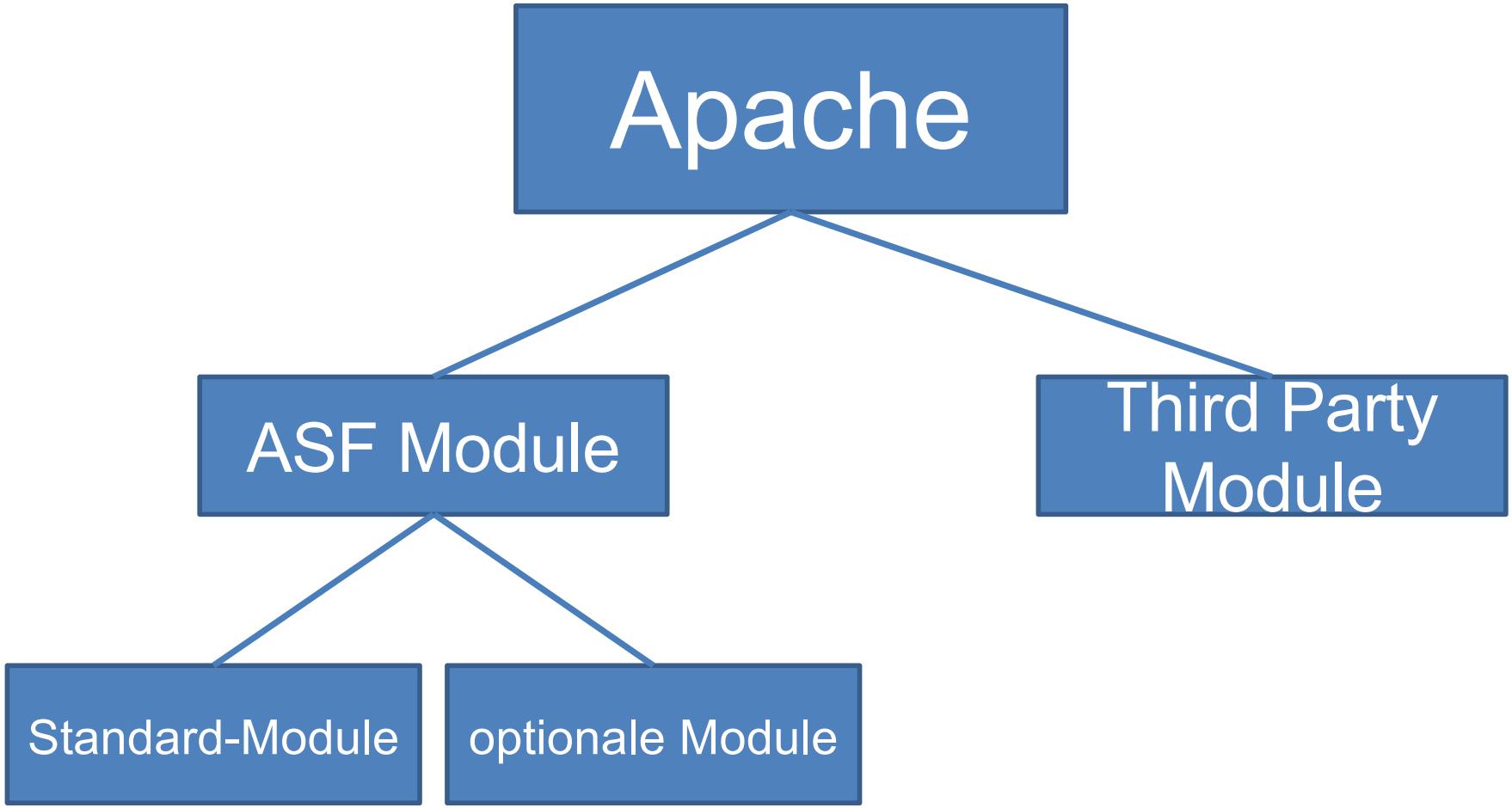
Klassifizierung von Modulen

- Module der ASF
  - Standard-Module: werden mit Default-Konfiguration installiert
  - weitere ASF-Module
- Module von Drittanbietern
  - werden typischerweise dynamisch installiert



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Klassifizierung von Modulen



```
thomas@PetitMouton-2 => ./configure --help
`configure' configures this package to adapt to many kinds of systems.

Usage: ./configure [OPTION]... [VAR=VALUE]...

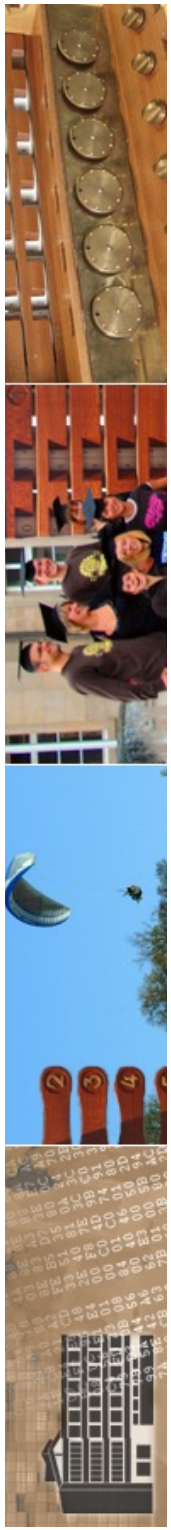
To assign environment variables (e.g., CC, CFLAGS...), specify them as
VAR=VALUE.  See below for descriptions of some of the useful variables.

Defaults for the options are specified in brackets.

Configuration:
  -h, --help              display this help and exit
      --help=short        display options specific to this package
      --help=recursive    display the short help of all the included packages
  -V, --version           display version information and exit
  -q, --quiet, --silent   do not print `checking ...' messages
      --cache-file=FILE   cache test results in FILE [disabled]
  -C, --config-cache      alias for `--cache-file=config.cache'
  -n, --no-create         do not create output files
      --srcdir=DIR        find the sources in DIR [configure dir or `..']

Installation directories:
  --prefix=PREFIX         install architecture-independent files in PREFIX
                          [/usr/local/apache2]
  --exec-prefix=EPREFIX   install architecture-dependent files in EPREFIX
                          [PREFIX]

By default, `make install' will install all the files in
`/usr/local/apache2/bin', `/usr/local/apache2/lib' etc.  You can specify
an installation prefix other than `/usr/local/apache2' using `--prefix',
for instance `--prefix=$HOME'.
```

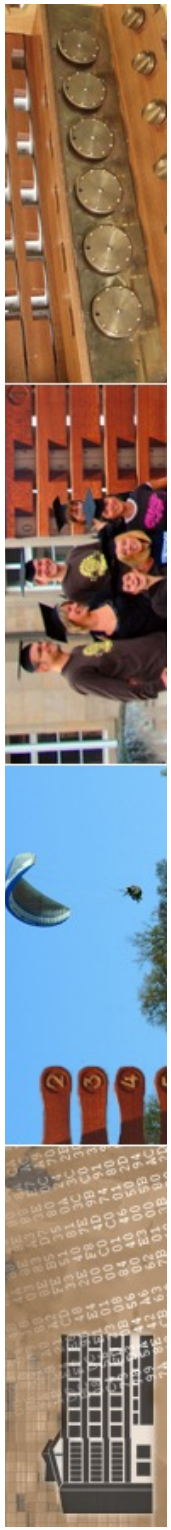


```
For better control, use the options below.

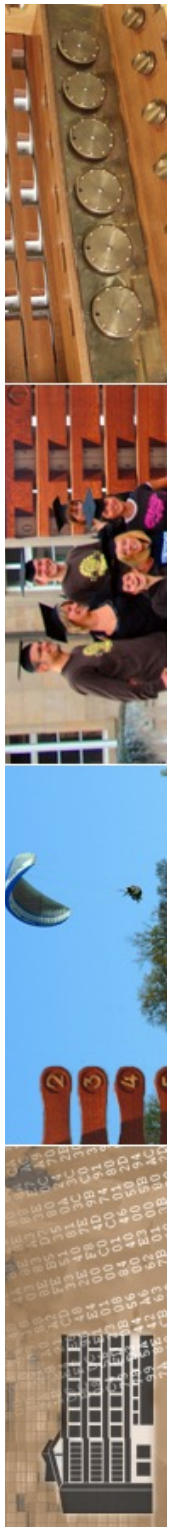
Fine tuning of the installation directories:
  --bindir=DIR            user executables [EPREFIX/bin]
  --sbindir=DIR           system admin executables [EPREFIX/sbin]
  --libexecdir=DIR        program executables [EPREFIX/libexec]
  --sysconfdir=DIR        read-only single-machine data [PREFIX/etc]
  --sharedstatedir=DIR    modifiable architecture-independent data [PREFIX/com]
  --localstatedir=DIR     modifiable single-machine data [PREFIX/var]
  --libdir=DIR            object code libraries [EPREFIX/lib]
  --includedir=DIR        C header files [PREFIX/include]
  --oldincludedir=DIR     C header files for non-gcc [/usr/include]
  --datarootdir=DIR       read-only arch.-independent data root [PREFIX/share]
  --datadir=DIR           read-only architecture-independent data [DATAROOTDIR]
  --infodir=DIR           info documentation [DATAROOTDIR/info]
  --localedir=DIR         locale-dependent data [DATAROOTDIR/locale]
  --mandir=DIR            man documentation [DATAROOTDIR/man]
  --docdir=DIR            documentation root [DATAROOTDIR/doc/PACKAGE]
  --htmldir=DIR           html documentation [DOCDIR]
  --dvidir=DIR            dvi documentation [DOCDIR]
  --pdfdir=DIR            pdf documentation [DOCDIR]
  --psdir=DIR             ps documentation [DOCDIR]

System types:
  --build=BUILD     configure for building on BUILD [guessed]
  --host=HOST       cross-compile to build programs to run on HOST [BUILD]
  --target=TARGET   configure for building compilers for TARGET [HOST]

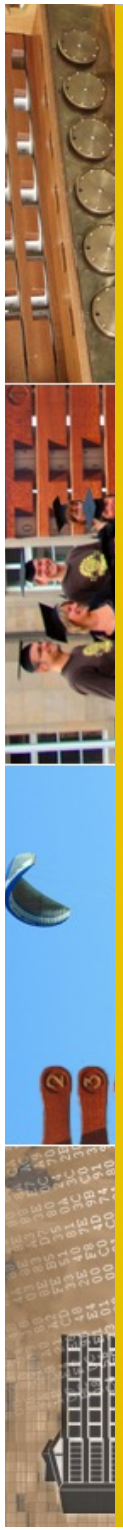
Optional Features:
  --disable-option-checking  ignore unrecognized --enable/--with options
  --disable-FEATURE       do not include FEATURE (same as --enable-FEATURE=no)
  --enable-FEATURE[=ARG]  include FEATURE [ARG=yes]
  --enable-layout=LAYOUT
  --enable-dtrace         Enable DTrace probes
  --enable-hook-probes    Enable APR hook probes
  --enable-exception-hook Enable fatal exception hook
  --enable-load-all-modules
                          Load all modules
  --enable-maintainer-mode
                          Turn on debugging and compile time warnings and load
                          all compiled modules
```



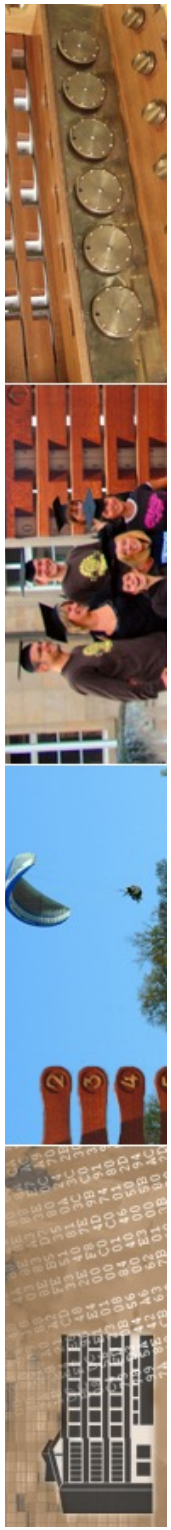
```
--enable-debugger-mode  Turn on debugging and compile time warnings and turn
                        off optimization
--enable-pie            Build httpd as a Position Independent Executable
--enable-modules=MODULE-LIST
                        Space-separated list of modules to enable | "all" |
                        "most" | "few" | "none" | "reallyall"
--enable-mods-shared=MODULE-LIST
                        Space-separated list of shared modules to enable |
                        "all" | "most" | "few" | "reallyall"
--enable-mods-static=MODULE-LIST
                        Space-separated list of static modules to enable |
                        "all" | "most" | "few" | "reallyall"
--disable-authn-file    file-based authentication control
--enable-authn-dbm      DBM-based authentication control
--enable-authn-anon     anonymous user authentication control
--enable-authn-dbd      SQL-based authentication control
--enable-authn-socache  Cached authentication control
--disable-authn-core    core authentication module
--disable-authz-host    host-based authorization control
--disable-authz-groupfile
                        'require group' authorization control
--disable-authz-user    'require user' authorization control
--enable-authz-dbm      DBM-based authorization control
--enable-authz-owner    'require file-owner' authorization control
--enable-authz-dbd      SQL based authorization and Login/Session support
--disable-authz-core    core authorization provider vector module
--enable-authnz-ldap    LDAP based authentication
--enable-authnz-fcgi    FastCGI authorizer-based authentication and
                        authorization
--disable-access-compat mod_access compatibility
--disable-auth-basic    basic authentication
--enable-auth-form      form authentication
--enable-auth-digest    RFC2617 Digest authentication
--enable-allowmethods   restrict allowed HTTP methods
--enable-isapi          isapi extension support
--enable-file-cache     File cache
--enable-cache          dynamic file caching. At least one storage
                        management module (e.g. mod_cache_disk) is also
                        necessary.
--enable-cache-disk     disk caching module
--enable-cache-socache  shared object caching module
```



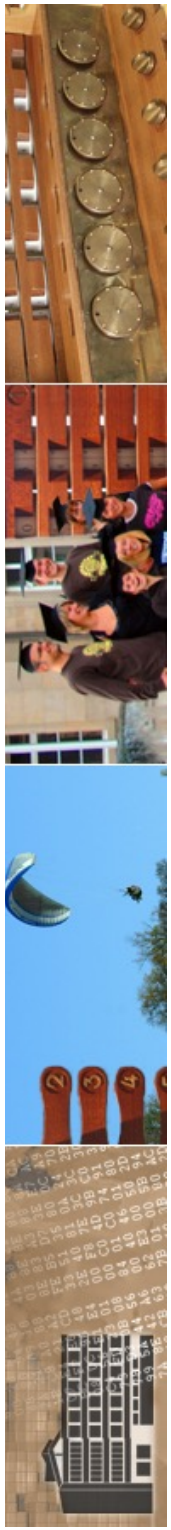
```
--enable-socache-shmcb  shmcb small object cache provider
--enable-socache-dbm    dbm small object cache provider
--enable-socache-memcache
                        memcache small object cache provider
--enable-socache-dc     distcache small object cache provider
--enable-so             DSO capability. This module will be automatically
                        enabled unless you build all modules statically.
--enable-watchdog       Watchdog module
--enable-macro          Define and use macros in configuration files
--enable-dbd            Apache DBD Framework
--enable-bucketeer      buckets manipulation filter. Useful only for
                        developers and testing purposes.
--enable-dumpio         I/O dump filter
--enable-echo           ECHO server
--enable-example-hooks  Example hook callback handler module
--enable-case-filter    Example uppercase conversion filter
--enable-case-filter-in Example uppercase conversion input filter
--enable-example-ipc    Example of shared memory and mutex usage
--enable-buffer         Filter Buffering
--enable-data           RFC2397 data encoder
--enable-ratelimit      Output Bandwidth Limiting
--disable-reqtimeout    Limit time waiting for request from client
--enable-ext-filter     external filter module
--enable-request        Request Body Filtering
--enable-include        Server Side Includes
--disable-filter        Smart Filtering
--enable-reflector      Reflect request through the output filter stack
--enable-substitute     response content rewrite-like filtering
--enable-sed            filter request and/or response bodies through sed
--disable-charset-lite  character set translation. Enabled by default only
                        on EBCDIC systems.
--enable-charset-lite   character set translation. Enabled by default only
                        on EBCDIC systems.
--enable-deflate        Deflate transfer encoding support
--enable-xml2enc        i18n support for markup filters
--enable-proxy-html     Fix HTML Links in a Reverse Proxy
--enable-http           HTTP protocol handling. The http module is a basic
                        one that enables the server to function as an HTTP
                        server. It is only useful to disable it if you want
                        to use another protocol module instead. Don't
                        disable this module unless you are really sure what
                        you are doing. Note: This module will always be
                        linked statically.
```



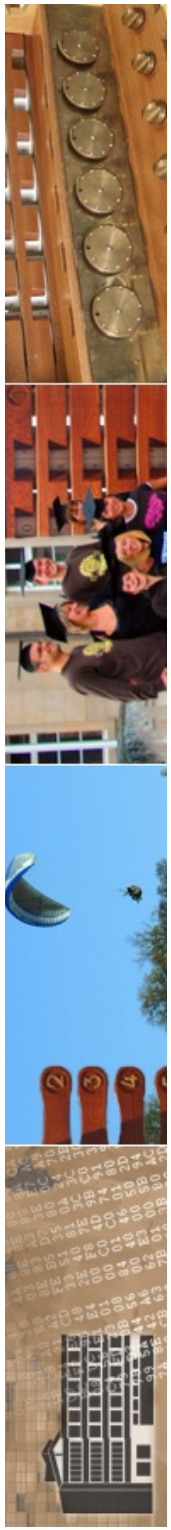
```
--disable-mime           mapping of file-extension to MIME. Disabling this
                         module is normally not recommended.
--enable-ldap            LDAP caching and connection pooling services
--disable-log-config     logging configuration. You won't be able to log
                         requests to the server without this module.
--enable-log-debug       configurable debug logging
--enable-log-forensic    forensic logging
--enable-logio           input and output logging
--enable-lua             Apache Lua Framework
--enable-luajit          Enable LuaJit Support
--disable-env            clearing/setting of ENV vars
--enable-mime-magic      automagically determining MIME type
--enable-cern-meta       CERN-type meta files
--enable-expires         Expires header control
--disable-headers        HTTP header control
--enable-ident           RFC 1413 identity check
--enable-usertrack       user-session tracking
--enable-unique-id       per-request unique ids
--disable-setenvif       basing ENV vars on headers
--disable-version        determining httpd version in config files
--enable-remoteip        translate header contents to an apparent client
                         remote_ip
--enable-proxy           Apache proxy module
--enable-proxy-connect   Apache proxy CONNECT module. Requires
                         --enable-proxy.
--enable-proxy-ftp       Apache proxy FTP module. Requires --enable-proxy.
--enable-proxy-http      Apache proxy HTTP module. Requires --enable-proxy.
--enable-proxy-fcgi      Apache proxy FastCGI module. Requires
                         --enable-proxy.
--enable-proxy-scgi      Apache proxy SCGI module. Requires --enable-proxy.
--enable-proxy-fdpass    Apache proxy to Unix Daemon Socket module. Requires
                         --enable-proxy.
--enable-proxy-wstunnel Apache proxy Websocket Tunnel module. Requires
                         --enable-proxy.
--enable-proxy-ajp       Apache proxy AJP module. Requires --enable-proxy.
--enable-proxy-balancer Apache proxy BALANCER module. Requires
                         --enable-proxy.
--enable-proxy-express   mass reverse-proxy module. Requires --enable-proxy.
--enable-proxy-hcheck    reverse-proxy health-check module. Requires
                         --enable-proxy and --enable-watchdog.
--enable-session         session module
--enable-session-cookie session cookie module
--enable-session-crypto session crypto module
```



```
--enable-session-dbd    session dbd module
--enable-slotmem-shm    slotmem provider that uses shared memory
--enable-slotmem-plain  slotmem provider that uses plain memory
--enable-ssl            SSL/TLS support (mod_ssl)
--enable-ssl-staticlib-deps
                        link mod_ssl with dependencies of OpenSSL's static
                        libraries (as indicated by "pkg-config --static").
                        Must be specified in addition to --enable-ssl.
--enable-optional-hook-export
                        example optional hook exporter
--enable-optional-hook-import
                        example optional hook importer
--enable-optional-fn-import
                        example optional function importer
--enable-optional-fn-export
                        example optional function exporter
--enable-dialup         rate limits static files to dialup modem speeds
--enable-static-support Build a statically linked version of the support
                        binaries
--enable-static-htpasswd
                        Build a statically linked version of htpasswd
--enable-static-htdigest
                        Build a statically linked version of htdigest
--enable-static-rotatelogs
                        Build a statically linked version of rotatelogs
--enable-static-logresolve
                        Build a statically linked version of logresolve
--enable-static-htdbm   Build a statically linked version of htdbm
--enable-static-ab      Build a statically linked version of ab
--enable-static-checkgid
                        Build a statically linked version of checkgid
--enable-static-htcacheclean
                        Build a statically linked version of htcacheclean
--enable-static-httxt2dbm
                        Build a statically linked version of httxt2dbm
--enable-static-fcgistarter
                        Build a statically linked version of fcgistarter
--enable-http2          HTTP/2 protocol handling in addition to HTTP
                        protocol handling. Implemented by mod_http2. This
                        module requires a libnghttp2 installation. See
                        --with-nghttp2 on how to manage non-standard
                        locations. This module is usually linked shared and
                        requires loading.
```



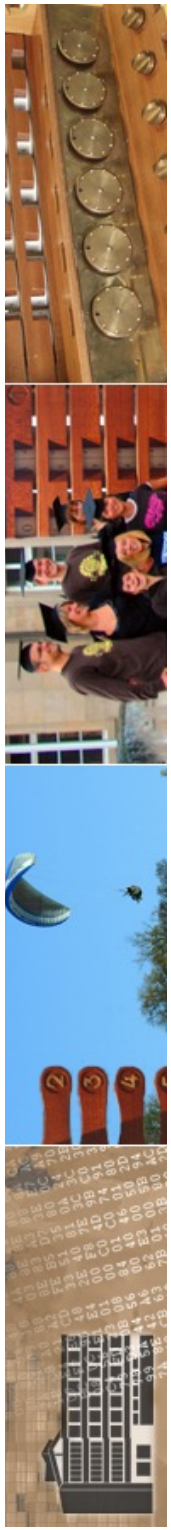
```
--enable-nghttp2-staticlib-deps
                        link mod_http2 with dependencies of libnghttp2's
                        static libraries (as indicated by "pkg-config
                        --static"). Must be specified in addition to
                        --enable-http2.
--enable-proxy-http2    HTTP/2 proxy module. This module requires a
                        libnghttp2 installation. See --with-nghttp2 on how
                        to manage non-standard locations. Also requires
                        --enable-proxy.
--enable-lbmethod-byrequests
                        Apache proxy Load balancing by request counting
--enable-lbmethod-bytraffic
                        Apache proxy Load balancing by traffic counting
--enable-lbmethod-bybusyness
                        Apache proxy Load balancing by busyness
--enable-lbmethod-heartbeat
                        Apache proxy Load balancing from Heartbeats
--enable-mpms-shared=MPM-LIST
                        Space-separated list of MPM modules to enable for
                        dynamic loading. MPM-LIST=list | "all"
--enable-unixd          unix specific support
--enable-privileges     Per-virtualhost Unix UserIDs and enhanced security
                        for Solaris
--enable-heartbeat      Generates Heartbeats
--enable-heartmonitor   Collects Heartbeats
--enable-dav            WebDAV protocol handling. --enable-dav also enables
                        mod_dav_fs
--disable-status        process/thread monitoring
--disable-autoindex     directory listing
--enable-asis           as-is filetypes
--enable-info           server information
--enable-suexec         set uid and gid for spawned processes
--enable-cgid           CGI scripts. Enabled by default with threaded MPMs
--enable-cgi            CGI scripts. Enabled by default with non-threaded
                        MPMs
--enable-dav-fs         DAV provider for the filesystem. --enable-dav also
                        enables mod_dav_fs.
--enable-dav-lock       DAV provider for generic locking
--enable-vhost-alias    mass virtual hosting module
--enable-negotiation    content negotiation
--disable-dir           directory request handling
--enable-imagemap       server-side imagemaps
--enable-actions        Action triggering on requests
```



```
  --with-apr=PATH         prefix for installed APR or the full path to
                             apr-config
  --with-apr-util=PATH    prefix for installed APU or the full path to
                             apu-config
  --with-pcre=PATH        Use external PCRE library
  --with-port=PORT        Port on which to listen (default is 80)
  --with-sslport=SSLPORT  Port on which to securelisten (default is 443)
  --with-distcache=PATH   Distcache installation directory
  --with-z=PATH           use a specific zlib library
  --with-libxml2=PATH     location for libxml2
  --with-lua=PATH         Path to the Lua 5.2/5.1 prefix
  --with-ssl=PATH         OpenSSL installation directory
  --with-nghttp2=PATH     nghttp2 installation directory
  --with-mpm=MPM          Choose the process model for Apache to use by
                          default. MPM={event|worker|prefork|winnt} This will
                          be statically linked as the only available MPM
                          unless --enable-mpms-shared is also specified.
  --with-module=module-type:module-file
                          Enable module-file in the modules/<module-type>
                          directory.
  --with-program-name     alternate executable name
  --with-suexec-bin       Path to suexec binary
  --with-suexec-caller    User allowed to call SuExec
  --with-suexec-userdir   User subdirectory
  --with-suexec-docroot   SuExec root directory
  --with-suexec-uidmin    Minimal allowed UID
  --with-suexec-gidmin    Minimal allowed GID
  --with-suexec-logfile   Set the logfile
  --with-suexec-safepath  Set the safepath
  --with-suexec-umask     umask for suexec'd process

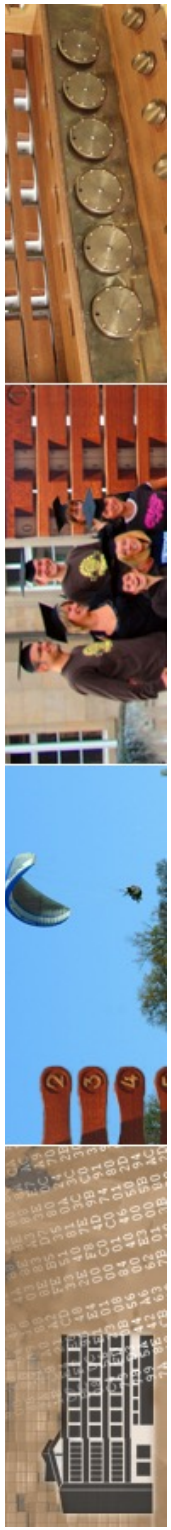
Some influential environment variables:
  CC          C compiler command
  CFLAGS      C compiler flags
  LDFLAGS     linker flags, e.g. -L<lib dir> if you have libraries in a
              nonstandard directory <lib dir>
  LIBS        libraries to pass to the linker, e.g. -l<library>
  CPPFLAGS    (Objective) C/C++ preprocessor flags, e.g. -I<include dir> if
              you have headers in a nonstandard directory <include dir>
  CPP         C preprocessor

Use these variables to override the choices made by `configure' or to help
it to find libraries and programs with nonstandard names/locations.
```



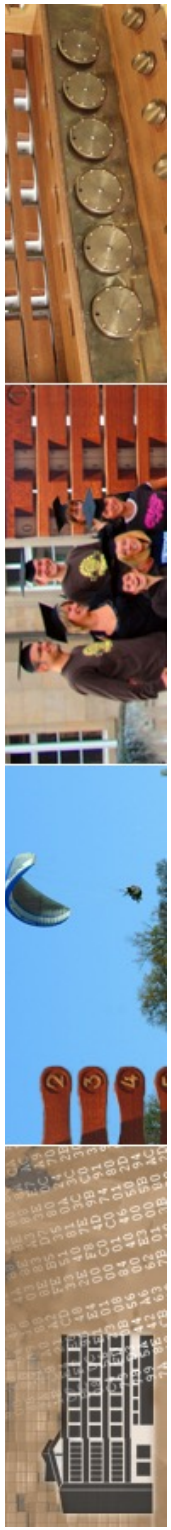
einige ASF-Standardmodule

- mod_core
  - "Ständig verfügbare Kernfunktionen des Apache HTTP Servers"
  - Kerndirektiven wie Directory und DocumentRoot
  - muss vorhanden sein
- mod_access
  - Zugriffssteuerung auf Apache über Hostnamen oder IP-Adressen



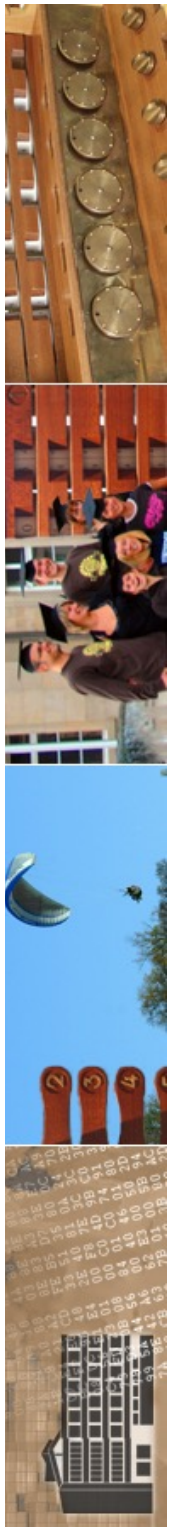
einige ASF-Standardmodule

- mod_actions
  - Zuordnung von MIME-Typen zu CGI-Scripten
- mod_alias
  - ändern von URL-Pfaden auf anderes Verzeichnis (Alias) oder komplett andere URL
- mod_asis
  - Auslieferung von Dokumenten ohne HTTP-Header
  - nützlich für Redirects (Umleitungen)



einige ASF-Standardmodule

- mod_auth
  - passwortgeschütze Inhalte
  - einfachste Variante der Authentifizierung
- mod_autoindex
  - wenn keine Indexdatei vorhanden ist, wird ein Verzeichnisindex erzeugt
  - konfigurierbar
  - Sicherheit?!
    - etwa im cgi-bin-Verzeichnis…

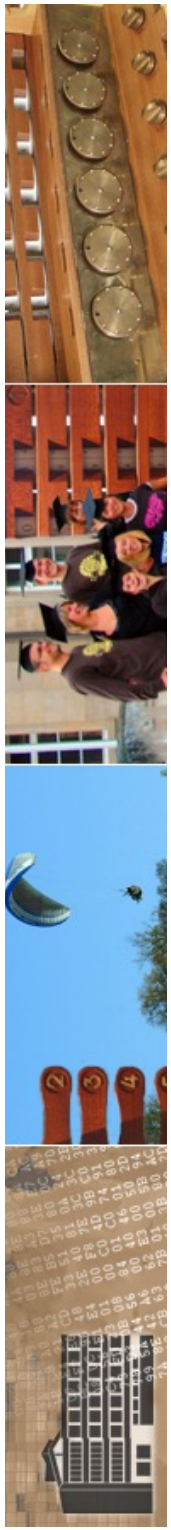


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

mod_autoindex

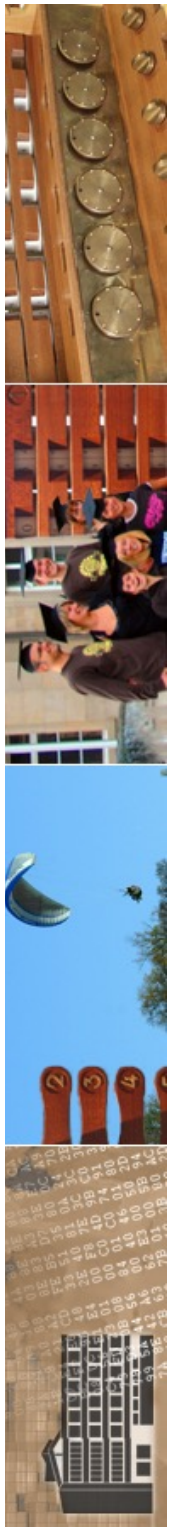


Vorsicht Security, etwa bei cgi-bin



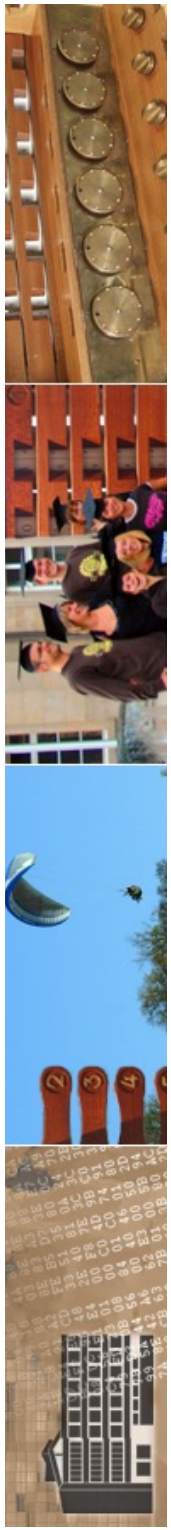
einige ASF-Standardmodule

- mod_cgi
  - Ausführen von CGI-Scripten
- mod_cgid
  - ähnlich mod_cgi, aber für multi-thread-Model
- mod_dir
  - legt Standards bei Zugriff auf Verzeichnis fest
  - Direktive **`DirectoryIndex`**
  - auch mehrere Angaben möglich, Reihenfolge wichtig
    - etwa **`index.html index.php index.shtml`**



einige ASF-Standardmodule

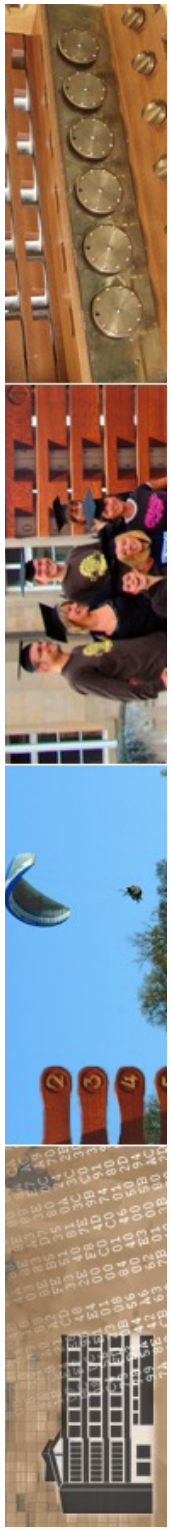
- mod_env
  - Zugriff auf Umgebungsvariablen für Apache bzw. Script
- mod_imap
  - serverseitge Image-Maps
- mod_include
  - Server Side Includes (SSI)
- mod_log_config
  - konfigurierbare Logfiles



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

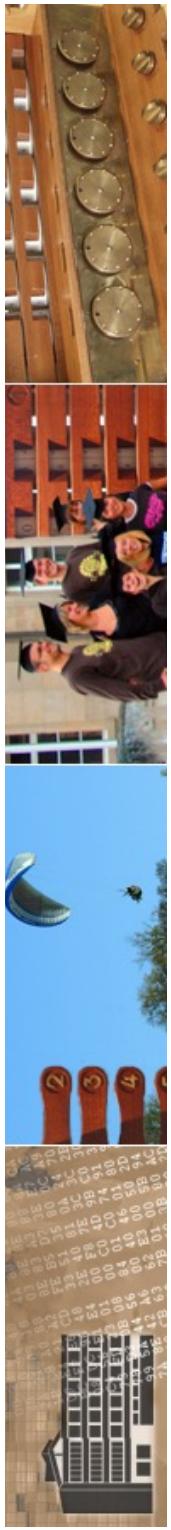
einige ASF-Standardmodule

- mod_mime
  - Erzeugung von MIME-Headers (Multipurpose Internet Mail Extensions) aufgrund der Endung der auszuliefernden Datei
- mod_negotiation
  - Steuerung der Inhalte durch den HTTP-Header des Clientrequests
    - Bsp: Sprachanpassung an Client



einige ASF-Standardmodule

- mod_setenvif
  - Steuerung des Apaches (Umgebungsvariablen) durch die Clientanfrage
- mod_so
  - ermöglicht dynamische Module
  - dieses Modul muss statisch geladen werden
- mod_status
  - Statusbericht des Apache



mod_status

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Apache Status

127.0.0.1/server-status

Apps M240 D800 HIS U Tü Zabbix Support für SVP... Benutzerhandb... Sony : SVZ1311...

Apache Server Status for 127.0.0.1 (via 127.0.0.1)

Server Version: Apache/2.4.9 (Win64) PHP/5.5.12
Server MPM: WinNT
Apache Lounge VC11 Server Built: Mar 16 2014 12:42:59

Current Time: Friday, 24-Oct-2014 12:12:28 Mitteleuropäische Sommerzeit
Restart Time: Friday, 24-Oct-2014 12:10:33 Mitteleuropäische Sommerzeit
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 1 minute 55 seconds
Server load: -1.00 -1.00 -1.00
Total accesses: 3 - Total Traffic: 70 kB
.0261 requests/sec - 623 B/second - 23.3 kB/request
1 requests currently being processed, 63 idle workers

_____W_____

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	SS	Req	Conn	Child	Slot	Client	VHost	Request
0-0	25352	0/3/3	W	0	0	0.0	0.07	0.07	127.0.0.1	::1:80	GET /server-status HTTP/1.1

Srv Child Server number - generation

Apache Server Status for 134.2.6.167 (via 134.2.6.167)

Server Version: Apache/2.4.46 (Unix)
Server MPM: event
Server Built: Nov 12 2020 09:37:06

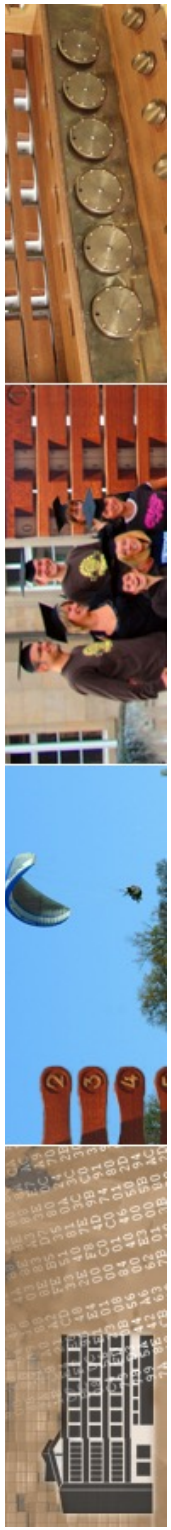
Current Time: Wednesday, 25-Nov-2020 20:51:20 CET
Restart Time: Wednesday, 25-Nov-2020 20:48:57 CET
Parent Server Config. Generation: 5
Parent Server MPM Generation: 4
Server uptime: 2 minutes 23 seconds
Server load: 0.03 0.03 0.00
Total accesses: 11 - Total Traffic: 152 kB - Total Duration: 188
CPU Usage: u.02 s.05 cu.04 cs.06 - .119% CPU load
.0769 requests/sec - 1088 B/second - 13.8 kB/request - 17.0909 ms/request
1 requests currently being processed, 99 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections		
			total	accepting	busy	idle	writing	keep-alive	closing
0	27535	no	0	yes	1	24	0	0	0
1	27536	no	0	yes	0	25	0	0	0
2	27538	no	0	yes	0	25	0	0	0
3	27864	no	0	yes	0	25	0	0	0
Sum	4	0	0		1	99	0	0	0

```
_____W_____________________________________________________
_____________________________.............................
..........................................................
..........................................................
..........................................................
..........................................................
..........................................................
..........................................................
..........
```

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process

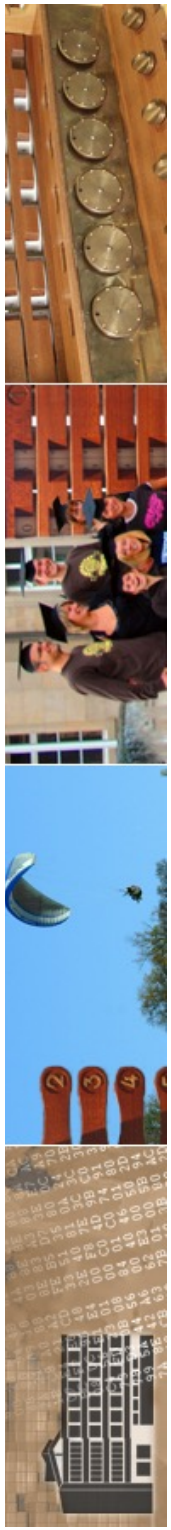
Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-4	27535	0/1/1	_	0.00	65	2	2	0.0	0.01	0.01	134.2.72.137	http/1.1	infodienste.uni-tuebingen.de:47	GET /manual/style/css/manual-print.css HTTP/1.1
0-4	27535	0/1/1	_	0.00	68	73	73	0.0	0.00	0.00	134.2.72.137	http/1.1	infodienste.uni-tuebingen.de:47	GET /manual/ HTTP/1.1
0-4	27535	0/1/1	_	0.00	65	0	0	0.0	0.00	0.00	134.2.72.137	http/1.1	infodienste.uni-tuebingen.de:47	GET /manual/style/css/manual-loose-100pc.css HTTP/1.1
0-4	27535	0/1/1	_	0.00	65	0	0	0.0	0.00	0.00	134.2.72.137	http/1.1	infodienste.uni-tuebingen.de:47	GET /manual/images/left.gif HTTP/1.1



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

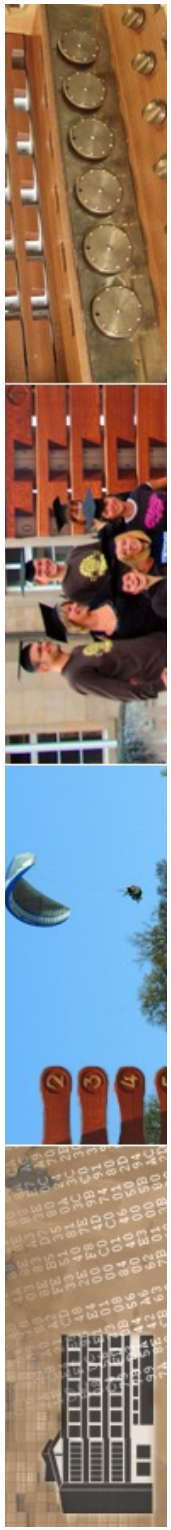
ASF-Standardmodule

- mod_userdir
  - private Verzeichnisse für lokale User nach der Syntax
    http://servername/~username



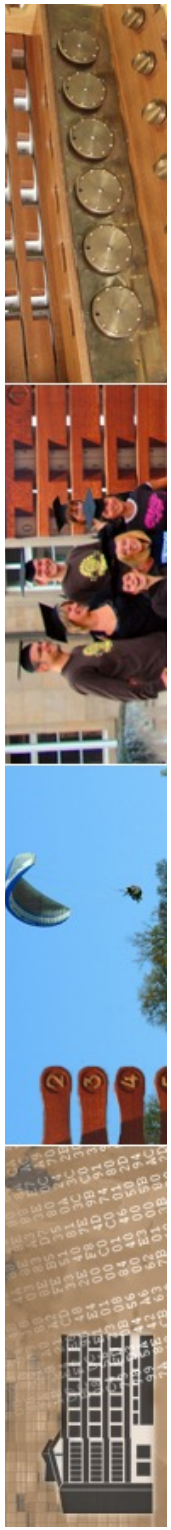
ASF-Module

- mod_auth_ldap
  - ldap-Authentifizierung
- mod_cache
  - Caching für Proxy
- mod_expires
  - setzt im HTTP-Header Expires

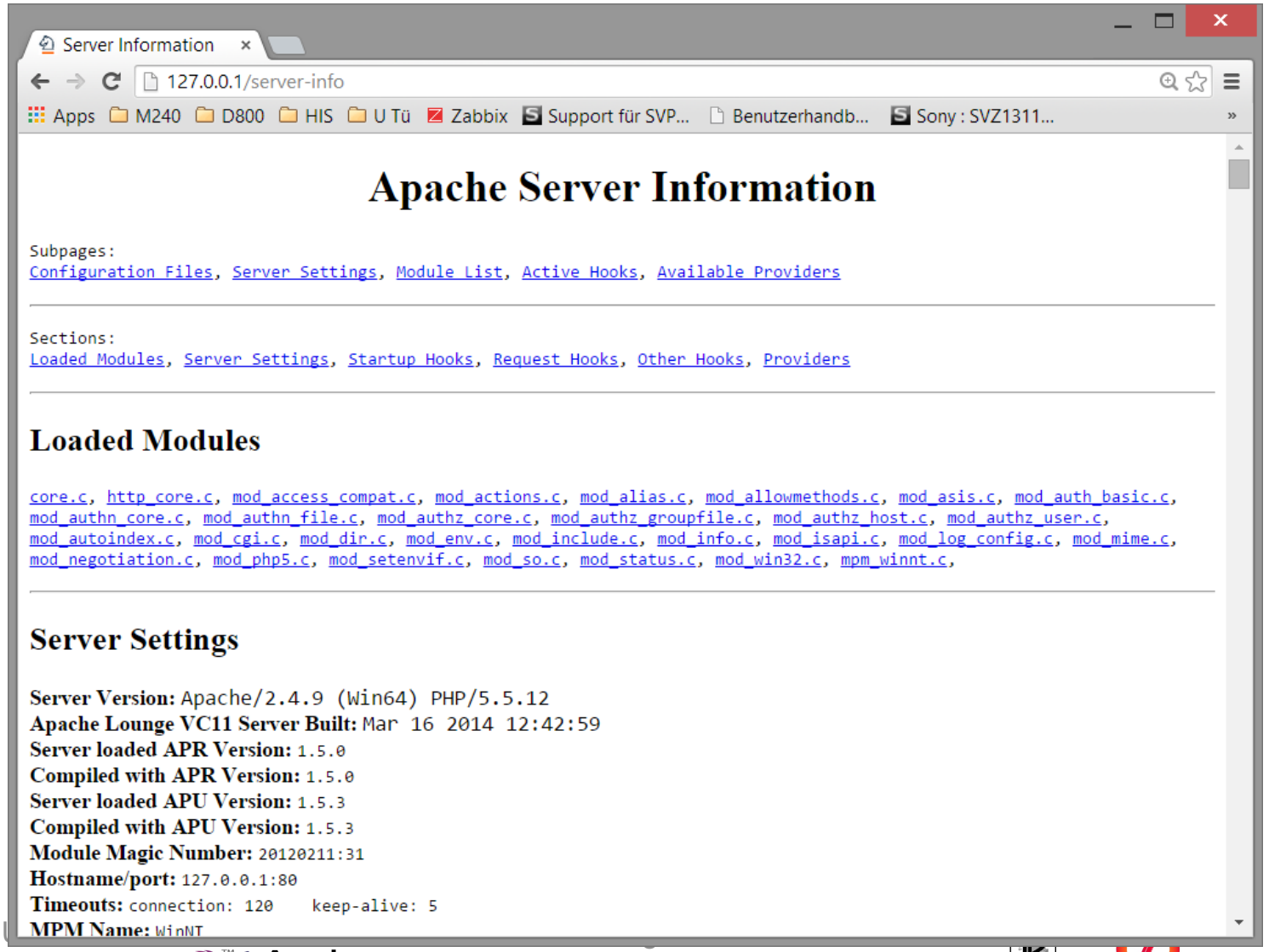


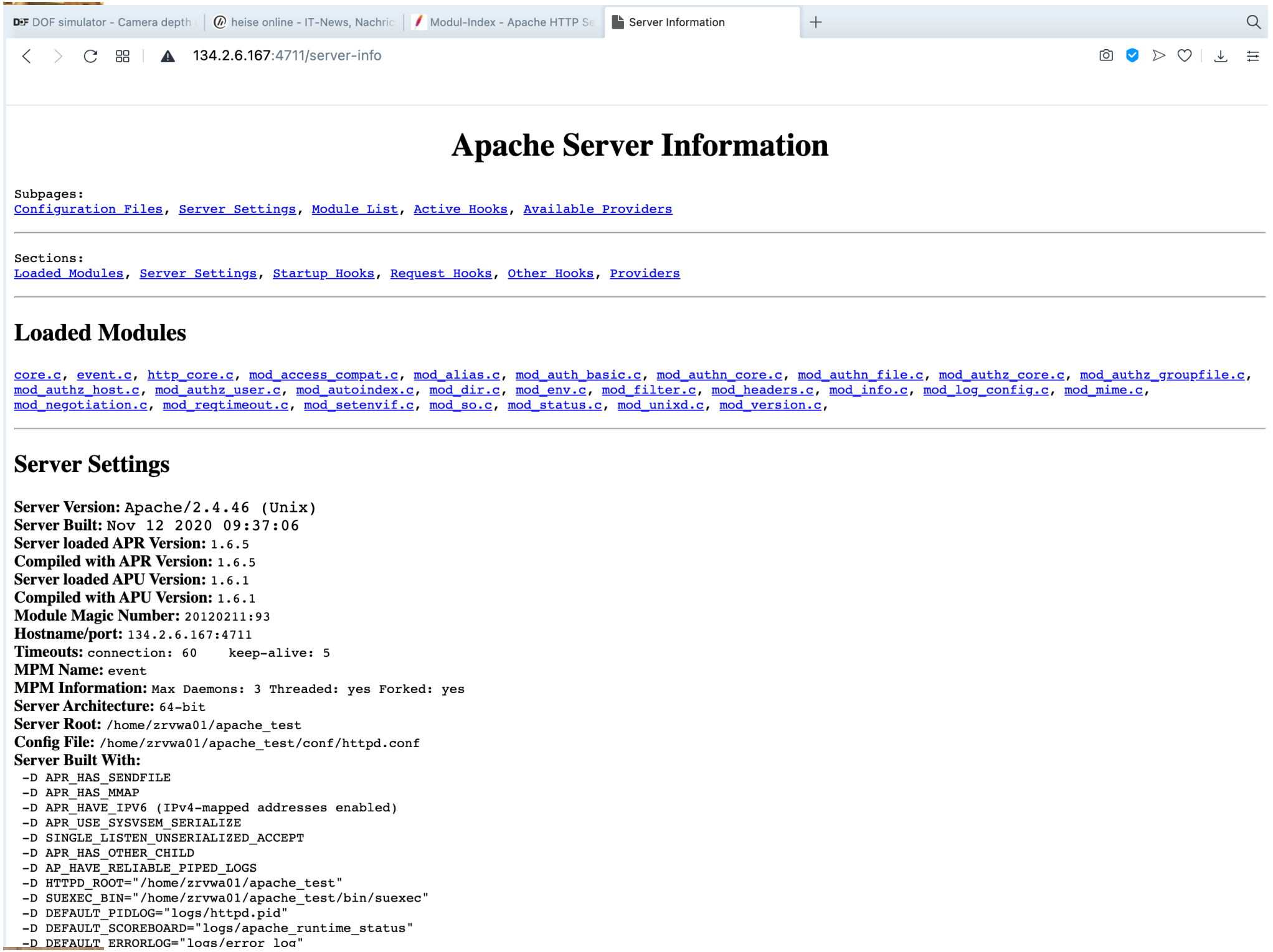
ASF-Module

- mod_headers
  - beliebige Veränderung des HTTP-Headers
- mod_info
  - Informationen über den Server – Vorschicht wg. Security
- mod_perl
  - integriert Perl-Interpreter in Apache und steigert damit wesentlich die Performance von Perl-Scripten
  - auch Caching compilierter Perl-Scripte
  - es gibt auch mod_python, mod_ruby, mod_mono, …



mod_info


Server Information
127.0.0.1/server-info
Apps M240 D800 HIS U Tü Zabbix Support für SVP... Benutzerhandb... Sony : SVZ1311...
Apache Server Information
Subpages:
Configuration Files, Server Settings, Module List, Active Hooks, Available Providers
Sections:
Loaded Modules, Server Settings, Startup Hooks, Request Hooks, Other Hooks, Providers
Loaded Modules
core.c, http_core.c, mod_access_compat.c, mod_actions.c, mod_alias.c, mod_allowmethods.c, mod_asis.c, mod_auth_basic.c, mod_authn_core.c, mod_authn_file.c, mod_authz_core.c, mod_authz_groupfile.c, mod_authz_host.c, mod_authz_user.c, mod_autoindex.c, mod_cgi.c, mod_dir.c, mod_env.c, mod_include.c, mod_info.c, mod_isapi.c, mod_log_config.c, mod_mime.c, mod_negotiation.c, mod_php5.c, mod_setenvif.c, mod_so.c, mod_status.c, mod_win32.c, mpm_winnt.c,
Server Settings
Server Version: Apache/2.4.9 (Win64) PHP/5.5.12
Apache Lounge VC11 Server Built: Mar 16 2014 12:42:59
Server loaded APR Version: 1.5.0
Compiled with APR Version: 1.5.0
Server loaded APU Version: 1.5.3
Compiled with APU Version: 1.5.3
Module Magic Number: 20120211:31
Hostname/port: 127.0.0.1:80
Timeouts: connection: 120 keep-alive: 5
MPM Name: WinNT



Apache Server Information

Subpages:
Configuration Files, Server Settings, Module List, Active Hooks, Available Providers

---

Sections:
Loaded Modules, Server Settings, Startup Hooks, Request Hooks, Other Hooks, Providers

---

Loaded Modules

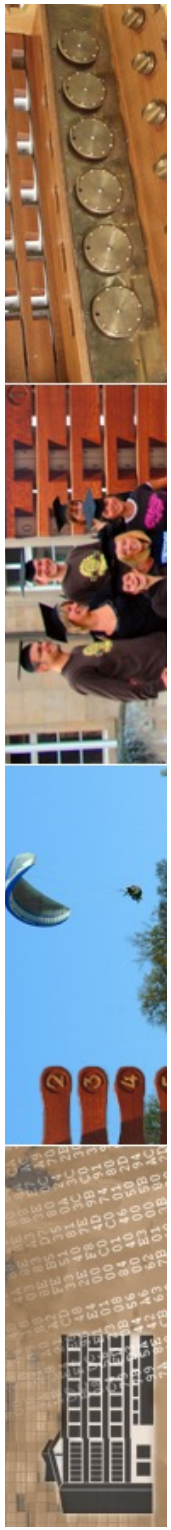
core.c, event.c, http_core.c, mod_access_compat.c, mod_alias.c, mod_auth_basic.c, mod_authn_core.c, mod_authn_file.c, mod_authz_core.c, mod_authz_groupfile.c, mod_authz_host.c, mod_authz_user.c, mod_autoindex.c, mod_dir.c, mod_env.c, mod_filter.c, mod_headers.c, mod_info.c, mod_log_config.c, mod_mime.c, mod_negotiation.c, mod_reqtimeout.c, mod_setenvif.c, mod_so.c, mod_status.c, mod_unixd.c, mod_version.c,

---

Server Settings

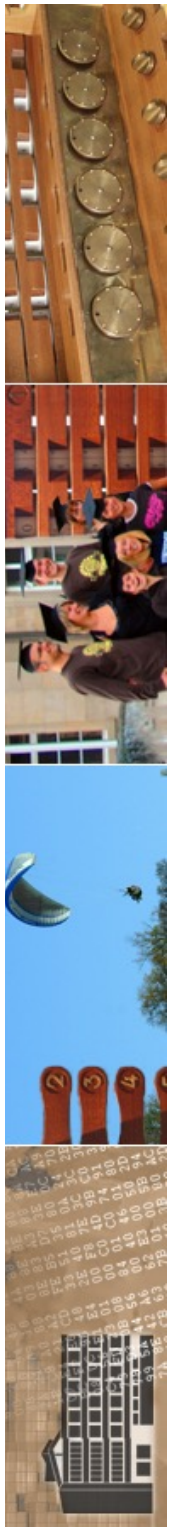
Server Version: Apache/2.4.46 (Unix)
Server Built: Nov 12 2020 09:37:06
Server loaded APR Version: 1.6.5
Compiled with APR Version: 1.6.5
Server loaded APU Version: 1.6.1
Compiled with APU Version: 1.6.1
Module Magic Number: 20120211:93
Hostname/port: 134.2.6.167:4711
Timeouts: connection: 60 keep-alive: 5
MPM Name: event
MPM Information: Max Daemons: 3 Threaded: yes Forked: yes
Server Architecture: 64-bit
Server Root: /home/zrvwa01/apache_test
Config File: /home/zrvwa01/apache_test/conf/httpd.conf
Server Built With:

```
 -D APR_HAS_SENDFILE
 -D APR_HAS_MMAP
 -D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
 -D APR_USE_SYSVSEM_SERIALIZE
 -D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
 -D APR_HAS_OTHER_CHILD
 -D AP_HAVE_RELIABLE_PIPED_LOGS
 -D HTTPD_ROOT="/home/zrvwa01/apache_test"
 -D SUEXEC_BIN="/home/zrvwa01/apache_test/bin/suexec"
 -D DEFAULT_PIDLOG="logs/httpd.pid"
 -D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
 -D DEFAULT_ERRORLOG="logs/error_log"
```



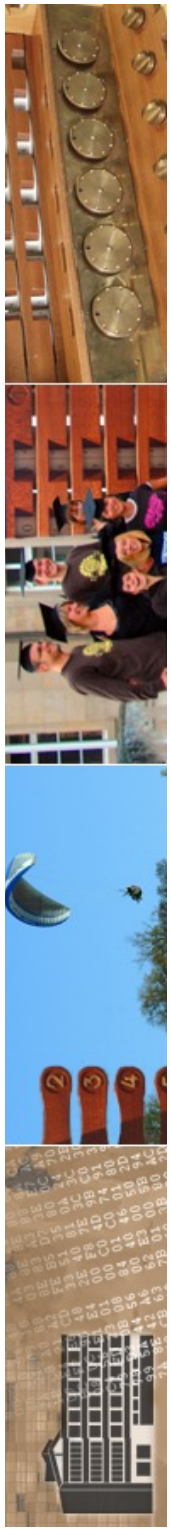
ASF-Module

- mod_proxy
  - Proxy für HTTP und FTP
  - Altenative: Squid
- mod_rewrite
  - Umschreiben der URL durch regulären Ausdruck



ASF-Module

- mod_speling
  - korrigiert Schreibfehler in URL
  - wird tatsächlich so geschrieben
- mod_ssl
  - Schnittstelle von Apache zu OpenSSL
  - Ralf D. Engelschall
- mod_usertrack
  - verfolgt Benutzeraktionen
  - verwendet Cookies

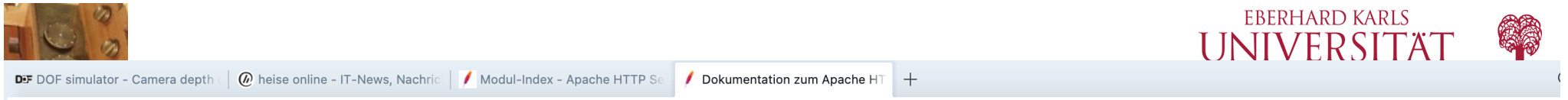


ASF-Module

- „manual“

  kein Modul, ein Alias

  - Apache wird mit der eigenen Dokumentation ausgeliefert



APACHE HTTP SERVER PROJECT

Module | Direktiven | FAQ | Glossar | Seitenind

Apache HTTP Server Version 2.4

Apache > HTTP-Server > Dokumentation

Dokumentation zum Apache HTTP Server Version 2.4

Verfügbare Sprachen: da | de | en | es | fr | ja | ko | pt-br | tr | zh-cn

Diese Übersetzung ist möglicherweise nicht mehr aktuell. Bitte prüfen Sie die englische Version auf die neuesten Änderungen.

Google-Suche

Hinweise zur Version

- Neue Funktionen in Version 2.3/2.4
- Neue Funktionen in Version 2.1/2.2
- Neue Funktionen in Version 2.0
- Upgrade auf Version 2.4
- Apache-Lizenz

Bedienungsanleitung

- Kompilieren und Installieren
- Starten
- Stoppen und Neustarten
- Konfigurationsanweisungen
- Module
- Multi-Processing-Module (MPMs)
- Filter
- Handler
- Der Ausdrucksparser
- Server und Hilfsprogramme
- Glossar

Benutzerhandbuch

- Erste Schritte
- An Adressen und Ports binden
- Konfigurationsdateien
- Konfigurationsabschnitte
- Caching von Inhalten
- Content Negotiation
- Dynamic Shared Objects (DSO)
- Umgebungsvariablen
- Log-Dateien
- URLs auf das Dateisystem abbilden
- Performance-Hinweise
- Tipps zur Sicherheit
- Serverweite Konfiguration
- SSL/TLS-Verschlüsselung
- Suexec für CGI
- Einführung in die URL-Manipulation
- Virtual-Hosts

Praxis / Anleitungen

- Authentisierung und Autorisierung
- Zugriffskontrolle
- CGI: Dynamische Inhalte
- .htaccess-Dateien
- Server Side Includes (SSI)
- Web-Verzeichnisse für Benutzer (public_html)

Plattform-spezifische Anmerkungen

- Microsoft Windows
- RPM-basierte Systeme (Redhat / CentOS / Fedora)
- Novell NetWare
- EBCDIC-Portierung

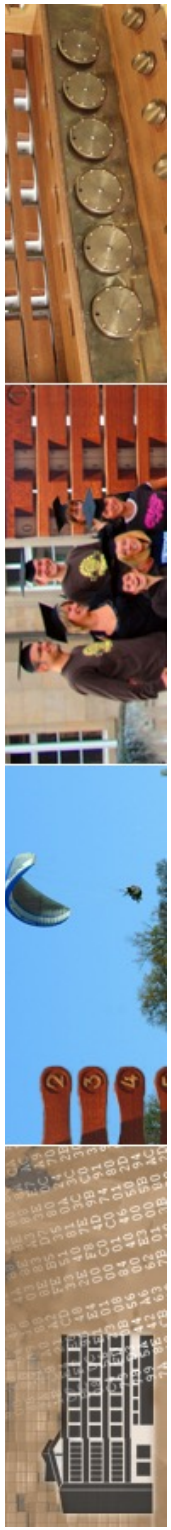
Weitere Themen

- Häufig gestellte Fragen (FAQ)
- Seitenindex
- Dokumentation für Entwickler
- Mithilfe bei der Dokumentation
- Sonstiges
- Wiki

Verfügbare Sprachen: da | de | en | es | fr | ja | ko | pt-br | tr | zh-cn

Copyright 2020 The Apache Software Foundation.
Lizenziert unter der Apache License, Version 2.0.

Module | Direktiven | FAQ | Glossar | Seitenind

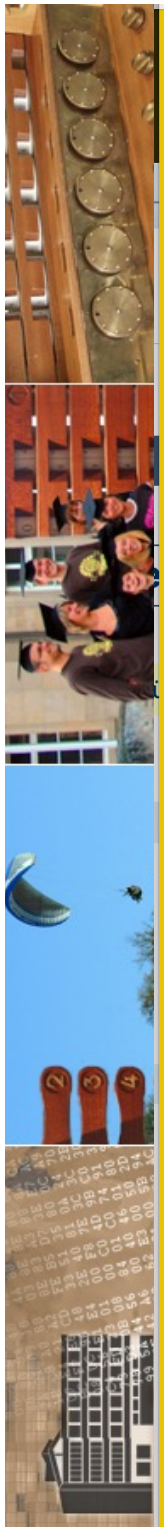


der config-Unterordner extra

- unter

  <apachedir>/conf/extera

  ausgelagerte Konfigurationsdateien, die je nach Setting von der httpd.conf nachgeladen werden



```
# Supplemental configuration
#
# The configuration files in the conf/extra/ directory can be
# included to add extra features or to modify the default configuration of
# the server, or you may simply copy their contents here and change as
# necessary.

# Server-pool management (MPM specific)
#Include conf/extra/httpd-mpm.conf

# Multi-language error messages
#Include conf/extra/httpd-multilang-errordoc.conf

# Fancy directory listings
#Include conf/extra/httpd-autoindex.conf

# Language settings
#Include conf/extra/httpd-languages.conf

# User home directories
#Include conf/extra/httpd-userdir.conf

# Real-time info on requests and configuration
Include conf/extra/httpd-info.conf

# Virtual hosts
#Include conf/extra/httpd-vhosts.conf

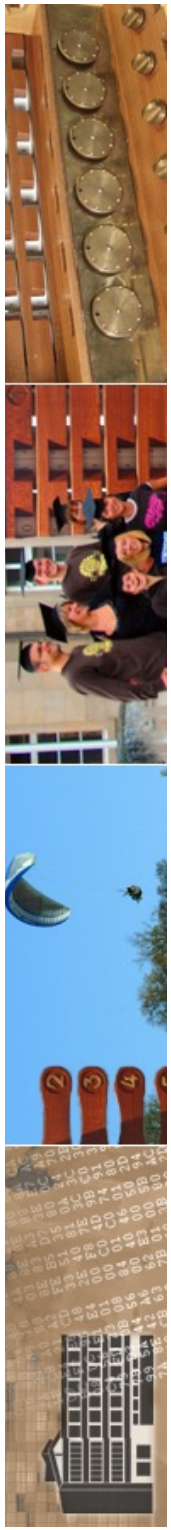
# Local access to the Apache HTTP Server Manual
Include conf/extra/httpd-manual.conf

# Distributed authoring and versioning (WebDAV)
#Include conf/extra/httpd-dav.conf

# Various default settings
#Include conf/extra/httpd-default.conf
```



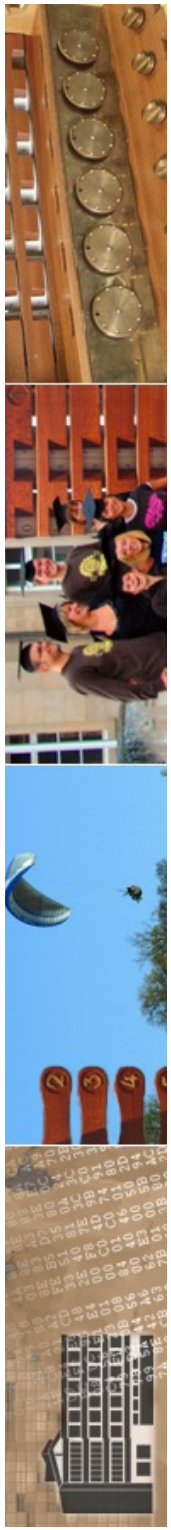
```
zrvwa01@infodienste => ll
insgesamt 72
drwxr-xr-x 2 zrvwa01  4096 27. Okt 09:31 ./
drwxr-xr-x 4 zrvwa01    79 27. Okt 09:31 ../
-rw-r--r-- 1 zrvwa01  2895 27. Okt 09:31 httpd-autoindex.conf
-rw-r--r-- 1 zrvwa01  1852 27. Okt 09:31 httpd-dav.conf
-rw-r--r-- 1 zrvwa01  2942 27. Okt 09:31 httpd-default.conf
-rw-r--r-- 1 zrvwa01  1119 27. Okt 09:31 httpd-info.conf
-rw-r--r-- 1 zrvwa01  5078 27. Okt 09:31 httpd-languages.conf
-rw-r--r-- 1 zrvwa01  1409 27. Okt 09:31 httpd-manual.conf
-rw-r--r-- 1 zrvwa01  4444 27. Okt 09:31 httpd-mpm.conf
-rw-r--r-- 1 zrvwa01  2243 27. Okt 09:31 httpd-multilang-errordoc.conf
-rw-r--r-- 1 zrvwa01 13518 27. Okt 09:31 httpd-ssl.conf
-rw-r--r-- 1 zrvwa01   694 27. Okt 09:31 httpd-userdir.conf
-rw-r--r-- 1 zrvwa01  1485 27. Okt 09:31 httpd-vhosts.conf
-rw-r--r-- 1 zrvwa01  3161 27. Okt 09:31 proxy-html.conf
zrvwa01@infodienste =>
```



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

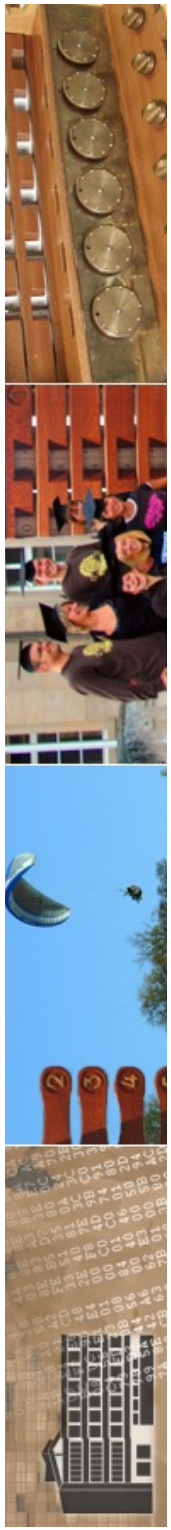
bedingte Konfiguration

- viele Direktiven sind nur möglich, wenn das entsprechende Modul geladen ist
- **bedingte Direktiven** werden nur aktiv, wenn das entsprechende Modul vorhanden ist:
  -
```
<IfModule ssl_module>
  SSLRandomSeed startup builtin
  SSLRandomSeed connect builtin
</IfModule>
```



Module von Drittanbietern

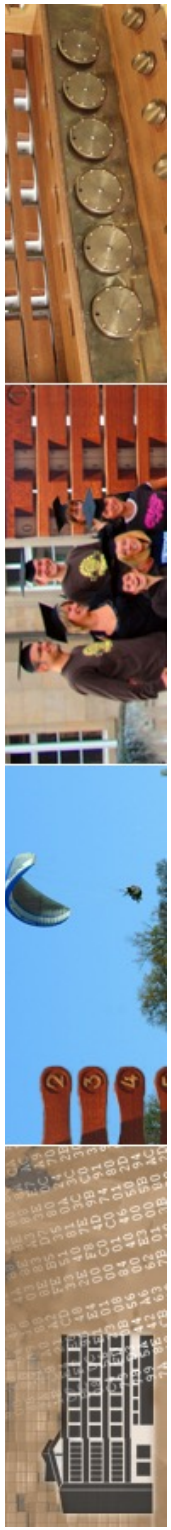
- neben den ASF-Modulen gibt es zahlreiche Module von **Drittanbietern**
- einige dieser Module sind mit der Zeit in das Apache-Projekt übernommen worden und werden heute mit dem Apache ausgeliefert
  - z.B. mod_ssl war bis Apache 2 ein Fremdmodul



Module von Drittanbietern

- wichtige Fremdmodule:
  - mod_php
  - mod_python
  - mod_perl
  - mod_ruby

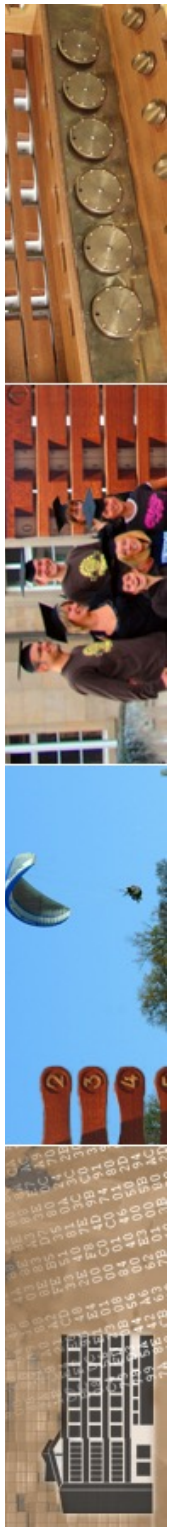
  - (mod_fastcgi -> inzwischen ASF)



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

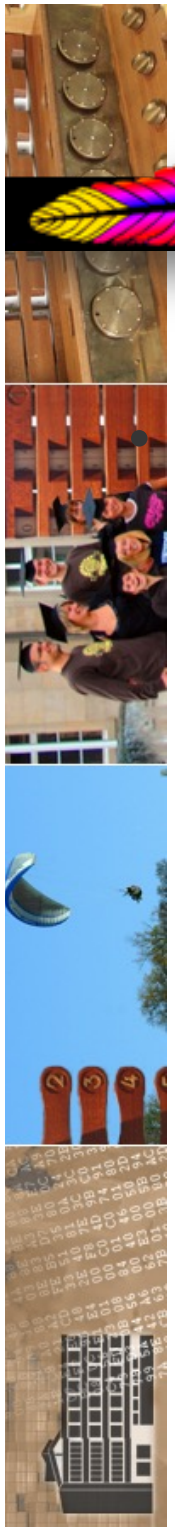
Apache Versionen

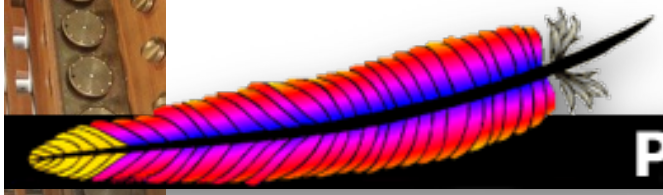
- aktuell ist 2.4
  - es gibt noch 1.3, 2.0 und 2.2
- Kernfunktionalität
  - Apache 1.3: Linux/Unix klont Prozesse durch fork
    Preforking: bereits im Vorfeld werden mehrere Kopien der Prozesse vorgehalten
    - funktioniert nicht unter Windows
  - Apache 2 (ab April 2002): Trennung von betriebssystem-abhängigen und betriebssystem-unabhängigen Funktionen



warum noch 1.3?

- Apache 1.3 ist immer noch im Einsatz und wird weiter entwickelt (insb. Security) – warum?
  - von 1.3 auf 2.0 hat sich die Apache-API stark verändert, so dass einige Fremdmodule nur für 1.3 verfügbar sind
  - Umstieg von 1.3 auf 2.x nicht ganz einfach

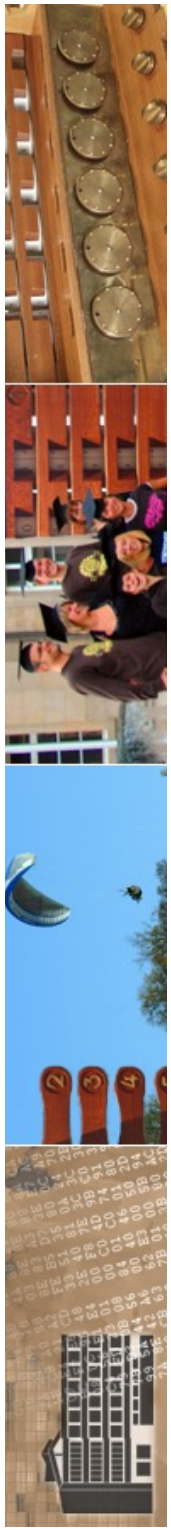




EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

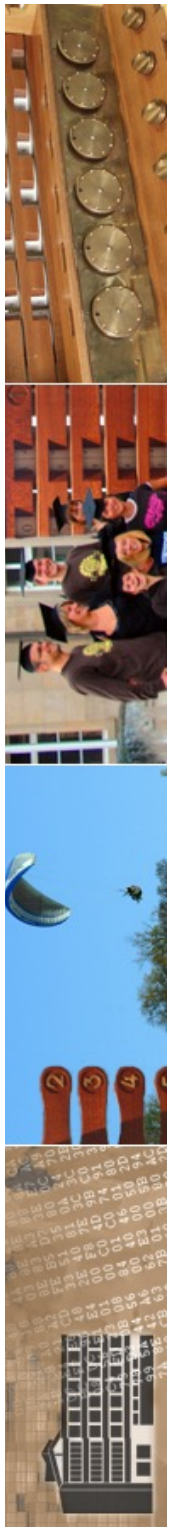
Apache Portable Runtime (APR)

- "The mission of the Apache Portable Runtime (APR) project is to create and maintain software libraries that provide a predictable and consistent interface to underlying platform-specific implementations. The primary goal is to provide an API to which software developers may code and be assured of predictable if not identical behaviour regardless of the platform on which their software is built, relieving them of the need to code special-case conditions to work around or take advantage of platform-specific deficiencies or features."



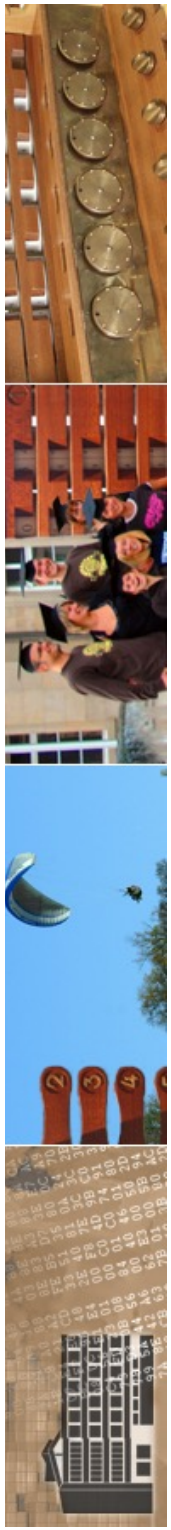
IP v6

- ab Apache 2 wird IP v6 voll unterstützt, wenn das zugrundeliegende Betriebssystem das neue Protokoll unterstützt



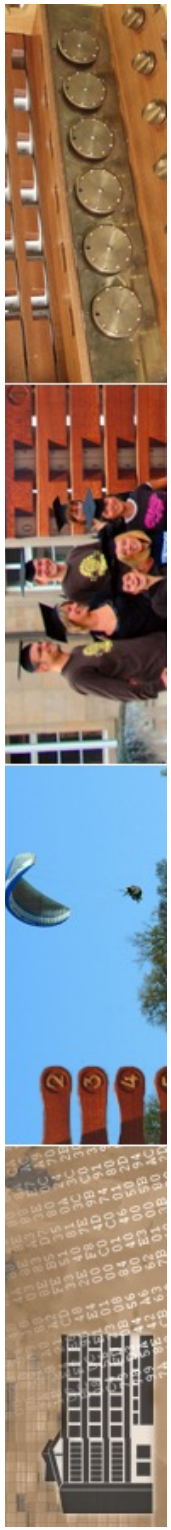
MPM: Multi Processing Modules

- ab Apache 2.0 MPMs
  - Auslagerung von Quellcode für Laufzeitverhalten
  - prozessbasiert oder Thread-basiert
    - auch Mischform: hybrides Laufzeitmodell (MPM_worker)
    - typische Direktiven
      - **`ThreadsPerChild`**
      - **`MaxClients`**
  - Beispiel: Modul MPM_winnt



Apache 2.2

- Änderungen von Apache 2.2 gegenüber 2.0
  - neue (einfachere) Authentifizierungs- und Autorisierungstools mod_authn_alias
  - load-balancer für mod_proxy (mod_proxy_balancer) und Integration des Jserv_Protokolls für Tomcat über mod_proxy_ajp
  - weitere Optimierungen bei MPM
  - vereinfachte Konfiguration

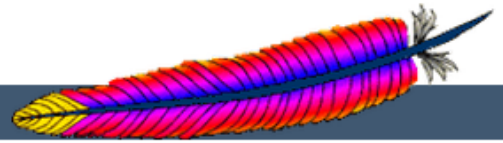


Apache 2.4

- Performanz verbessert
  - Default: Event-MPM-Modul setzt Threads ein
- über 40 (!) neue Module
  - fastCGI ist nun AFS-Modul
- zahlreiche Detailoptimierungen
  - Beispiel: Loglevel pro Verzeichnis







Overview of new features in Apache HTTP Server 2.4

Available Languages: en | fr | tr

This document describes some of the major changes between the 2.2 and 2.4 versions of the Apache HTTP Server. For new features since version 2.0, see the 2.2 new features document.

- Core Enhancements
- New Modules
- Module Enhancements
- Program Enhancements
- Documentation
- Module Developer Changes
- Comments

Core Enhancements

Run-time Loadable MPMs

Multiple MPMs can now be built as loadable modules at compile time. The MPM of choice can be configured at run time.

Event MPM

The Event MPM is no longer experimental but is now fully supported.

Asynchronous support

Better support for asynchronous read/write for supporting MPMs and platforms.

Per-module and per-directory LogLevel configuration

The `LogLevel` can now be configured per module and per directory. New levels `trace1` to `trace8` have been added above the `debug` log level.

Per-request configuration sections

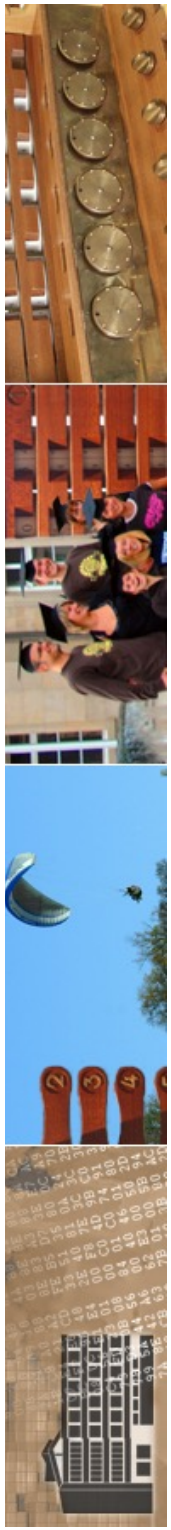
`<If>`, `<ElseIf>`, and `<Else>` sections can be used to set the configuration based on per-request criteria.

General-purpose expression parser

A new expression parser allows to specify complex conditions using a common syntax in directives like `SetEnvIfExpr`, `RewriteCond`, `Header`, `<If>`, and others.

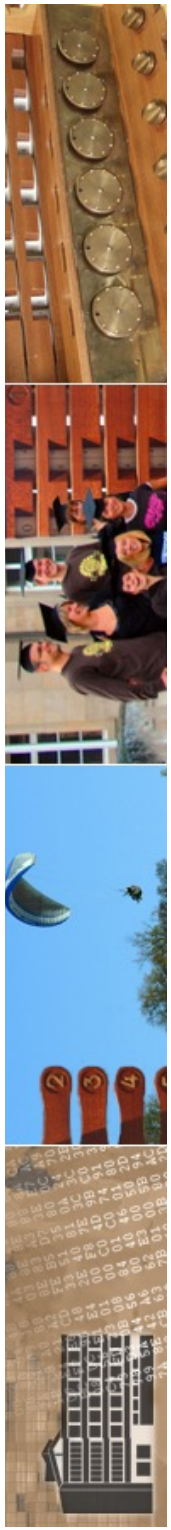
KeepAlive Timeout in milliseconds



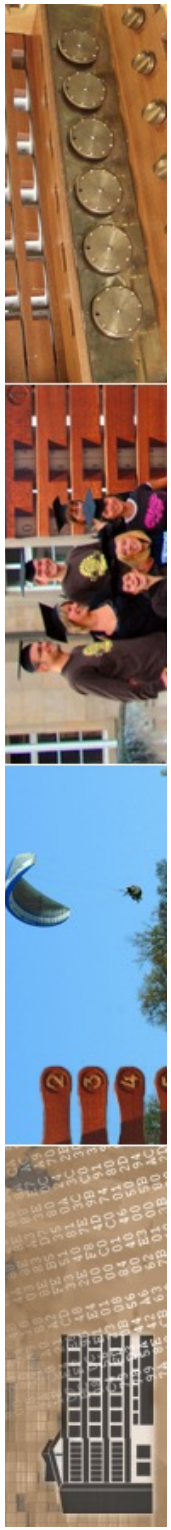


Apache und Module

- Apache wird modular erweitert
- Unterscheidung
  - statisch/dynamisch
  - ASF: Standard / Optional / Fremdmodule (z.B PHP)
- modulspezifische Konfiguration
  - Direktive <ifmodule ...>
- Performance
- Sicherheit

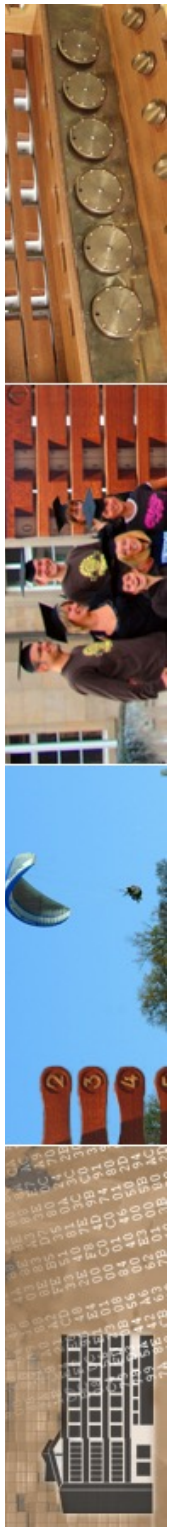






Prozess oder Thread

- in aller Kürze:
  - Prozess: Ausführen eines Programms (auch Task)
  - Thread: (auch leichtgewichtiger Prozess): Ausführungsstrang als Teil eines Prozesses
    - viele Threads zu einem Prozess (multithreading), die sich Ressourcen des Prozesses teilen
    - mögliches Problem: Kommunikation der Threads

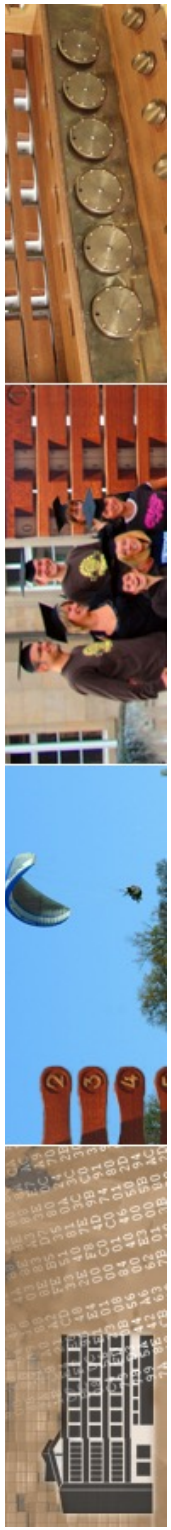


Prozessmanagement Apache 1.3

Hauptprozess des Apache

Kind 1 | Kind 2 | Kind 3 | ...

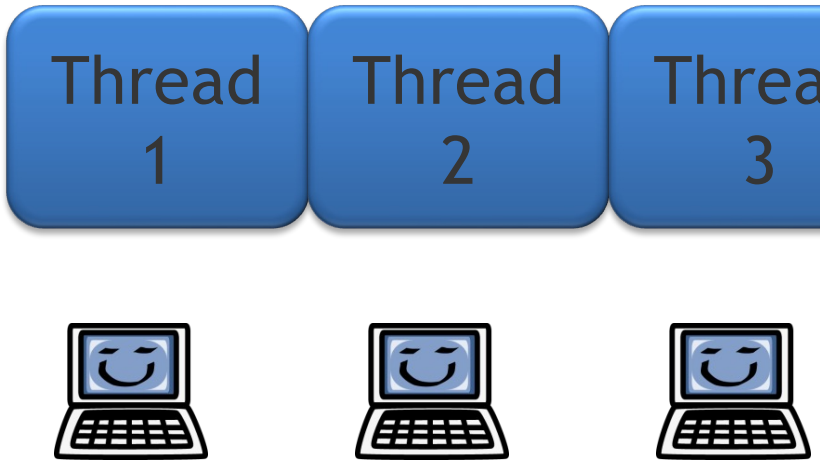




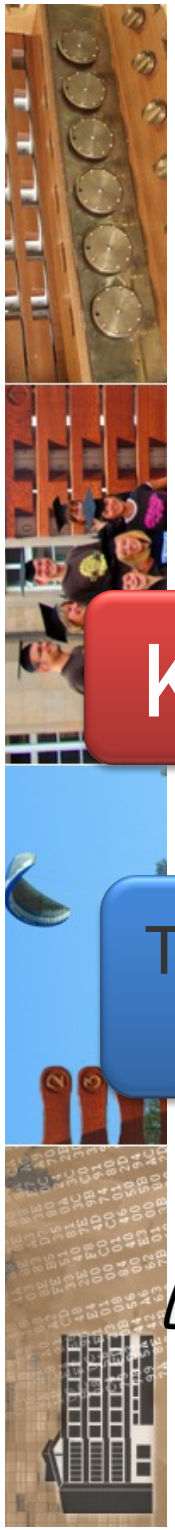
MPM_winnt-Prozessmodell

Hauptprozess des Apache

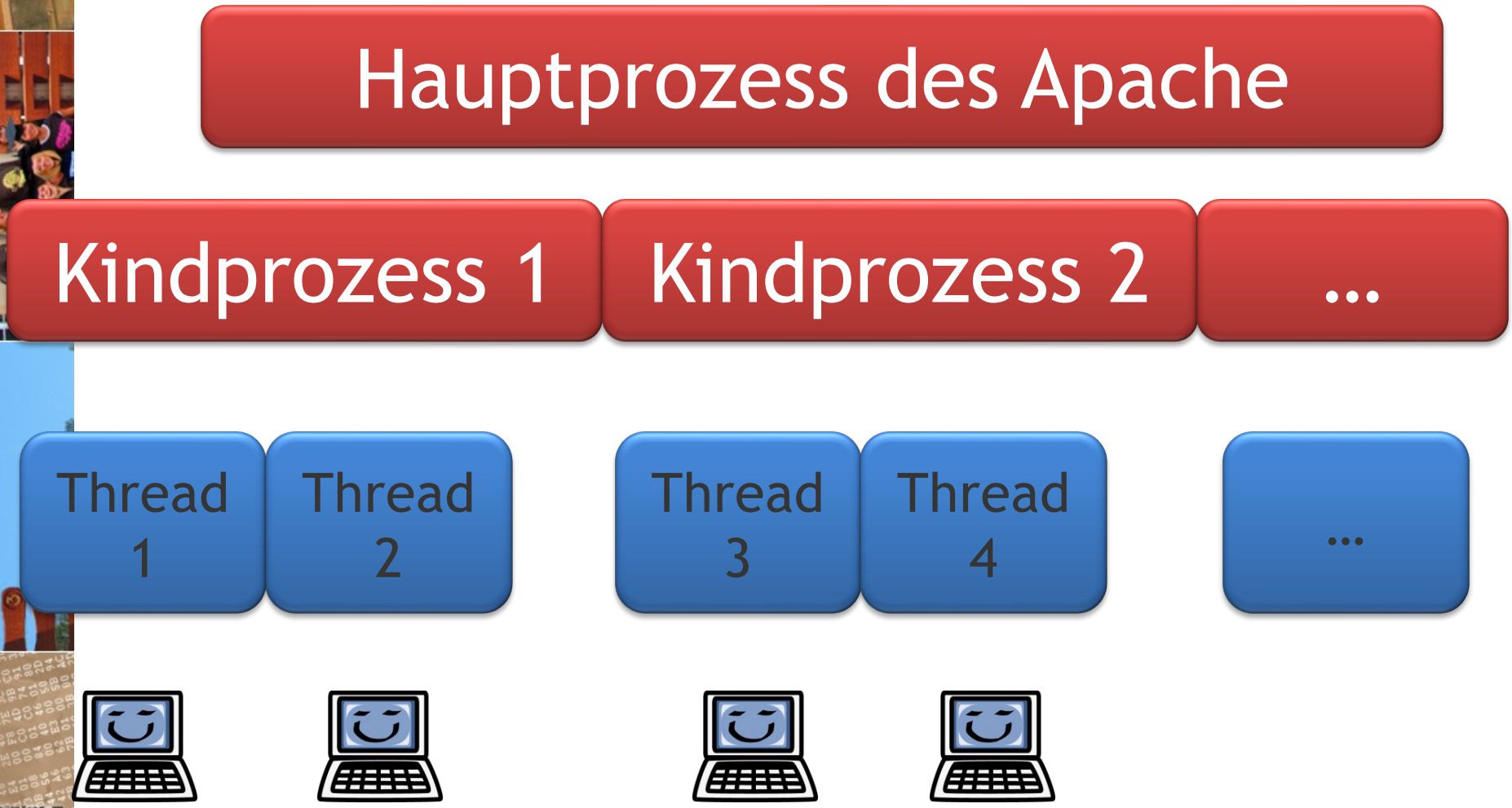
Kindprozess des Apache

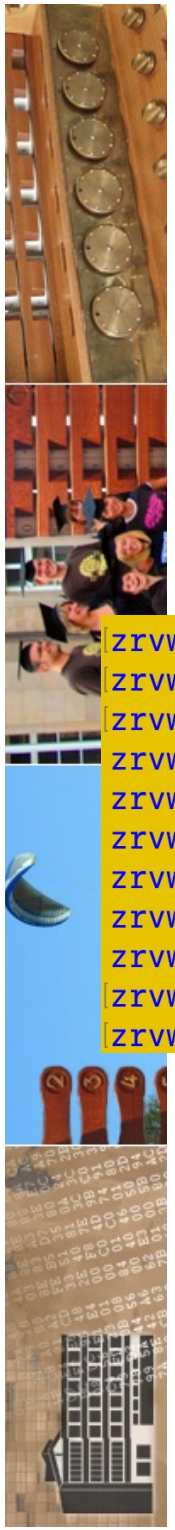


© 2026 Universität Tübingen



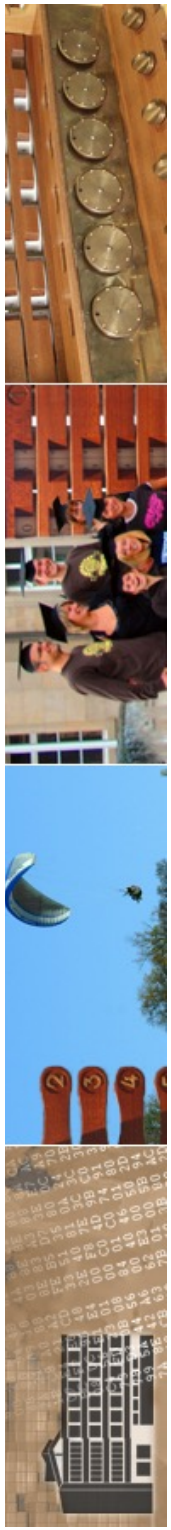
hybrides Laufzeitmodell





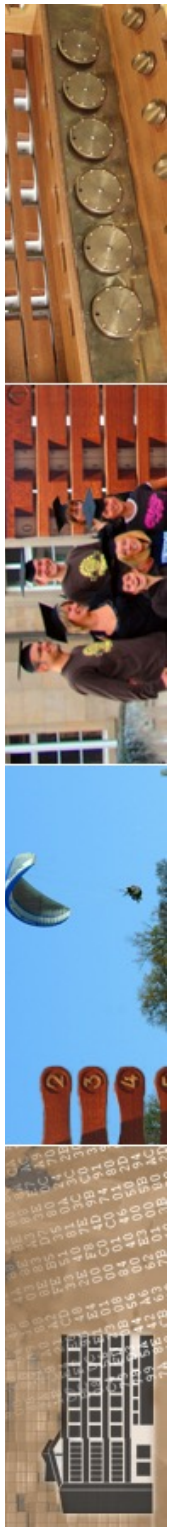
Apache 1x gestartet

```
[zrvwa01@infodienste =>
[zrvwa01@infodienste =>
[zrvwa01@infodienste => ps -ef|grep httpd|grep zrvwa01
zrvwa01  3519776       1  0 Nov02 ?        00:01:07 /home/zrvwa01/apache_test/bin/httpd -k start
zrvwa01  3519777 3519776  0 Nov02 ?        00:00:00 /home/zrvwa01/apache_test/bin/httpd -k start
zrvwa01  3519778 3519776  0 Nov02 ?        00:00:00 /home/zrvwa01/apache_test/bin/httpd -k start
zrvwa01  3519780 3519776  0 Nov02 ?        00:00:00 /home/zrvwa01/apache_test/bin/httpd -k start
zrvwa01  3567399 3519776  0 Nov02 ?        00:00:00 /home/zrvwa01/apache_test/bin/httpd -k start
zrvwa01  4125447 4125429  0 15:28 pts/91   00:00:00 grep httpd
[zrvwa01@infodienste =>
[zrvwa01@infodienste =>
```



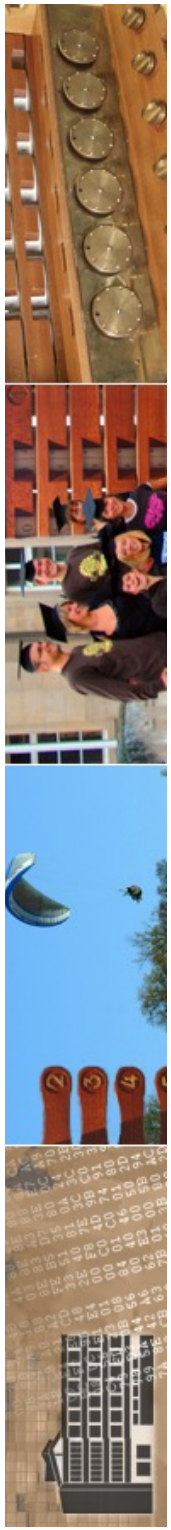
Apache und Security

- Apache ist nach der Grundinstallation schon brauchbar geschützt
- ein paar Optimierungen sind meistens sinnvoll



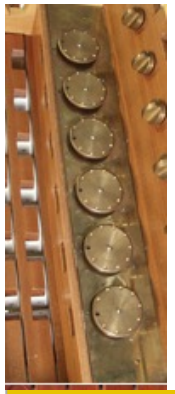
Adresse und Ports

- Standard-Port ist 80, kann faktisch nicht geändert werden
  - Standard in HTTP
- durch die Listen-Direktive kann aber zum Port auch die IP-Adresse für den Apache festgelegt werden
  - `Listen 134.2.6.167:80`



Benutzer und Gruppen

- in der httpd.conf wird die Benutzerkennung und die Gruppenkennung für die Apache-Kindprozesse festgelegt
- hier sollte eine spezielle Kennung mit geringen Zugriffsrechten gewählt werden

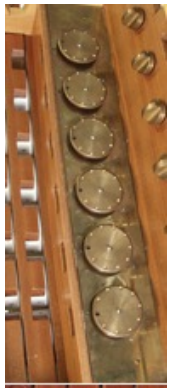


Benutzer und Gruppen

```
<IfModule unixd_module>
#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
# It is usually good practice to create a dedicated user and group for
# running httpd, as with most system services.
#
User daemon
Group daemon

</IfModule>
```



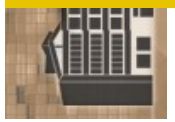


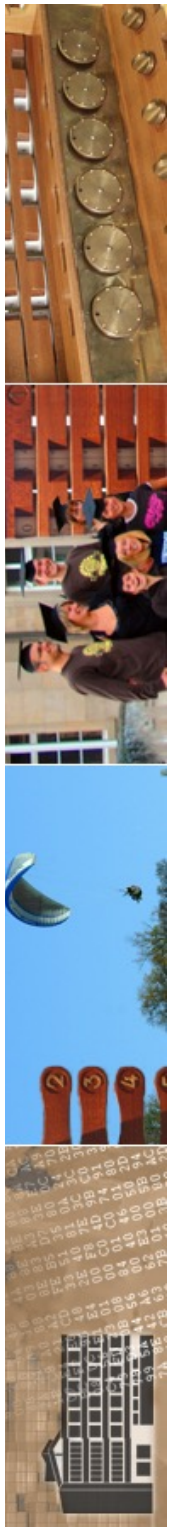

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Schutz des Wurzelverzeichnisses

```
#
# Deny access to the entirety of your server's filesystem. You must
# explicitly permit access to web content directories in other
# <Directory> blocks below.
#
<Directory />
    AllowOverride none
    Require all denied
</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#
```

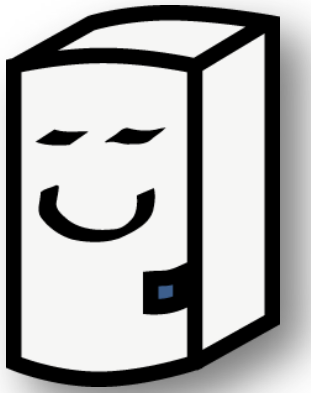


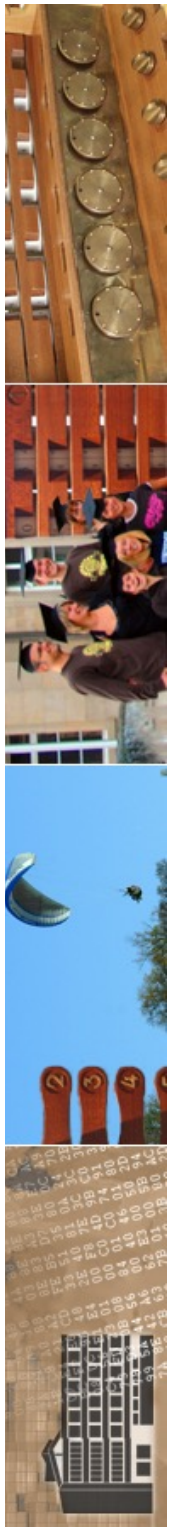


nicht zu viel über sich verraten

- Direktive **`ServerSignature`** auf Wert Off oder Email setzen

- mod_status und mod_info schützen!
  - das ist defaultmäßig eingestellt

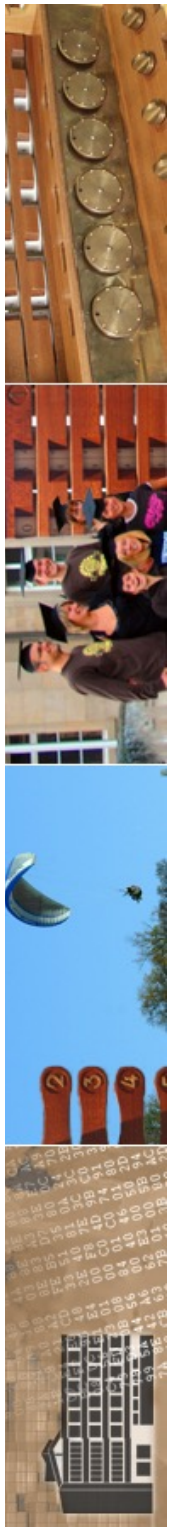




weniger ist mehr

- wird CGI wirklich benötigt?
- wird SSL wirklich benötigt?
- je weniger Module desto besser…

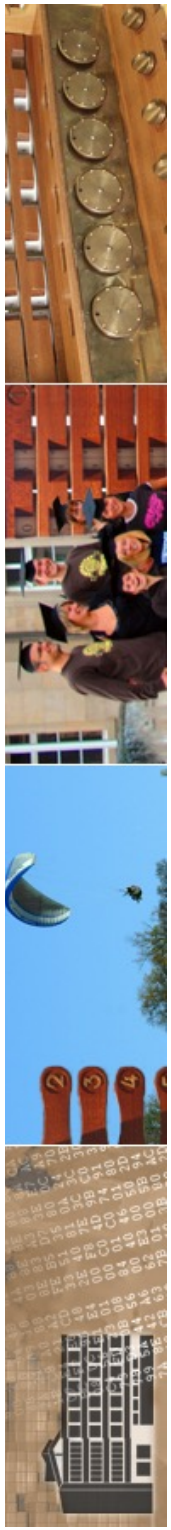




weitere Optimierung

- die Performance des Apache kann durch auf die Hardware angepasstes Compilieren merklich gesteigert werden

- für den gcc-Compiler relevant:
  Die Umgebungsvariable **`CFLAGS`**
  - Festlegung der CPU-Architektur, der Speicherstruktur (L1/L2/L3), ...
  - Optimierungs-Level
    https://wiki.gentoo.org/wiki/GCC_optimization



gentoo linux™ Wiki

Main page | Recent changes | Help | Gentoo | Documentation

Page | Discussion

GCC optimization

Other languages:	English • español • français • italiano • 日本語 • 한국어 • por

This guide provides an introduction to optimizing compiled code using safe, sane *CFLAGS* and *CXX*

Contents [hide]

1 Introduction
- 1.1 What are CFLAGS and CXXFLAGS?
- 1.2 How are they used?
- 1.3 Misconceptions
- 1.4 Ready?

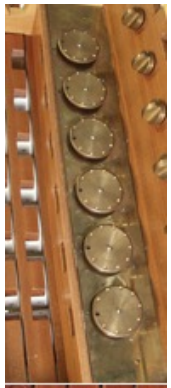
2 Optimizing
- 2.1 The basics
- 2.2 -march
- 2.3 -O
- 2.4 -pipe
- 2.5 -fomit-frame-pointer
- 2.6 -msse, -msse2, -msse3, -mmmx, -m3dnow

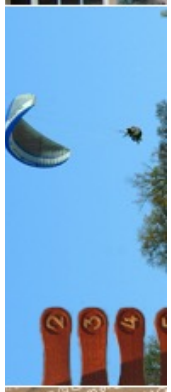
3 Optimization FAQs
- 3.1 But I get better performance with -funroll-loops -fomg-optimize!
- 3.2 What about -O levels higher than 3?
- 3.3 What about compiling outside the target machine?
- 3.4 What about redundant flags?
- 3.5 What about LDFLAGS?
- 3.6 Can I use per-package flags?

4 See also

5 External resources

© 2026 Tübing








3.11 Options That Control Optimization

These options control various sorts of optimizations.

Without any optimization option, the compiler's goal is to reduce the cost of compilation and to make debugging produce the expected results. Statements are independent: if you stop the program with a breakpoint between statements, you can then assign a new value to any variable or change the program counter to any other statement in the function and get exactly the results you expect from the source code.

Turning on optimization flags makes the compiler attempt to improve the performance and/or code size at the expense of compilation time and possibly the ability to debug the program.

The compiler performs optimization based on the knowledge it has of the program. Compiling multiple files at once to a single output file mode allows the compiler to use information gained from all of the files when compiling each of them.

Not all optimizations are controlled directly by a flag. Only optimizations that have a flag are listed in this section.

Most optimizations are completely disabled at `-O0` or if an `-O` level is not set on the command line, even if individual optimization flags are specified. Similarly, `-Og` suppresses many optimization passes.

Depending on the target and how GCC was configured, a slightly different set of optimizations may be enabled at each `-O` level than those listed here. You can invoke GCC with `-Q --help=optimizers` to find out the exact set of optimizations that are enabled at each level. See Options Controlling the Kind of Output, for examples.

`-O`
`-O1`

Optimize. Optimizing compilation takes somewhat more time, and a lot more memory for a large function.

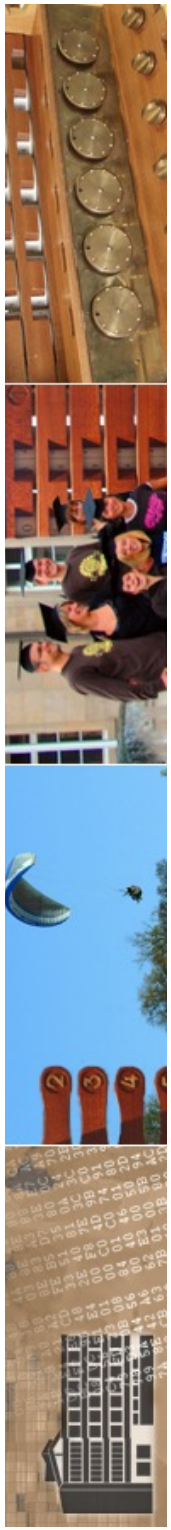
With `-O`, the compiler tries to reduce code size and execution time, without performing any optimizations that take a great deal of compilation time.

`-O` turns on the following optimization flags:

```
-fauto-inc-dec
-fbranch-count-reg
-fcombine-stack-adjustments
-fcompare-elim
-fcprop-registers
-fdce
-fdefer-pop
-fdelayed-branch
-fdse
-fforward-propagate
-fguess-branch-probability
-fif-conversion
```

https://gcc.gnu.org/onlinedocs/gcc/Optimize-Options.html

© Tü



virtuelle Hosts

- ein Apache-Server kann *viele Domainen hosten:*

  virtual hosts

  – gibt es namensbasiert und IP-basiert



Apache-Dokumentation zu virtuellen Hosts

Diese Übersetzung ist möglicherweise nicht mehr aktuell. Bitte prüfen Sie die englische Version auf die neuesten Änderungen.

Der Begriff *virtueller Host* (*Anm.d.Ü.:* engl. 'virtual host') bezieht sich auf die Praxis, mehr als ein Webangebot (z.B. `www.company1.com` und `www.company2.com`) auf einer einzigen Maschine zu betreiben. Virtuelle Hosts können "IP-basiert" sein, was bedeutet, dass jedes Webangebot eine andere IP besitzt, oder "Namens-basiert", was bedeutet, dass unter jeder IP-Adresse mehrere Namen laufen. Die Tatsache, dass sie auf dem gleichen physischen Server laufen, ist für den Endbenutzer nicht offensichtlich.

Der Apache war einer der ersten Server, der IP-basierte virtuelle Hosts von Haus aus direkt unterstützt hat. Seit Version 1.1 unterstützt der Apache sowohl IP-basierte als auch namensbasierte virtuelle Hosts (vhosts). Letzteres wird zuweilen auch *Host-basiert* oder *non-IP-Virtual-Host* genannt.

Nachfolgend finden Sie eine Liste von Dokumenten, die alle Details der Unterstützung von virtuellen Hosts ab Apache Version 1.3 beschreiben.

Unterstützung virtueller Hosts

- Namensbasierte virtuelle Hosts (Mehr als ein Webangebot pro IP-Adresse)
- IP-basierte virtuelle Hosts (Eine IP-Adresse für jedes Webangebot)
- Beispiele für virtuelles Hosts in typischen Installationen
- Datei-Deskriptor-Begrenzungen (oder *Zu viele Protokolldateien*)
- Dynamisch konfiguriertes Massen-Virtual-Hosting
- Tiefergehende Erörterung der Zuweisung virtueller Hosts

Konfigurationsdirektiven

- `<VirtualHost>`
- `NameVirtualHost`
- `ServerName`
- `ServerAlias`
- `ServerPath`

Bei der Suche von Fehlern in Ihrer Virtual-Host-Konfiguration ist die Apache-Befehlszeilenoption `-S` möglicherweise hilfreich. Geben Sie dazu den folgenden Befehl ein:

```
/usr/local/apache2/bin/httpd -S
```

Diese Anweisung gibt eine Beschreibung aus, wie der Apache die Konfigurationsdatei analysiert hat. Eine sorgfältige Überprüfung der IP-Adressen und Servernamen kann helfen, Konfigurationsfehler aufzudecken. (Lesen Sie die Dokumentation zum `httpd`-Programm für weitere Befehlszeilenoptionen.)





<VirtualHost>-Direktive

Beschreibung:	Enthält Direktiven, die nur auf bestimmte Hostnamen oder IP-Adressen angewendet werden
Syntax:	`<VirtualHost Adresse[:Port] [Adresse[:Port]] ...> ... </VirtualHost>`
Kontext:	Serverkonfiguration
Status:	Core
Modul:	core

`<VirtualHost>` und `</VirtualHost>` werden dazu verwendet, eine Gruppe von Direktiven zusammenzufassen, die nur auf einen bestimmten virtuellen Host angewendet werden. Jede Direktive, die im Virtual-Host-Kontext zulässig ist, kann verwendet werden. Wenn der Server eine Anfrage für ein bestimmtes Dokument eines bestimmten virtuellen Hosts empfängt, dann benutzt er die im `<VirtualHost>`-Container enthaltenen Konfigurationsanweisungen. *Adresse* kann sein:

- Die IP-Adresse des virtuellen Hosts.
- Ein voll qualifizierter Domainname für die IP-Adresse des virtuellen Hosts.
- Das Zeichen `*`, welches nur in Kombination mit `NameVirtualHost *` verwendet wird, um allen IP-Adressen zu entsprechen.
- Die Zeichenkette `_default_`, die nur mit IP-basierten virtuellen Hosts verwendet wird, um nicht zugewiesene IP-Adressen aufzufangen.

Beispiel

```
<VirtualHost 10.1.2.3>
   ServerAdmin webmaster@host.foo.com
   DocumentRoot /www/docs/host.foo.com
   ServerName host.foo.com
   ErrorLog logs/host.foo.com-error_log
   TransferLog logs/host.foo.com-access_log
</VirtualHost>
```

IPv6-Adressen müssen in eckigen Klammern angegeben werden, da die optionale Portnummer sonst nicht erkannt werden kann. Hier ein IPv6-Beispiel:

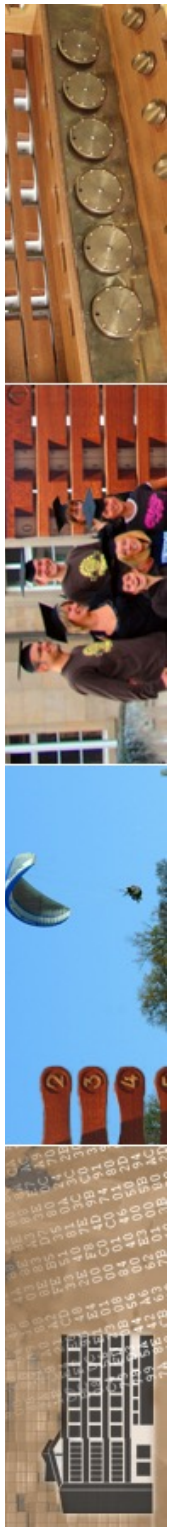
```
<VirtualHost [2001:db8::a00:20ff:fea7:ccea]>
   ServerAdmin webmaster@host.example.com
   DocumentRoot /www/docs/host.example.com
   ServerName host.example.com
   ErrorLog logs/host.example.com-error_log
   TransferLog logs/host.example.com-access_log
</VirtualHost>
```

Jeder virtuelle Host muss einer anderen IP-Adresse, einem anderen Port oder einem anderen Hostnamen für den Server entsprechen. Im ersten Fall muss die Servermaschine so eingerichtet sein, dass sie IP-Pakete für mehrere Adressen akzeptiert. (Wenn der Rechner nicht mehrere Netzwerkkarten besitzt, kann dies mit dem Befehl `ifconfig alias` durchgeführt werden -- sofern Ihr Betriebssystem das unterstützt).

Anmerkung

Die Verwendung von `<VirtualHost>` beeinflusst **nicht**, an welchen Adressen der Apache lauscht. Sie müssen mit `Listen` sicherstellen, dass der Apache an der richtigen Adresse lauscht.



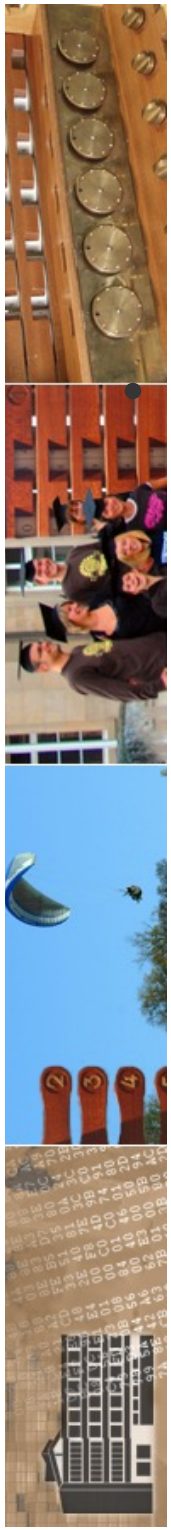


```
zrvwa01@infodienste => more httpd-vhosts.conf
# Virtual Hosts
#
# Required modules: mod_log_config

# If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs/2.4/vhosts/>
# for further details before you try to setup virtual hosts.
#
# You may use the command line option '-S' to verify your virtual host
# configuration.

#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for all requests that do not
# match a ServerName or ServerAlias in any <VirtualHost> block.
#
<VirtualHost *:4711>
    ServerAdmin webmaster@dummy-host.example.com
    DocumentRoot "/home/zrvwa01/apache_test/docs/dummy-host.example.com"
    ServerName dummy-host.example.com
    ServerAlias www.dummy-host.example.com
    ErrorLog "logs/dummy-host.example.com-error_log"
    CustomLog "logs/dummy-host.example.com-access_log" common
</VirtualHost>

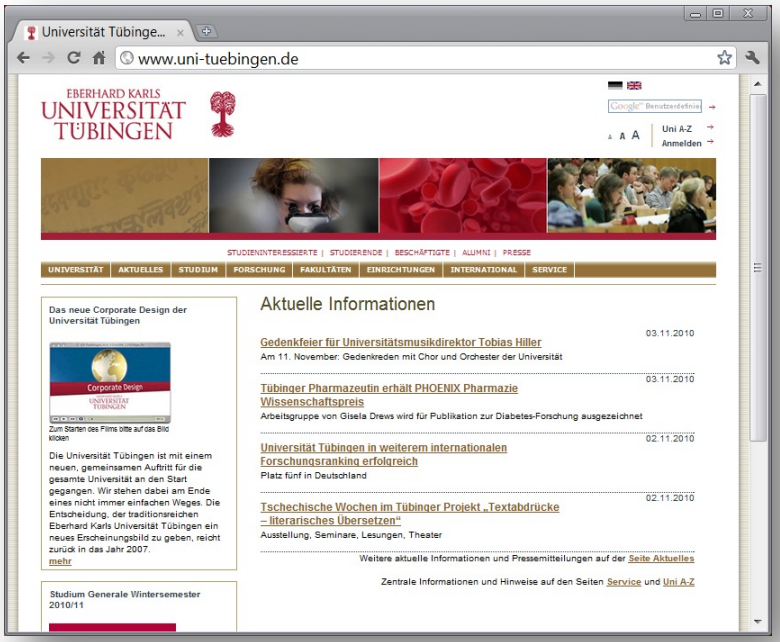
<VirtualHost *:4711>
    ServerAdmin webmaster@dummy-host2.example.com
    DocumentRoot "/home/zrvwa01/apache_test/docs/dummy-host2.example.com"
    ServerName dummy-host2.example.com
    ErrorLog "logs/dummy-host2.example.com-error_log"
    CustomLog "logs/dummy-host2.example.com-access_log" common
</VirtualHost>
```



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

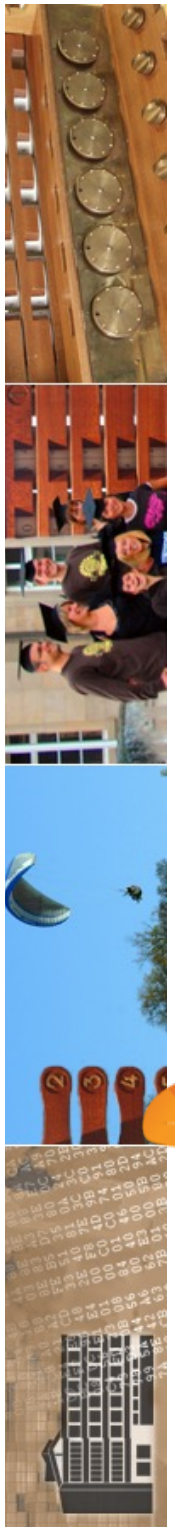
ein konkretes Beispiel

www.uni-tuebingen.de



- besteht zunächst aus zwei Proxy-Servern (Apaches mit mod_proxy)
- diese Verteilen die Anfragen auf acht Apaches jeweils auf einem virtuellen Server
- dort jeweils Apache mit PHP und damit TYPO3
- dahinter eine HA-MySQL-Lösung und ein zentrales nfs-Filesystem
- interne HeartBeat-Kommunikation





…und nun…

- haben wir den Apache-Web-Server genauer kennen gelernt
  - wir können Apache installieren, konfigurieren und haben verstanden, welche Rolle er heute einnimmt und wie er funktioniert



- als nächstes: Der Web-Server NGINX