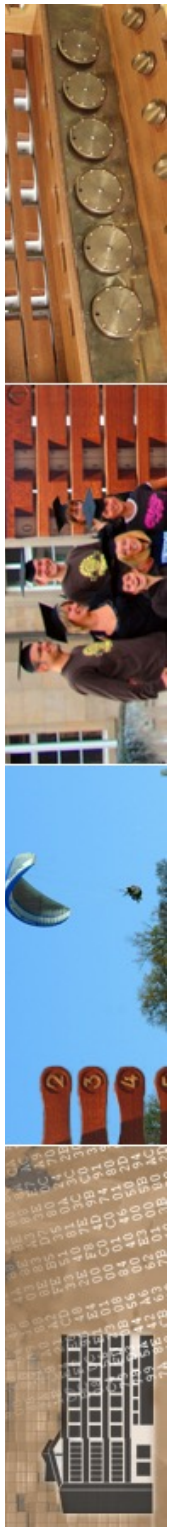


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATHEMATISCH-
NATURWISSENSCHAFTLICHE FAKULTÄT
FACHBEREICH INFORMATIK
INFORMATIONSDIENSTE

Grundlagen der Web-Entwicklung

INF3172

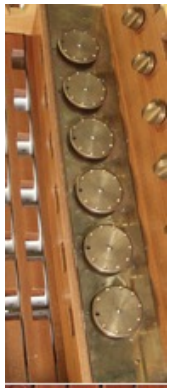
fastCGI und ServerSideIncludes

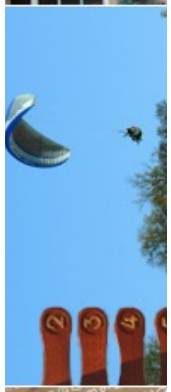
Thomas Walter

20.11.2025

Version 1.0

FastCGI

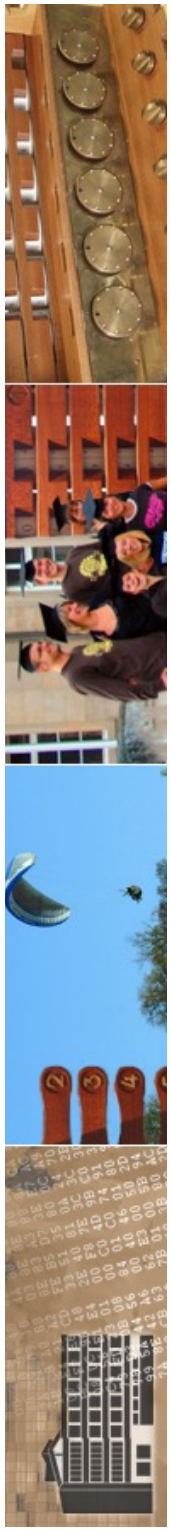





das Weihnachtsrätsel

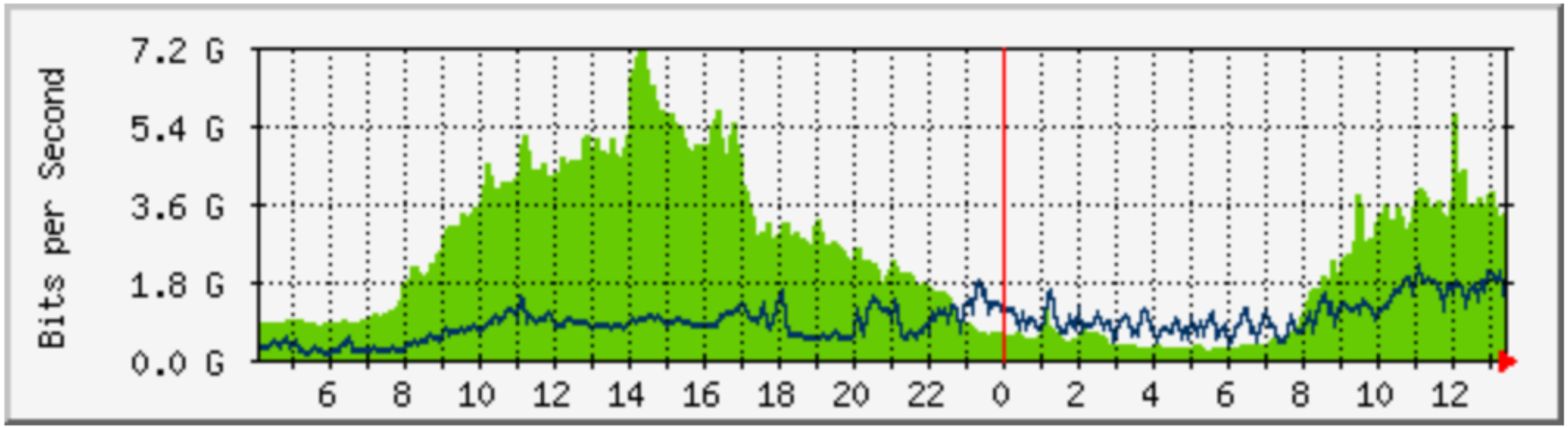
- am **1. Advent 30.11.2025 ab 11.00h:** das Weihnachtsrätsel

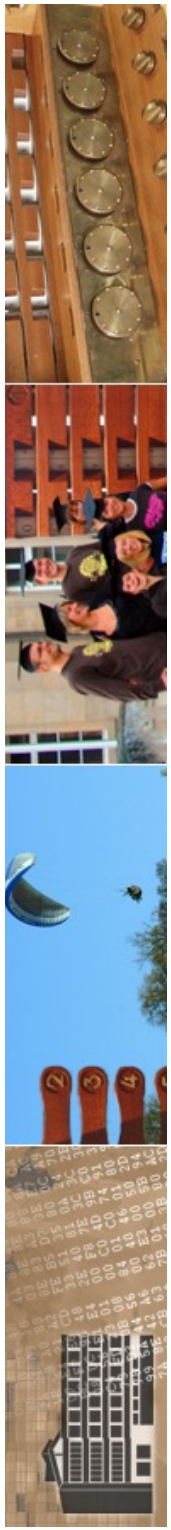
- ~~Preise für schnellste Lösung und Verlosung weiterer~~
- ~~Auflösung und Verlosung der Gewinne in der Vorlesung vor Weihnachten~~



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Fußball-WM und Netzwerk





Studie: Unternehmen an Feiertagen und Wochenenden anfälliger für Cyberangriffe

Gerade an Wochenenden und Feiertagen erleben Unternehmen häufig Cyberangriffe. Das geht aus einer Befragung hervor.

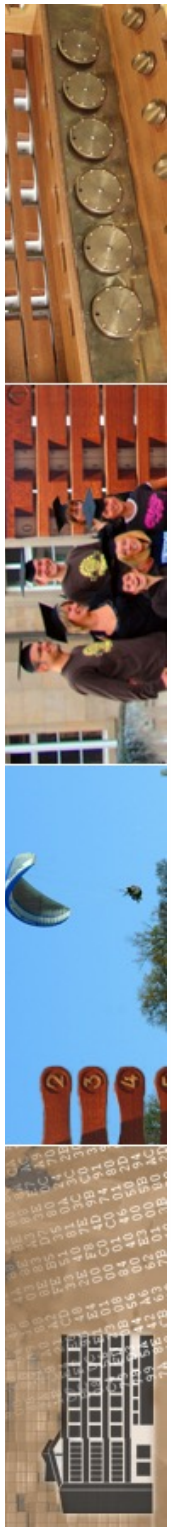
11

(Bild: BeeBright / Shutterstock.com)

13:22 Uhr **Lesezeit: 2 Min.** | **Security**

Von Marie-Claire Koch

Unternehmen sind an Feiertagen und Wochenenden besonders gefährdet für Cyberangriffe, da das Sicherheitspersonal in dieser Zeit oft reduziert ist. Das bestätigt jetzt auch eine neue Studie zu Ransomware-Angriffen von Semperis, einem Anbieter im Bereich identitätsbasierter Cyber-Resilienz. Demnach wurden im Durchschnitt 86 Prozent der befragten Unternehmen aus den USA, Großbritannien, Frankreich und Deutschland an Feiertagen oder am Wochenende angegriffen.



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

SUPPORT SERVICE-KATALOG FORSCHUNG UND LEHRE DAS URZ

URZ UNIVERSITÄTS-
RECHENZENTRUM

UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

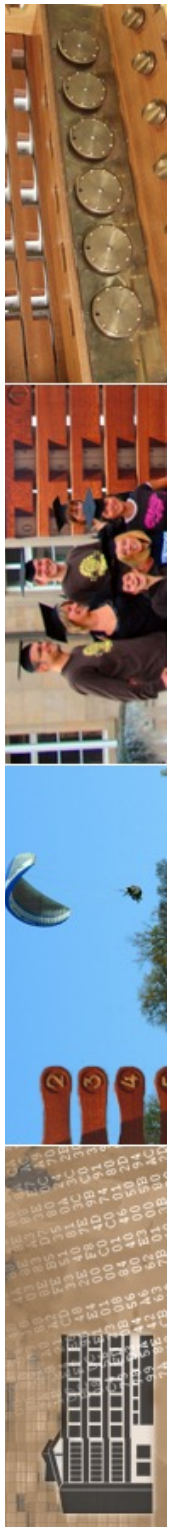
BETRIEBSMELDUNG - 19. NOVEMBER 2025

ANGRIFF AUF DIE IT-INFRASTRUKTUR: VORSICHTSMASSNAHMEN

ZUR SOFORTIGEN UMSETZUNG

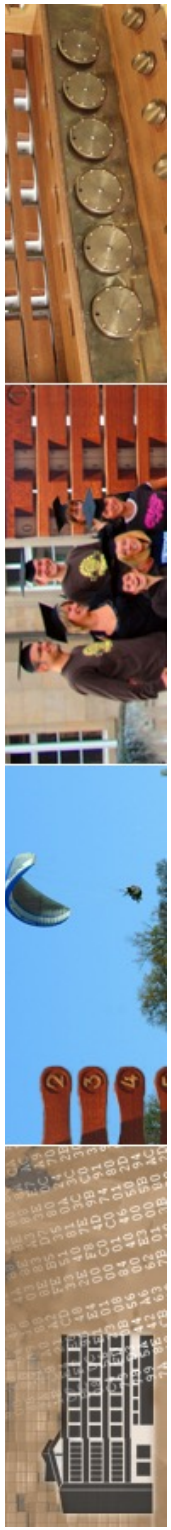
Die Universität Heidelberg hat Vorbereitungen auf einen weitreichenden Angriff auf die IT-Infrastruktur identifiziert. Die Abwehr erfordert die **sofortige Umsetzung notwendiger Vorsichtsmaßnahmen**. Diese Maßnahmen beinhalten unter anderem eine umgehende **Änderung der Passwörter.** Zudem sind ab sofort zahlreiche I**T-Dienste nur noch über VPN oder aus dem Campusnetzwerk erreichbar.**
Es wird mit Nachdruck daran gearbeitet, Schaden von der Universität abzuwenden. Für technische Rückfragen wenden Sie sich bitte an den IT-Service.

weitere Informationen
Vorsichtsmaßnahmen →



FastCGI

- Ansatz zum *wesentlichen Beschleunigen von Server-Anwendungen*
- hat sich zunächst selbst nicht direkt durchgesetzt, aber *Basis für viele ähnlichen Techniken* gesetzt
- für Ruby on Rails sehr wichtig
- *seit Apache 2.4 Standardmodul*

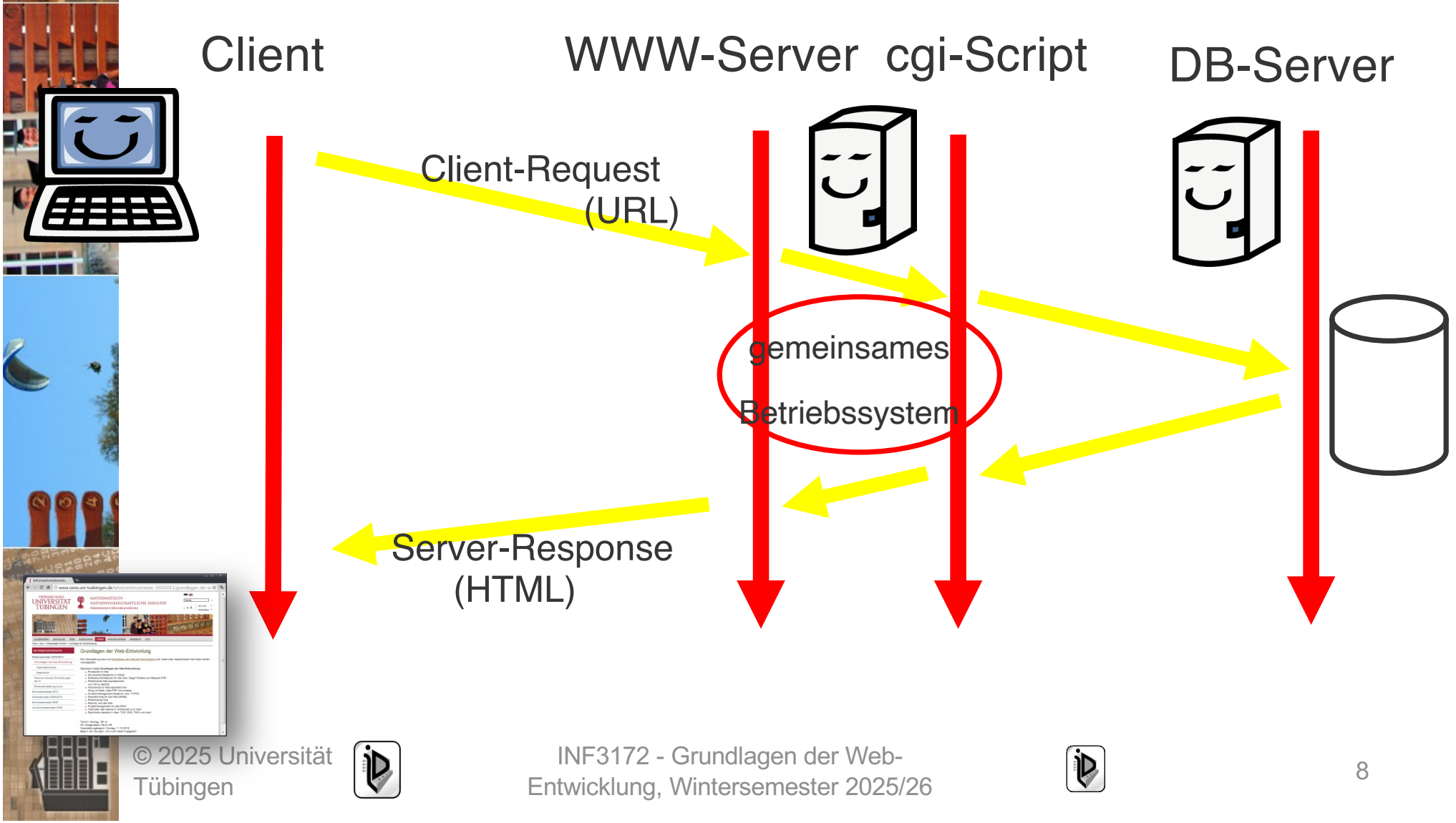


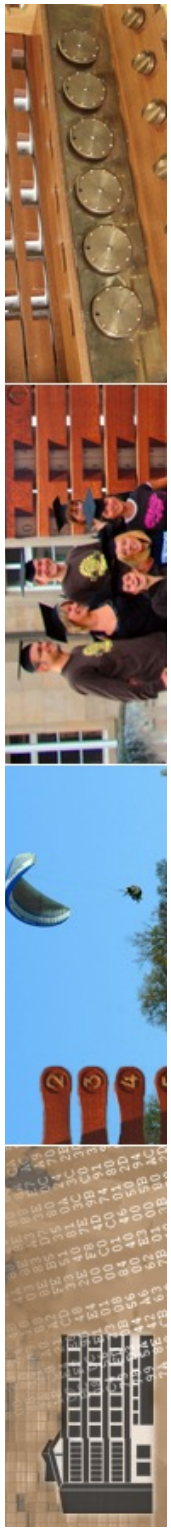
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

CGI Serverstruktur

- es sind **drei** Rechner im Spiel:
  - der Web-Client, der den HTTP-Request sendet
  - der Webserver, der zugleich auch den CGI-Prozeß ausführen muß
  - optional der Datenbankserver
  - je nach Fall sind es weniger Server, insbesondere ist häufig der Datenbank- und der Webserver die gleiche Maschine

Struktur einer HTTP-Transaktion mit cgi *und Datenbank*

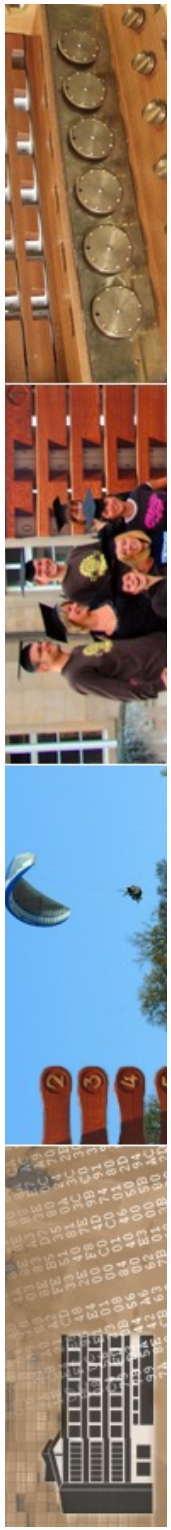




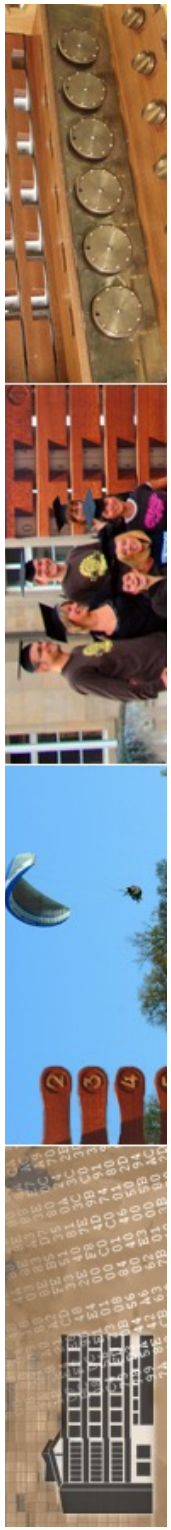
CGI: Umgebungsvariablen

- Server legt bei Aufruf des cgi-Programms spezielle **Umgebungsvariablen** fest, die dem cgi-Programm die notwendigen Informationen liefern
  (interner Mechanismus der Datenübergabe)

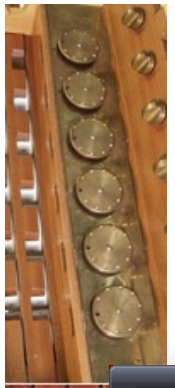
  einige dieser Umgebungsvariablen:
  - **`SERVER_NAME, SERVER_PROTOCOL, SERVER_PORT, ...`**
  - **`PATH_INFO, SCRIPT_NAME, QUERY_NAME`**
  - **`REMOTE_HOST, REMOTE_ADDR, REMOTE_USER`**

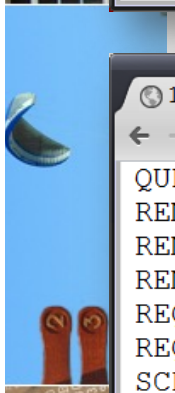
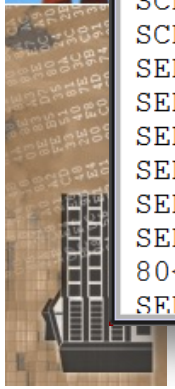


```
zrvwa01@infodienste =>
zrvwa01@infodienste => cd cgi-bin/
zrvwa01@infodienste =>
zrvwa01@infodienste =>
zrvwa01@infodienste => ll
insgesamt 16
drwxr-xr-x  2 zrvwa01   74 27. Okt 09:31 ./
drwxr-xr-x 14 zrvwa01  152 27. Okt 09:31 ../
-rw-r--r--  1 zrvwa01  820 17. Dez 2012  printenv
-rw-r--r--  1 zrvwa01 1074 17. Dez 2012  printenv.vbs
-rw-r--r--  1 zrvwa01 1133 17. Dez 2012  printenv.wsf
-rw-r--r--  1 zrvwa01 1261 17. Dez 2012  test-cgi
zrvwa01@infodienste =>
zrvwa01@infodienste =>
zrvwa01@infodienste =>
```



```
[zrvwa01@infodienste:~/apache_test/cgi-bin$
[zrvwa01@infodienste:~/apache_test/cgi-bin$
[zrvwa01@infodienste:~/apache_test/cgi-bin$
[zrvwa01@infodienste:~/apache_test/cgi-bin$
[zrvwa01@infodienste:~/apache_test/cgi-bin$ more printenv
#

# To permit this cgi, replace # on the first line above with the
# appropriate #!/path/to/perl shebang, and on Unix / Linux also
# set this script executable with chmod 755.
#
# ***** !!! WARNING !!! *****
# This script echoes the server environment variables and therefore
# leaks information - so NEVER use it in a live server environment!
# It is provided only for testing purpose.
# Also note that it is subject to cross site scripting attacks on
# MS IE and any other browser which fails to honor RFC2616.

##
##  printenv -- demo CGI program which just prints its environment
##
use strict;
use warnings;

print "Content-type: text/plain; charset=iso-8859-1\n\n";
foreach my $var (sort(keys(%ENV))) {
    my $val = $ENV{$var};
    $val =~ s|\n|\\n|g;
    $val =~ s|"|\\"|g;
    print "${var}=\"${val}\"\n";
}

[zrvwa01@infodienste:~/apache_test/cgi-bin$
[zrvwa01@infodienste:~/apache_test/cgi-bin$
[zrvwa01@infodienste:~/apache_test/cgi-bin$
```



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Script printenv.pl

127.0.0.1/cgi-bin/print ×

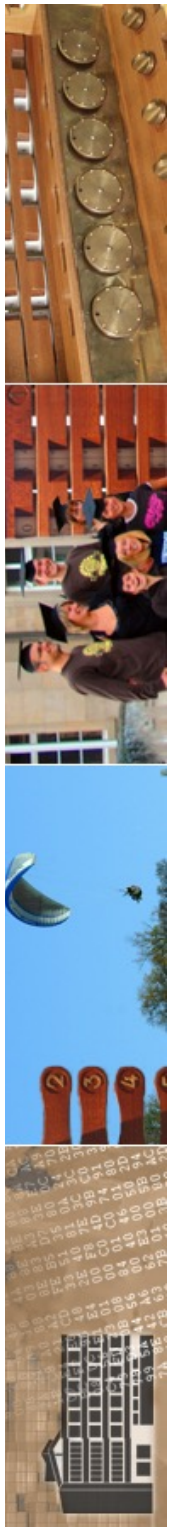
127.0.0.1/cgi-bin/printenv.pl

```
DOCUMENT_ROOT="D:/www"
GATEWAY_INTERFACE="CGI/1.1"
HTTP_ACCEPT="text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8"
HTTP_ACCEPT_CHARSET="ISO-8859-1,utf-8;q=0.7,*;q=0.3"
HTTP_ACCEPT_ENCODING="gzip,deflate,sdch"
HTTP_ACCEPT_LANGUAGE="de-DE,de;q=0.8,en-US;q=0.6,en;q=0.4"
HTTP_CONNECTION="keep-alive"
HTTP_HOST="127.0.0.1"
HTTP_USER_AGENT="Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.1 (KHTML, like Gecko)
```

127.0.0.1/cgi-bin/print ×

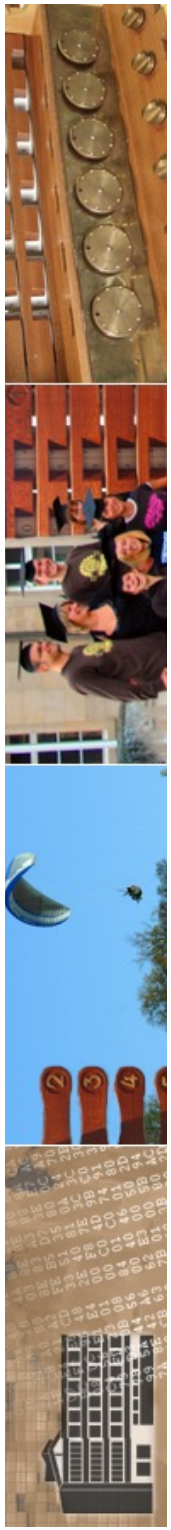
127.0.0.1/cgi-bin/printenv.pl

```
QUERY_STRING=""
REMOTE_ADDR="127.0.0.1"
REMOTE_HOST="mouton"
REMOTE_PORT="58323"
REQUEST_METHOD="GET"
REQUEST_URI="/cgi-bin/printenv.pl"
SCRIPT_FILENAME="C:/Program Files (x86)/Apache Software Foundation/Apache2.2/cgi-bin/printenv.pl"
SCRIPT_NAME="/cgi-bin/printenv.pl"
SERVER_ADDR="127.0.0.1"
SERVER_ADMIN="thomas.walter@uni-tuebingen.de"
SERVER_NAME="127.0.0.1"
SERVER_PORT="80"
SERVER_PROTOCOL="HTTP/1.1"
SERVER_SIGNATURE="<address>Apache/2.2.15 (Win32) PHP/5.3.2 Server at 127.0.0.1 Port
80</address>\n"
SERVER_SOFTWARE="Apache/2.2.15 (Win32) PHP/5.3.2"
```



Vorteile CGI

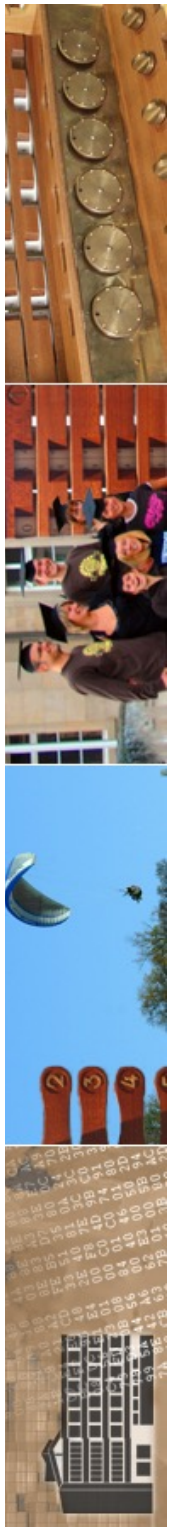
- einfach
- Sprachunabhängigkeit
- offener Standard
- Architekturunabhängigkeit
- Kapselung in einzelnen Prozessen
- läuft mit den meisten (allen?) Webservern, wenn entsprechend konfiguriert



Nachteile CGI

- bei jeder Anfrage wird CGI-Prozeß gestartet, nach abarbeiten wieder beendet
  - Beispielsweise wird bei jeder Anfrage für ein Python-CGI der Python-Interpreter neu gestartet
- ➔ keine optimale Performance

- Idee: »**Daemonize It**«
  - Anstelle eines einzelnen Prozesses **Hintergrundprozess** (Daemon) verwenden
  - Daemon wird nur **einmal** gestartet und wartet dauerhaft auf Anfragen

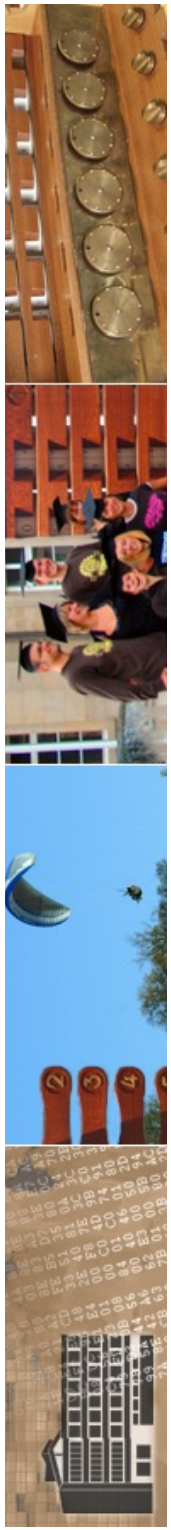


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Grundkonzept fastCGI

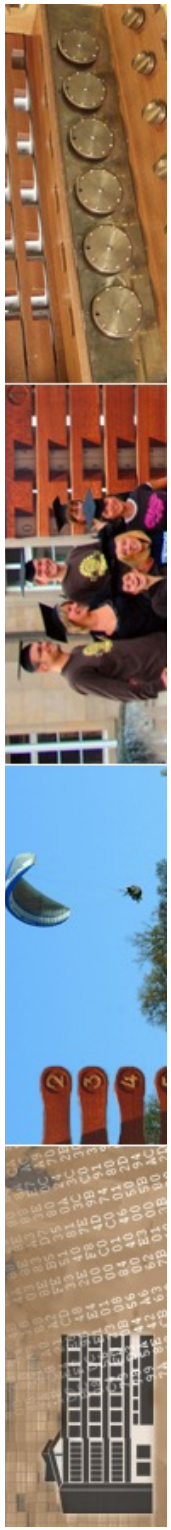
- das um 1995 eingeführte fastCGI-Konzept greift an dieser Stelle die Vorteile von CGI auf, um einen Schritt weiter zu gehen.

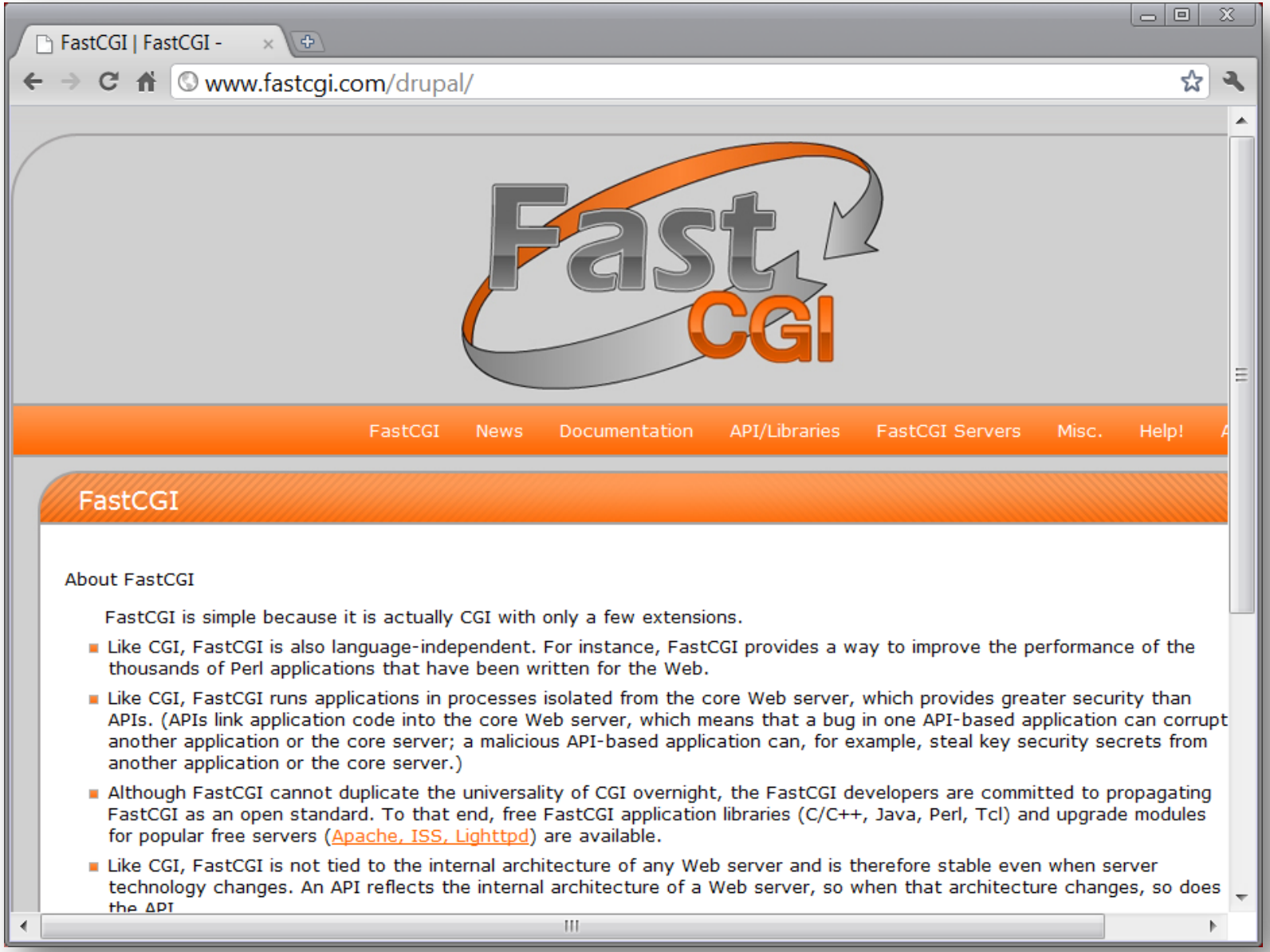
- Wir wollen folgendes erreichen:
  - höhere Performance der CGIs
  - Trennung von Webserver und Applikationsserver (CGI-Server), um mehr Skalierungsmöglichkeiten zu haben

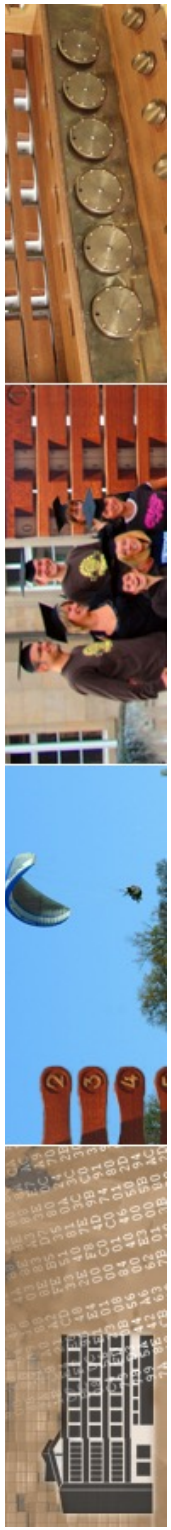


fastCGI

- entwickelt durch Open Market, Inc.
  - **`http://www.fastcgi.com`**
  - **`http://www.openmarket.com/fastcgi`**
- damit auch Trennung von Webserver und cgi-Server möglich, dadurch ideal *skalierbar*
  - cgi: Webserver und CGI kommunizieren über pipes und Umgebungsvariablen: ein einzelner Server für beides
  - fastcgi: Webserver und fastcgi kommunizieren über tcp-Socket: **Trennung der Server möglich**
  - »Umwandlung« eines CGI-Programmes notwendig




FastCGI | FastCGI -
www.fastcgi.com/drupal/
Fast CGI
FastCGI
News
Documentation
API/Libraries
FastCGI Servers
Misc.
Help!
FastCGI
About FastCGI
FastCGI is simple because it is actually CGI with only a few extensions.
Like CGI, FastCGI is also language-independent. For instance, FastCGI provides a way to improve the performance of the thousands of Perl applications that have been written for the Web.
Like CGI, FastCGI runs applications in processes isolated from the core Web server, which provides greater security than APIs. (APIs link application code into the core Web server, which means that a bug in one API-based application can corrupt another application or the core server; a malicious API-based application can, for example, steal key security secrets from another application or the core server.)
Although FastCGI cannot duplicate the universality of CGI overnight, the FastCGI developers are committed to propagating FastCGI as an open standard. To that end, free FastCGI application libraries (C/C++, Java, Perl, Tcl) and upgrade modules for popular free servers (Apache, ISS, Lighttpd) are available.
Like CGI, FastCGI is not tied to the internal architecture of any Web server and is therefore stable even when server technology changes. An API reflects the internal architecture of a Web server, so when that architecture changes, so does the API

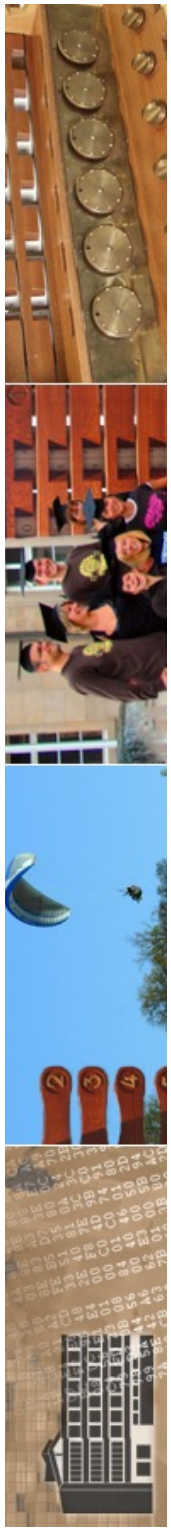


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

fastCGI

- »fastCGI is a fast, open, and secure Web server interface that solves the performance problems inherent in CGI, without introducing the overhead and complexity of proprietary APIs.«
  - »proprietary APIs«: Anwendungsentwicklung direkt für den jeweiligen Webserver, etwa als Apache-Modul

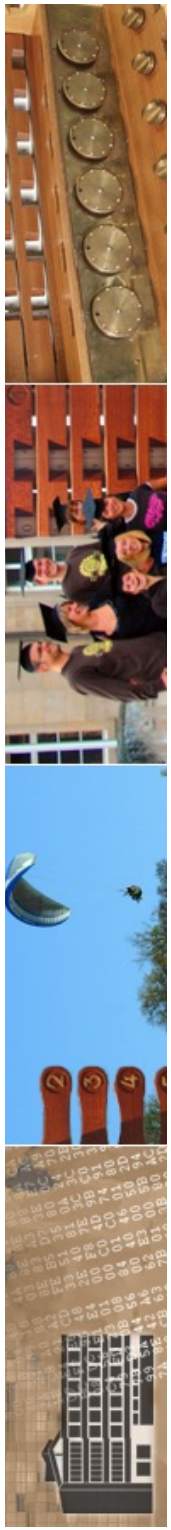
FastCGI



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

fastCGI

- Vorteile von fastCGI:
  - Performance
  - Einfachheit, mit einfacher Migration von CGI
    - nur unwesentlich komplexer als native CGI
  - Sprachunabhängigkeit
  - Prozeßisolation
  - offener, nichtproprietärer Standard
  - Architekturunabhängigkeit
  - unterstützt verteilte Systeme

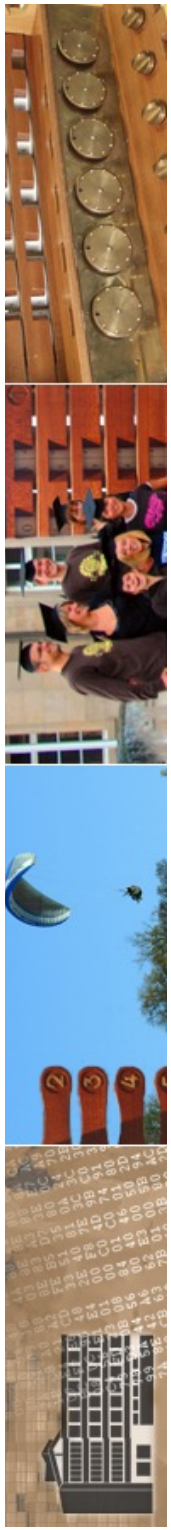


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Unterschied CGI zu fastCGI

- A: fastCGI-Prozesse sind **persistent**: Nach Beendigung einer Anfrage werden sie *nicht* beendet, sondern warten auf nächste Anfrage

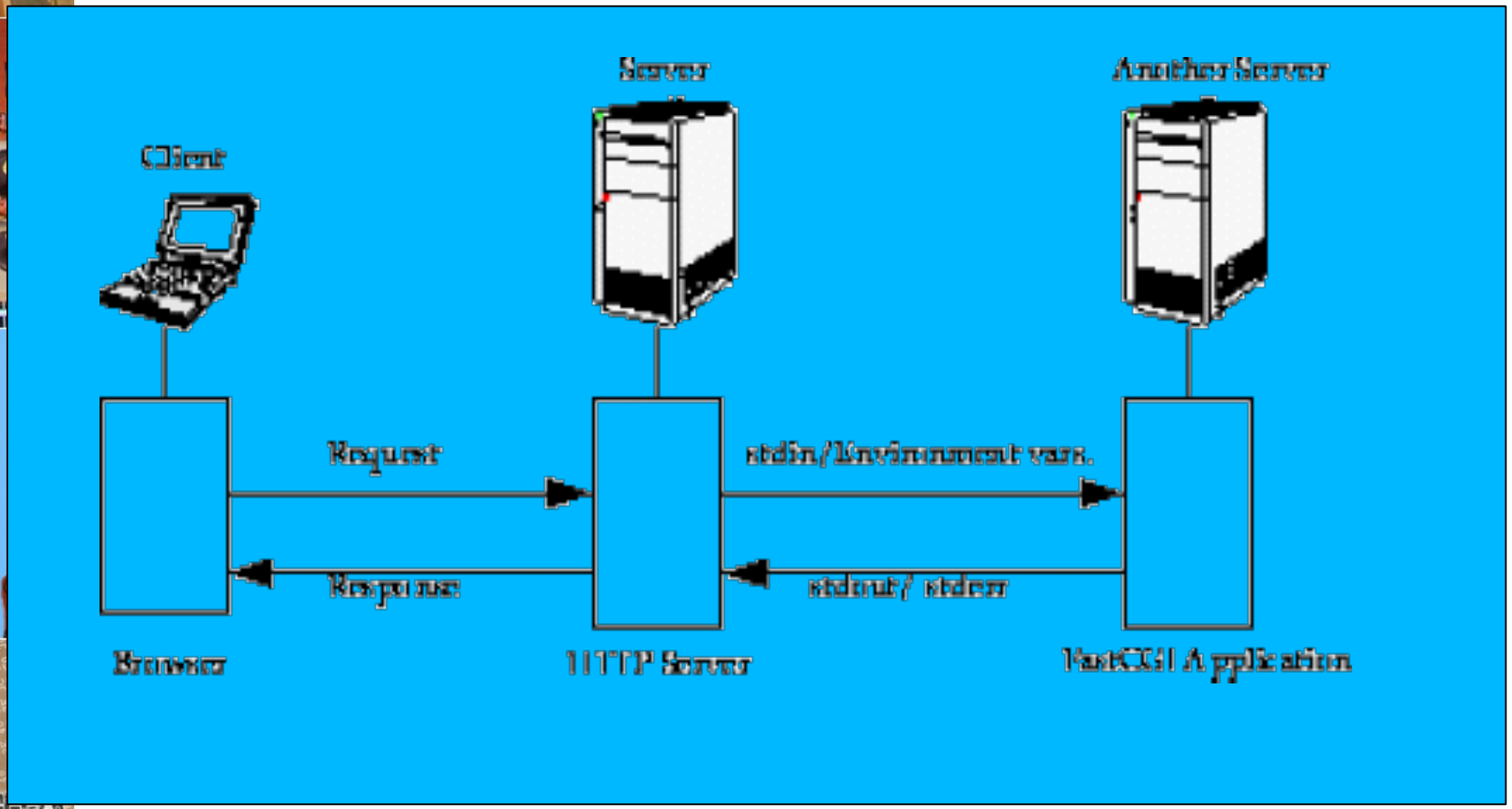
- B: fastCGI verwendet **keine Umgebungsvariablen** und *keine* pipes zur Kommunikation mit dem Webserver, sondern tcp-Verbindung; als Folge können Webserver und fastCGI-Server auch getrennt werden

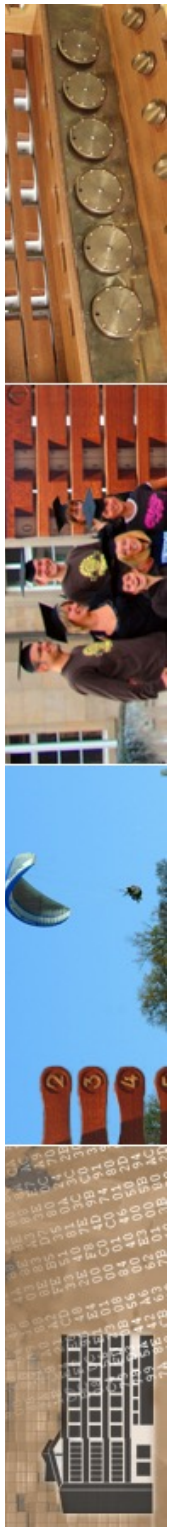


typischer Ablauf

- Webserver erzeugt fastCGI-Anwendungsprozeß
  - beim Start des Webservers oder bei erster Anfrage
- fastCGI-Prozeß initialisiert sich selbst und wartet auf Anfrage
- bei Client-Anfrage an Webserver: Webserver baut Verbindung zu fastCGI-Prozeß auf und sendet darüber CGI-Umgebungsvariablen und stdin
- fastCGI sendet stdout und stderr entsprechend zurück
- fastCGI wartet auf weitere Anfrage

Schema fastCGI

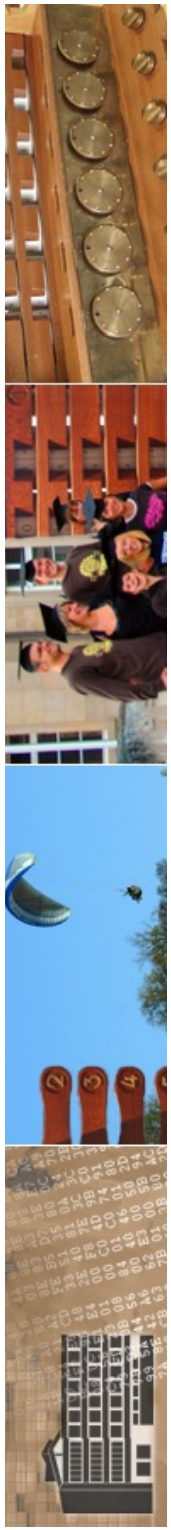




EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

fastCGI-Protokoll

- fastCGI verwendet ein spezielles Protokoll zur Kommunikation zwischen Webserver und Prozess
  - **`FCGI_PARAMS`**: name/value-Paare für CGI-Umgebungsvariablen
  - **`FCGI_STDIN`**
  - **`FCGI_STDOUT`**
  - **`FCGI_STDERR`**
  - **`FCGI_END_REQUEST`**: Ende der Anfrage
  - ...und weitere, siehe Web



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

für fastCGI notwendig...

- ...ist das (frei verfügbare) »**fastCGI Developer's Kit**«

- aktuell Version 2.4.6 (13.11.2007)
  - umfaßt viele Tools
  - u.a. fastCGI-Module für PERL, Python, C und Java
  - Online-Dokumentation
  - Protokoll-Beschreibung

  - https://fastcgi-archives.github.io/FastCGI_Developers_Kit_FastCGI.html

FastCGI Developer's Kit

This FastCGI Developer's Kit is designed to make developing FastCGI applications easy.
The kit currently supports FastCGI applications written in C/C++, Perl, Tcl, and Java.

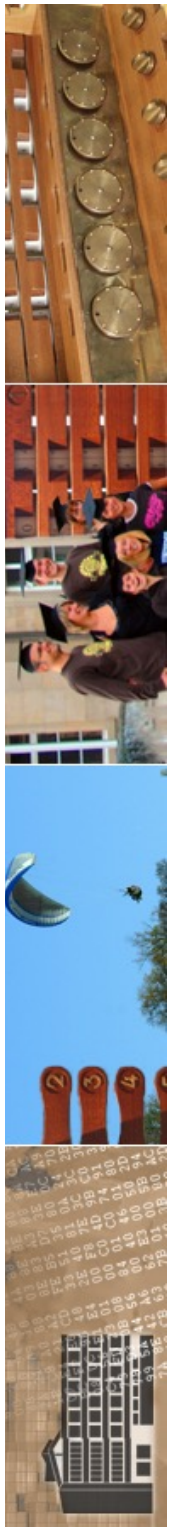
View on GitHub

Mark R. Brown

Open Market, Inc.

Document Version: 1.08
11 June 1996

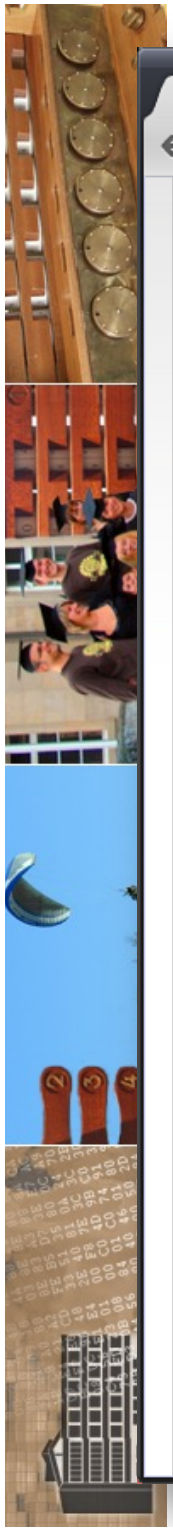
- 1. Introduction
- 2. Getting started
- 3. Writing applications
  - 3.1 Using the `fcgi_stdio` library
  - 3.2 Using the `fcgiapp` library
  - 3.3 Using Perl and Tcl
  - 3.4 Using Java



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

der Webserver und fastCGI

- die Nutzung von fastCGI erfordert serverseitige Voraussetzungen:
  - das entsprechende Modul muss vorhanden sein (für Apache **mod_fastcgi**)
  - der Server muß entsprechend konfiguriert sein
  - fastCGI ist mit zahlreichen Webservern kombinierbar
    - Apache
    - nginx
    - Microsoft IIS, ...




Module mod_fastcg...
www.fastcgi.com/drupal/node/25
Module mod_fastcgi
APACHE
Module mod_fastcgi
This 3rd party module provides support for the FastCGI protocol. FastCGI is a language independent, scalable, open extension to CGI that provides high performance and persistence without the limitations of server specific APIs.
FastCGI applications are not limited to a particular development language (the protocol is open). FastCGI application libraries currently exist for Perl, C/C++, Java, Python, TCL, SmallEiffel, and Smalltalk.
FastCGI applications use TCP or Unix sockets to communicate with the web server. This scalable architecture allows applications to run on the same platform as the web server or on many machines scattered across an enterprise network.
FastCGI applications are portable to other web server platforms. FastCGI is supported either directly or through commercial extensions by most popular web servers.
FastCGI applications are fast because they're persistent. There is no per-request startup and initialization overhead. This makes possible the development of applications which would otherwise be impractical within the CGI paradigm (e.g. a huge Perl script, or an application which requires a connection to one or more databases).
See the FastCGI website for more information. To receive FastCGI related announcements and notifications of software updates, subscribe to fastcgi-announce. To participate in the discussion of mod_fastcgi and FastCGI application development, subscribe to fastcgi-developers.
Summary
For information about building and installing the module, see the INSTALL document that came with the distribution.
FastCGI applications under mod_fastcgi are defined as one of three types: static, dynamic, or external. They're configured using the FastCgiServer, FastCgiConfig, and FastCgiExternalServer directives respectively. Any URI that Apache identifies as a FastCGI application and which hasn't been explicitly configured using a FastCgiServer or FastCgiExternalServer directive is handled as a dynamic application (see the FastCgiConfig directive for more information).
FastCGI static and dynamic applications are spawned and managed by the FastCGI Process Manager, fcgi-pm. The process manager is spawned by Apache at server initialization. External applications are presumed to be started and managed independently.
Apache must be configured to identify requests for FastCGI URIs. mod_fastcgi registers (with Apache) a handler type of fastcgi-script for this purpose.
To configure Apache to handle all files (within the scope of the directive) as FastCGI applications (e.g. for a fcgi-bin directory):

Apache Module mod_proxy_fcgi

Available Languages: en | fr

Description:	FastCGI support module for mod_proxy
Status:	Extension
Module Identifier:	proxy_fcgi_module
Source File:	mod_proxy_fcgi.c
Compatibility:	Available in version 2.3 and later

Summary

This module *requires* the service of mod_proxy. It provides support for the FastCGI protocol.

Thus, in order to get the ability of handling the FastCGI protocol, mod_proxy and mod_proxy_fcgi have to be present in the server.

Unlike mod_fcgid and mod_fastcgi, mod_proxy_fcgi has no provision for starting the application process; fcgistarter is provided (on some platforms) for that purpose. Alternatively, external launching or process management may be available in the FastCGI application framework in use.

Warning

Do not enable proxying until you have secured your server. Open proxy servers are dangerous both to your network and to the Internet at large.

Examples

Remember, in order to make the following examples work, you have to enable mod_proxy and mod_proxy_fcgi.

Single application instance

```
ProxyPass "/myapp/" "fcgi://localhost:4000/"
```

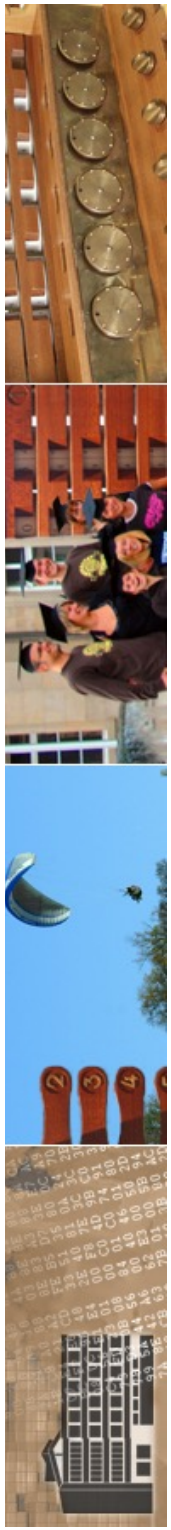
mod_proxy_fcgi disables connection reuse by default, so after a request has been completed the connection will NOT be

Topics
- Examples
- Environment Variables

Directives
ProxyFCGIBackendType
ProxyFCGISetEnvIf

Bugfix checklist
- httpd changelog
- Known issues
- Report a bug

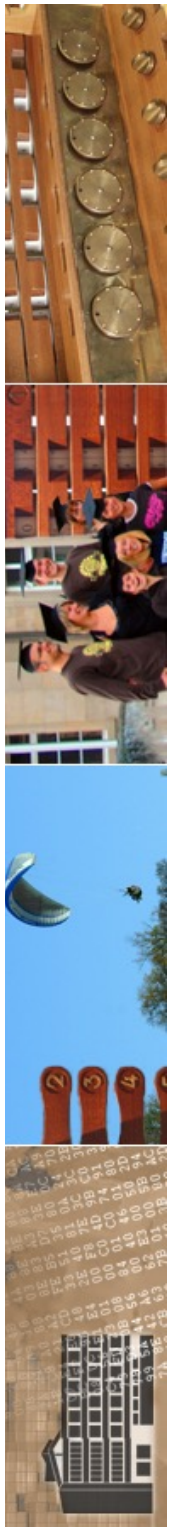
See also
- fcgistarter
- mod_proxy
- mod_authnz_fcgi
- Comments



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Konfiguration des Apache

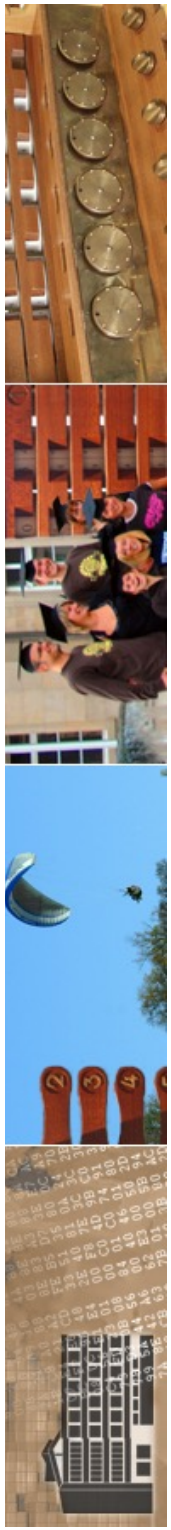
- **`SetHandler`** und **`AddHandler`** für fastCGI
- **`fastCGI-Server`**
  (mit sehr zahlreichen Optionen)
- es gibt einige weitere Direktiven für fastCGI:
  - **`FastCgiExternalServer`**
  - **`FastCgiConfig`**
  - **`FastCgiAuthenticator`**
  - **`FastCgiAuthenticatorAuthoritative`**
  - **`FastCgiAuthorizer`**
  - `...`



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

was passiert nun?

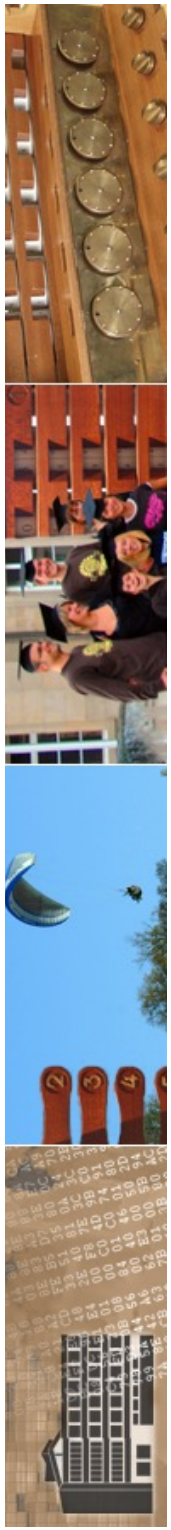
- beim Starten von Apache startet der FastCGI-Prozess **automatisch mit**
- der FastCGI-Prozess wird vom Prozeßmanager des mod_fastcgi gesteuert!!!! (konfigurierbar)
- ...jeweils, wenn beides auf einer Maschine, ansonsten separates Starten des FastCGI-Daemons



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

der fastCGI-Prozeßmanager für Apache

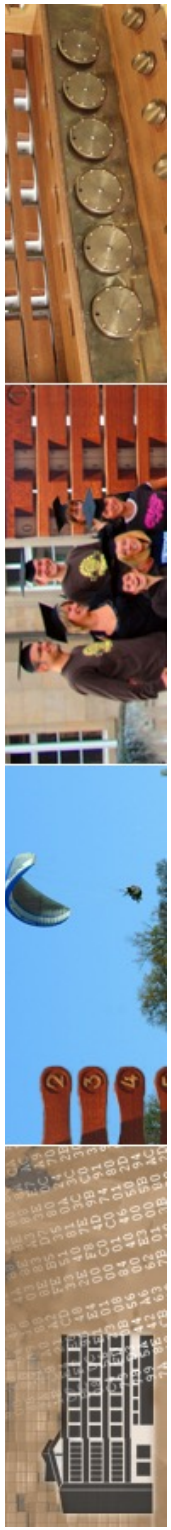
- der Prozessmanager startet den fastCGI-Prozess und achtet darauf, daß dieser „durchläuft“
  - ...ist auch noch konfigurierbar...
- beim Beenden des Webservers wird auch fastCGI-Prozess beendet



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

fastCGI selber programmieren

- fastCGI ist mit allen für CGI »gängigen« Programmiersprachen kombinierbar (alles notwendige ist Bestandteil des Developer's Kits):
  - C/C++
  - **PERL**
  - **Java**
  - Schema
  - Eiffel
  - **Python**
  - **Ruby**
  - TCL
  - Smalltalk



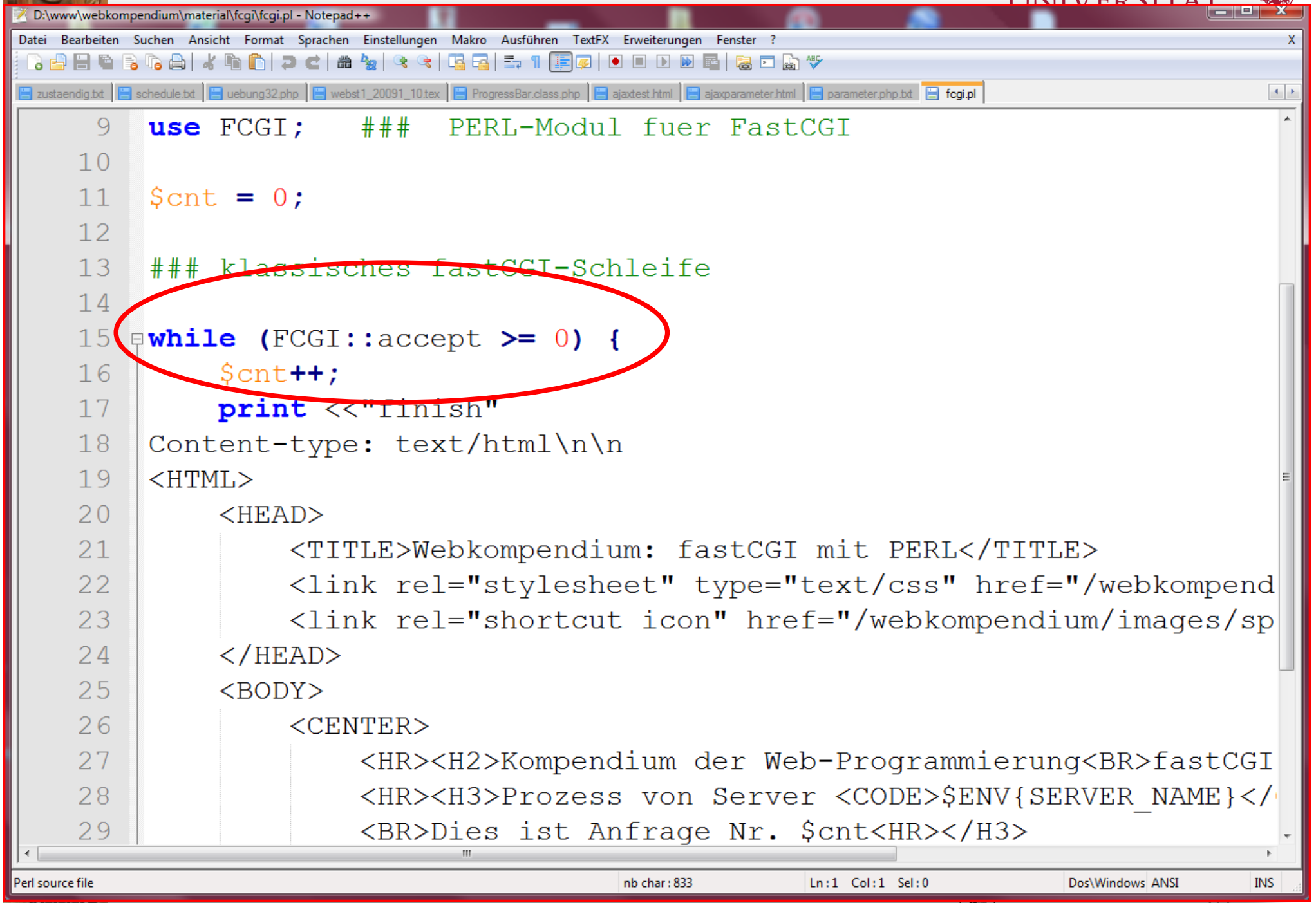
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

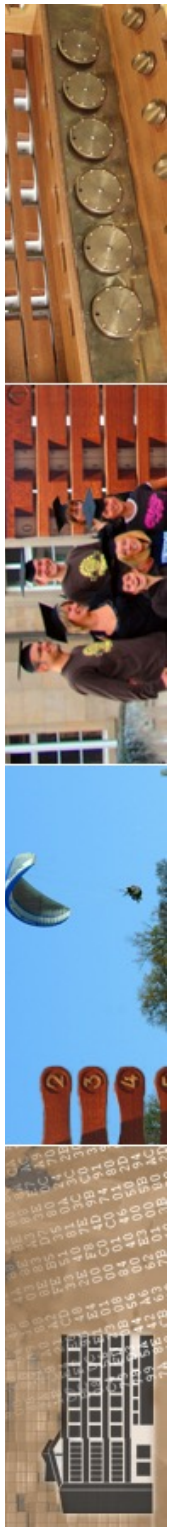
Entwicklung von fastCGI-Anwendungen

- Bestehende CGI-Applikationen können *nicht* direkt übernommen werden, sondern müssen etwas angepasst werden
  - Einbindung der entsprechenden fastCGI-Libraries
  - typisch *Endlos-Schleife* der Art:

```
Initialisierung;

while (FCGI_Accept() >= 0 )  {
   Process request;
}
```

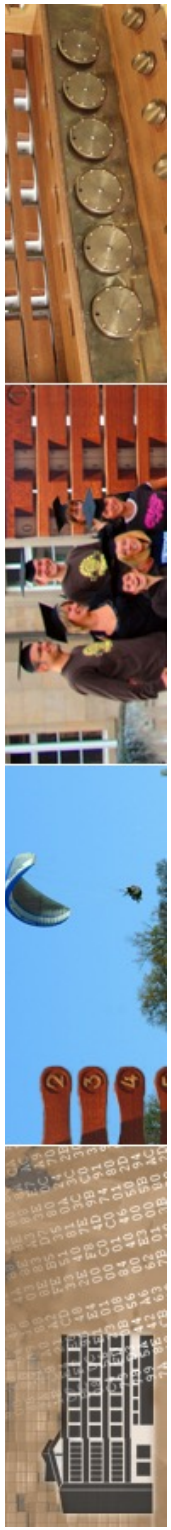
  - Schleife wartet »für immer« auf neue fastCGI-Anfrage


D:\www\webkompendium\material\fcgi\fcgi.pl - Notepad++
Datei Bearbeiten Suchen Ansicht Format Sprachen Einstellungen Makro Ausführen TextFX Erweiterungen Fenster ?
zustaendig.txt schedule.txt uebung32.php webst1_20091_10.tex ProgressBar.class.php ajaxtest.html ajaxparameter.html parameter.php.txt fcgi.pl
9 use FCGI; ### PERL-Modul fuer FastCGI
10
11 $cnt = 0;
12
13 ### klassisches fastCGI-Schleife
14
15 while (FCGI::accept >= 0) {
16 $cnt++;
17 print <<"finish"
18 Content-type: text/html\n\n
19 <HTML>
20 <HEAD>
21 <TITLE>Webkompendium: fastCGI mit PERL</TITLE>
22 <link rel="stylesheet" type="text/css" href="/webkompend
23 <link rel="shortcut icon" href="/webkompendium/images/sp
24 </HEAD>
25 <BODY>
26 <CENTER>
27 <HR><H2>Kompendium der Web-Programmierung
fastCGI
28 <HR><H3>Prozess von Server <CODE>$ENV{SERVER_NAME}</
29
Dies ist Anfrage Nr. $cnt<HR></H3>
Perl source file
nb char : 833
Ln : 1 Col : 1 Sel : 0
Dos\Windows ANSI INS



Performancevorteile durch fastCGI

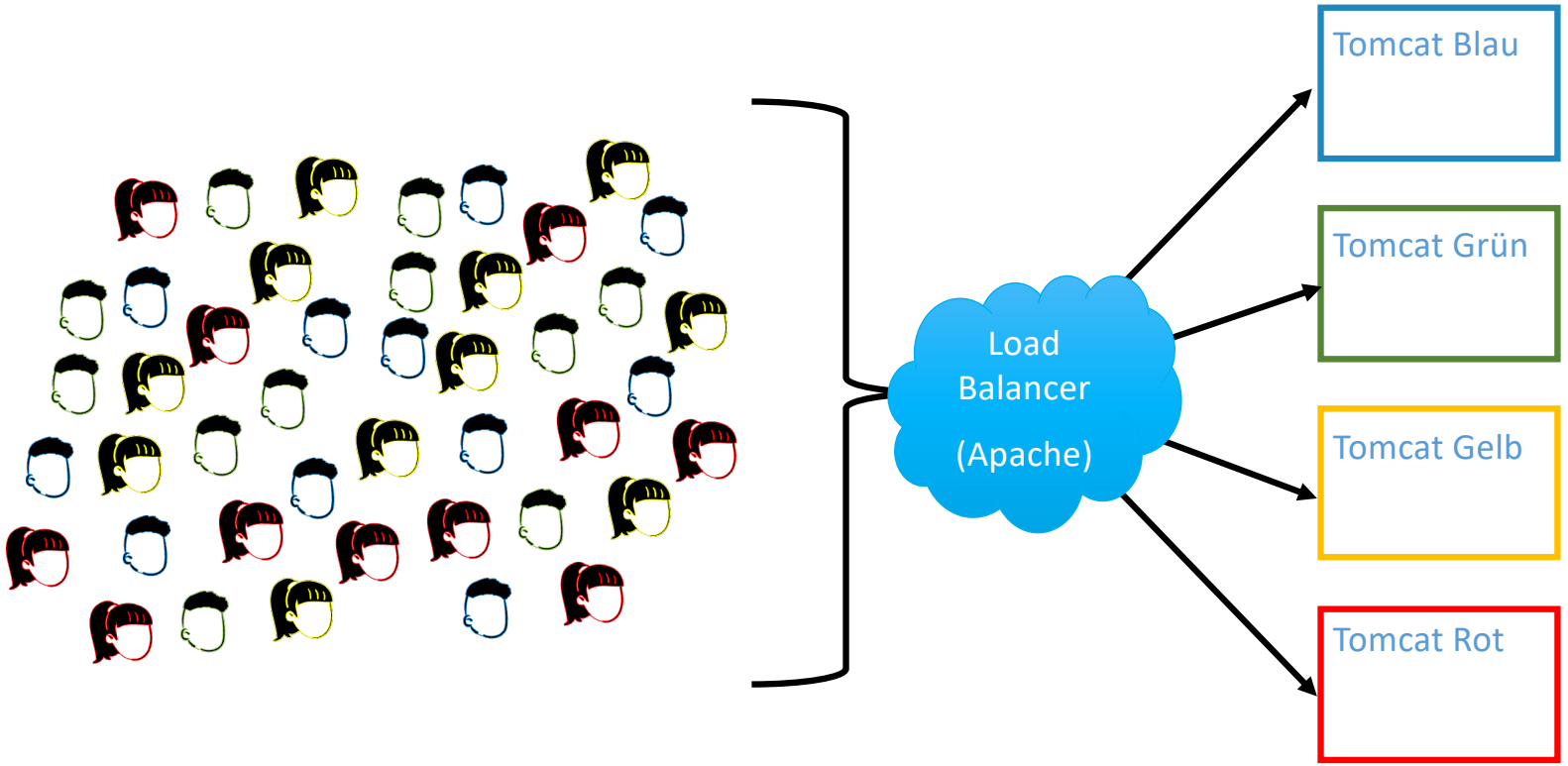
- Literaturangabe:
  - durch Einsatz von FastCGI werden Webanwendungen um bis zu Faktor 5 schneller
  - Beispiel (Herstellerangabe):
    - statische Datei: 21 ms + 0.19 ms / kB
    - fastCGI: 22 ms + 0.28 ms / kB
    - CGI: 59 ms + 0.37 ms / kB

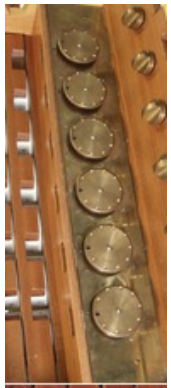


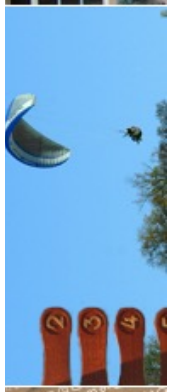
neue Einsatzbereiche

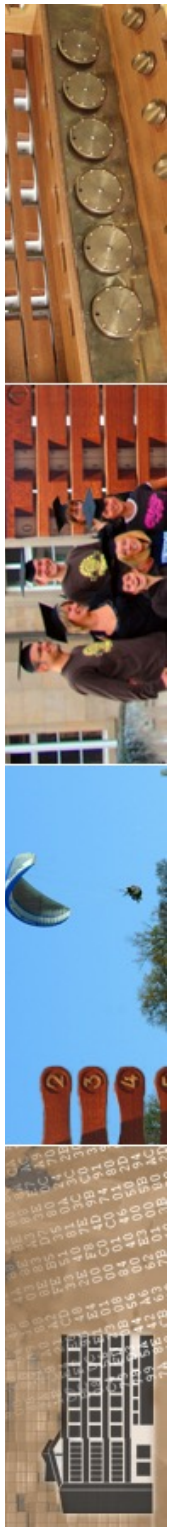
- fastCGI hatte 1995 wichtige neue Ansätze
- •diese sind von neueren Techniken übernommen und erweitert worden: Java Servlets, …
- **fastCGI ist aber für moderne Frameworks wie Ruby on Rails wieder sehr wichtig!**

Tomcat Clusterbetrieb





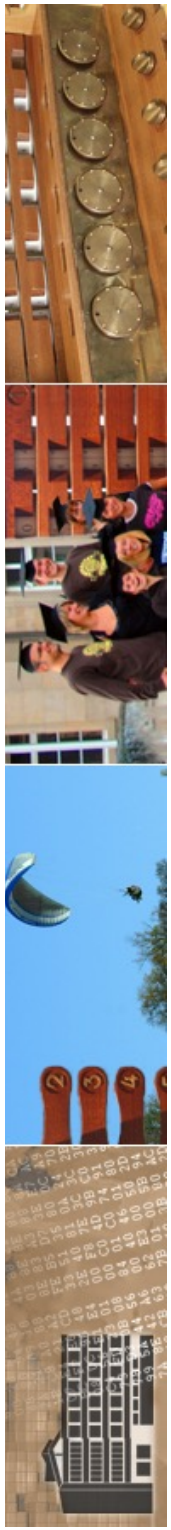


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

ServerSideIncludes (SSI)

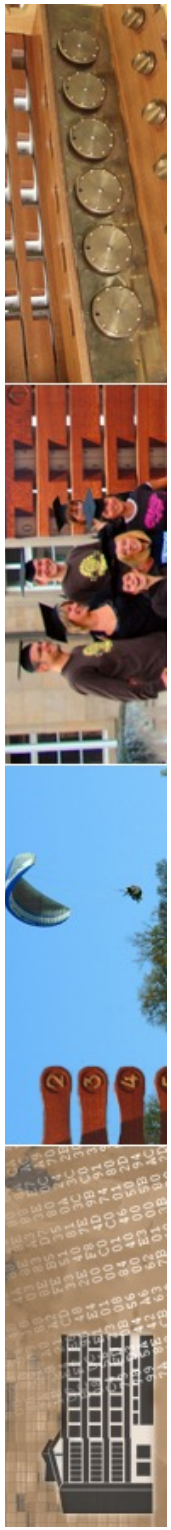
- ServerSideIncludes (SSI) – wofür?
  - einfach(st)e Möglichkeit zur Erstellung dynamischer Seiten
  - vollständig serverbasiert
  - von den Möglichkeiten beschränkt, aber für vieles ausreichend
  - wird durch Standard-Apachemodul **include** ermöglicht
  - wird insb. von Apache2 intensiv selbst genutzt



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

SSI: Voraussetzungen

- der Webserver Apache muss mit dem Modul **mod_include** betrieben werden
- bekannte Möglichkeiten:
  - statische Integration in Apache-Kernel
  - dynamische Integration über DSO



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Auszug aus httpd.conf

httpd.conf - Editor

Datei Bearbeiten Format Ansicht ?

```
# are static bound into the standard Apache binary distribution for Windows.
#
# Note: The order in which modules are loaded is important.  Don't change
# the order below without expert advice.
#
# [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE, UPDATE THIS TOO!]
ClearModuleList
#AddModule mod_vhost_alias.c
AddModule mod_env.c
AddModule mod_log_config.c
#AddModule mod_mime_magic.c
AddModule mod_mime.c
AddModule mod_negotiation.c
AddModule mod_status.c
AddModule mod_info.c
AddModule mod_include.c
AddModule mod_autoindex.c
AddModule mod_dir.c
```

Apache Module mod_include

Description:	Server-parsed html documents (Server Side Includes)
Status:	Base
Module Identifier:	include_module
Source File:	mod_include.c

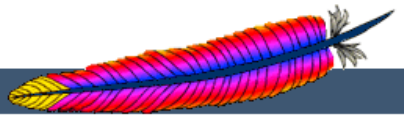
Summary

This module provides a filter which will process files before they are sent to the client. The processing is controlled by specially formatted SGML comments, referred to as *elements*. These elements allow conditional text, the inclusion of other files or programs, as well as the setting and printing of environment variables.

Enabling Server-Side Includes

Server Side Includes are implemented by the `INCLUDES` filter. If documents containing server-side include directives are given the extension .shtml, the following directives will make Apache parse them and assign the resulting document the mime type of `text/html`:

```
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
```



Apache Module mod_include

Description:	Server-parsed html documents (Server Side Includes)
Status:	Base
Module Identifier:	include_module
Source File:	mod_include.c

Summary

This module provides a filter which will process files before they are sent to the client. The processing is controlled by specially formatted SGML comments, referred to as *elements*. These elements allow conditional text, the inclusion of other files or programs, as well as the setting and printing of environment variables.

Enabling Server-Side Includes

Server Side Includes are implemented by the `INCLUDES` filter. If documents containing server-side include directives are given the extension .shtml, the following directives will make Apache parse them and assign the resulting document the mime type of `text/html`:

```
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
```

The following directive must be given for the directories containing the shtml files (typically in a `<Directory>` section, but this directive is also valid in `.htaccess` files if `AllowOverride Options` is set):

```
Options +Includes
```

For backwards compatibility, the `server-parsed` handler also activates the INCLUDES filter. As well, Apache will activate the INCLUDES filter for any document with mime type `text/x-server-parsed-html` or `text/x-server-parsed-html3` (and the resulting output will have the mime type `text/html`).

For more information, see our Tutorial on Server Side Includes.

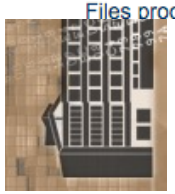
PATH_INFO with Server Side Includes

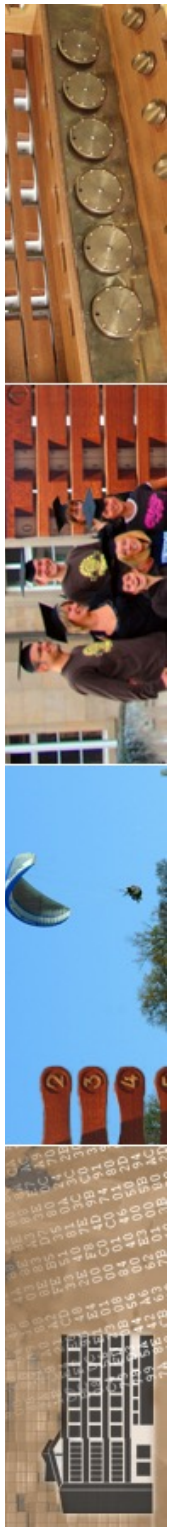
Files processed for server-side includes no longer accept requests with `PATH_INFO` (trailing pathname information) by default. You can use the `AcceptPathInfo`

Topics
- Enabling Server-Side Includes
- PATH_INFO with Server Side Includes
- Available Elements
- Include Variables
- Variable Substitution
- Flow Control Elements
- Legacy expression syntax

Directives
SSIEndTag
SSIErrorMsg
SSIETag
SSILastModified
SSILegacyExprParser
SSIStartTag
SSITimeFormat
SSIUndefinedEcho
XBitHack

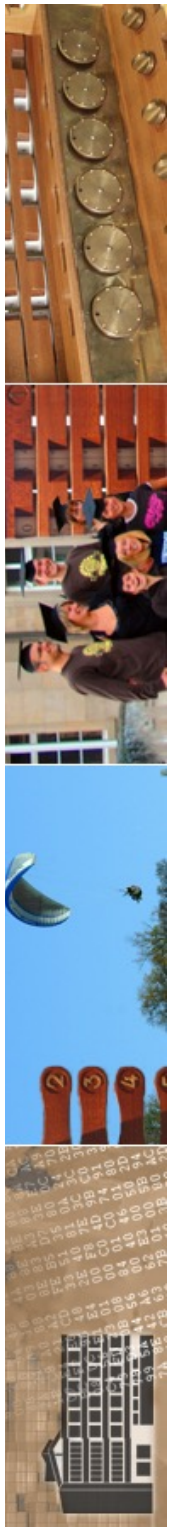
See also
- Options
- AcceptPathInfo
- Filters
- SSI Tutorial
- Comments





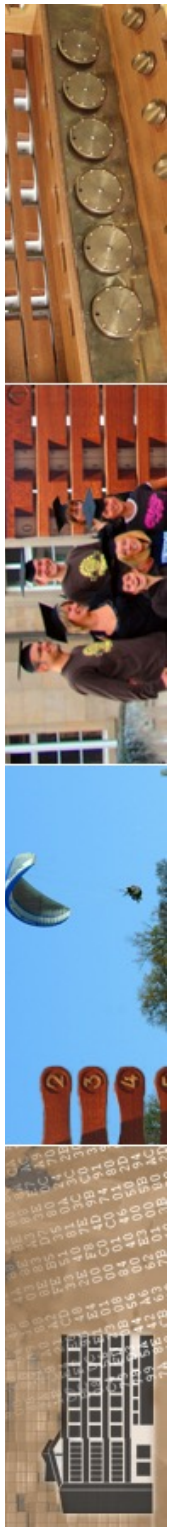
was sind SSI

- HTML-Anweisungen
- werden serverseitig bei Auslieferung der Seite ausgewertet
- dabei wird kein CGI-Prozeß etc. gestartet
- eine schnellere Alternative zu CGI
  - „SSI is a great way to add small pieces of information, such as the current time. But if a majority of your page is being generated at the time that it is served, you need to look for some other solution.“



SSI: Beispiele für Einsatz

- Beispiele:
  - Last-Modified des Dokuments
    - auch mit JavaScript möglich – aber mit SSI *clientunabhängig* (Format) und unbedenklich für Client
  - Einbinden einer Datei (etwa footer.html)
    - auch mit PHP möglich, aber größerer Aufwand und nicht mehr Datei-Endung ".html"



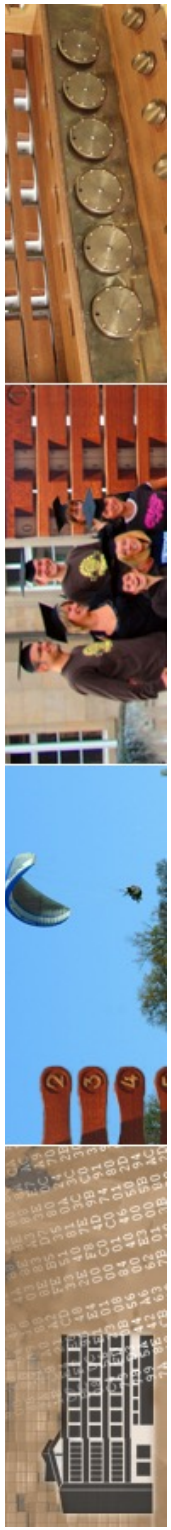
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

SSI: Voraussetzungen I

- in Apache-Konfiguration http.conf muß für ein Verzeichnis des Webservers und alle Unterverzeichnisse SSI erlaubt werden
  - Erlauben von SSI im Verzeichnis:
    - Direktive
      `Options +Includes`
    - Konkretes Beispiel

```
<Directory "d:/www/myssi">
      Options +Includes
</Directory>
```

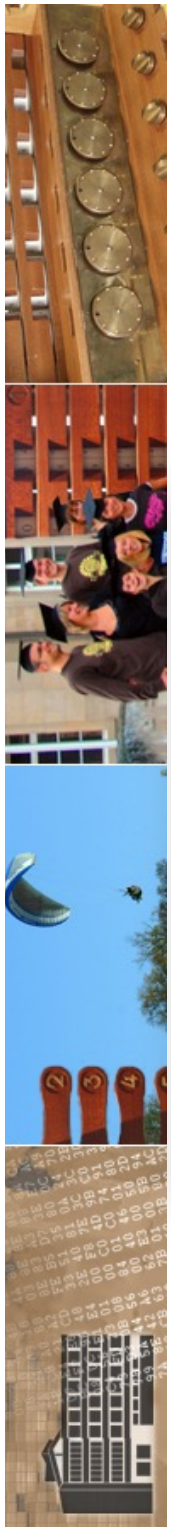
  - Alternative:
    - **`Options all`**

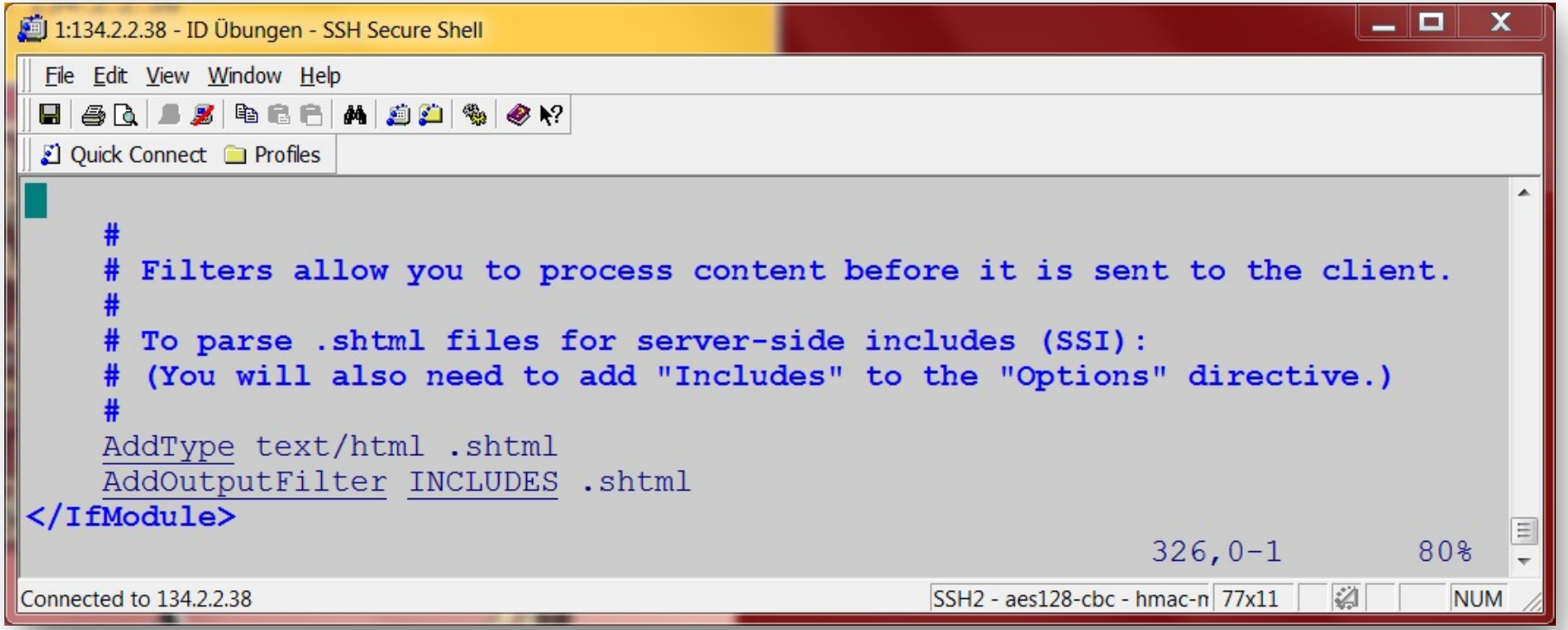


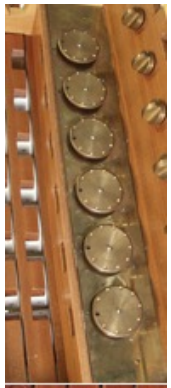
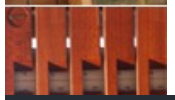
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

SSI: Voraussetzungen II

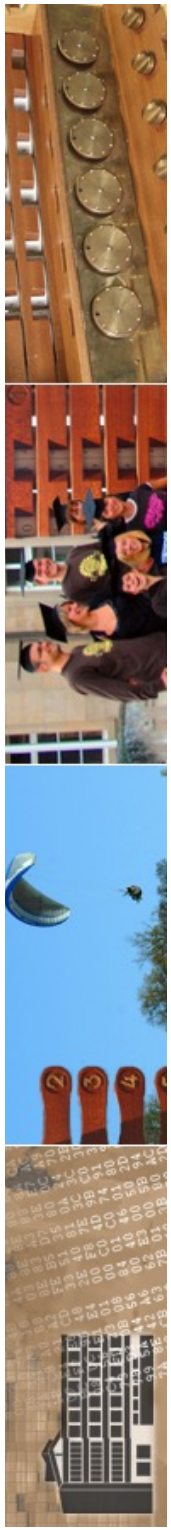
- Kennzeichnung einer HTML-Datei als SSI in http.conf:
  - Dateiendung shtml und Konfiguration
    - **`AddType text/html .shtml`**
      `AddHandler server-parsed .shtml`
  - oder *(nur* auf Unix): mit Direktive **`XBitHack on`** werden alle ausführbaren Dateien als SSI betrachtet (x-bit gesetzt)
    - **`chmod +x pagename.html`**
    - Voraussetung: eine weitere Direktive **`XBitHack`**




1:134.2.2.38 - ID Übungen - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
#
Filters allow you to process content before it is sent to the client.
#
To parse .shtml files for server-side includes (SSI):
(You will also need to add "Includes" to the "Options" directive.)
#
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
</IfModule>
326,0-1 80%
Connected to 134.2.2.38
SSH2 - aes128-cbc - hmac-n 77x11
NUM

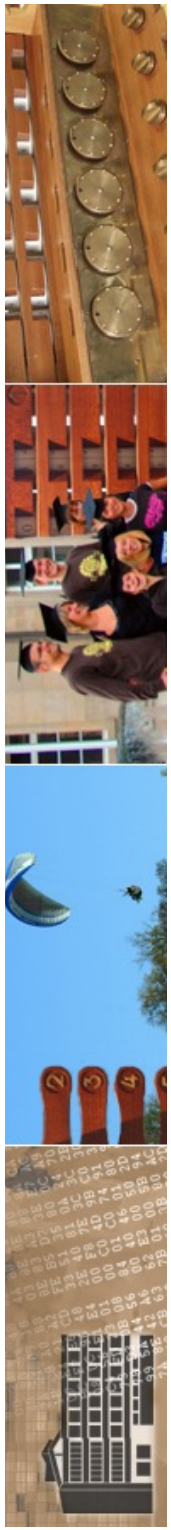
```
426       #
427       # Filters allow you to process content before it is sent to the client.
428       #
429       # To parse .shtml files for server-side includes (SSI):
430       # (You will also need to add "Includes" to the "Options" directive.)
431       #
432       #AddType text/html .shtml
433       #AddOutputFilter INCLUDES .shtml
434   </IfModule>
435
```

Syntax von SSI

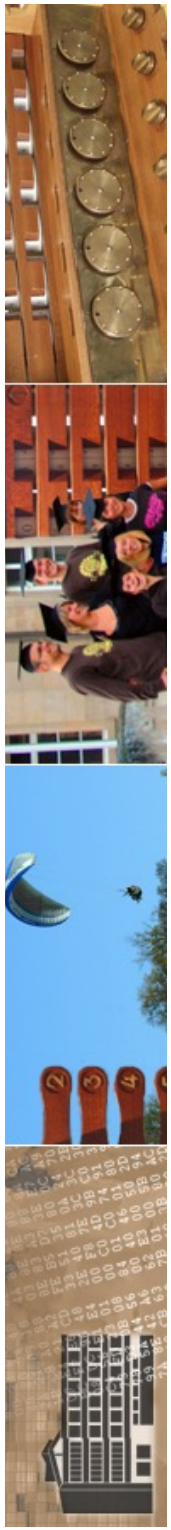
- Grundaufbau:
  - SSI-Befehle selbst sind **HTML-Kommentare!**
    - `<!--#element attribute="value" ... -->`
  - Beispiel: Ausgabe der Serverzeit:

```
<!--#echo var="DATE_LOCAL" -->
```



Beispiel

- wir erweitern unsere Startseite des Dokumentenservers
  - als erstes wollen wir die Serverzeit und Namen ausgeben
  - als zweites die Zeit der Aktualisierung der Seite ausgeben
    (über SSI, nicht wie häufig über JavaScript)
  - und letztlich noch eine footer-Datei einfügen



Ausgabe der Serverzeit und mehr

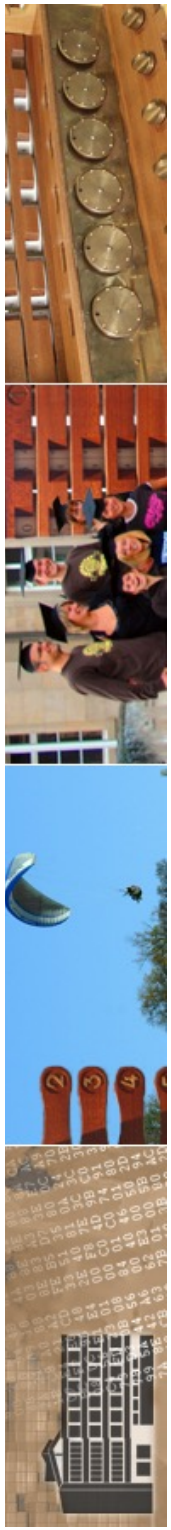
C:\data\www\webst1\ssi.shtml - Notepad++

Datei Bearbeiten Suchen Ansicht Kodierung Sprachen Einstellungen Makro Ausführen Erweiterungen Fenster ?

netzwerk_PM.txt | farben.txt | zusammenfassung.txt | AdwCleaner[S0].txt | ssi.shtml | ssi2.shtml | footer.html

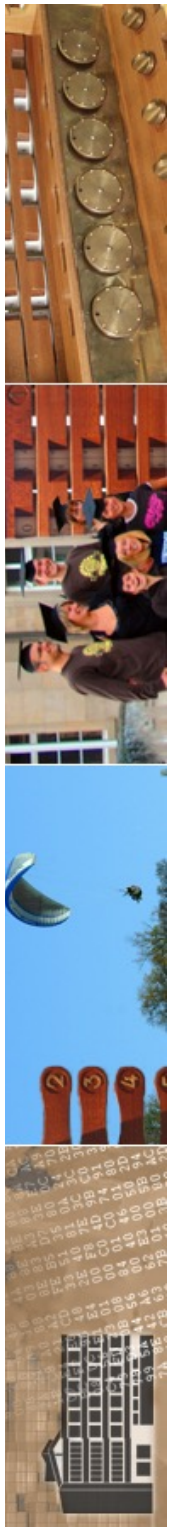
```
1  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.or
2  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de" lang="de">
3    <head>
4        <meta http-equiv="Content-type" content="text/html;charset=UTF-8" />
5        <meta name="author" content="Thomas Walter" />
6        <link rel="stylesheet" type="text/css" media="screen" href="css/mycss.css"
7        <title>Grundlagen der Web-Entwicklung - SSI</title>
8    </head>
9    <body bgcolor="orange">
10       <center>
11           <h2>Grundlagen der Web-Entwicklung<br />ServerSideIncludes</h2>
12       </center>
13         <hr />
14         Serverzeit mit SSI: <!--#echo var="DATE_LOCAL"-->
15         <BR />
16         Server-Name: <!--#echo var="SERVER_NAME"-->
17         <hr />
18         Last-Modified mit SSI: <!--#flastmod file="ssi.shtml" -->
19         <hr />
20         Import: <!--#include virtual="footer.html"-->
21         <hr />
22    </body>
23 </html>
```

Hyper Text Markup Language file | length : 900 | lines : 23 | Ln : 1 | Col : 1 | Sel : 0 | Dos\Windows | ANSI | INS



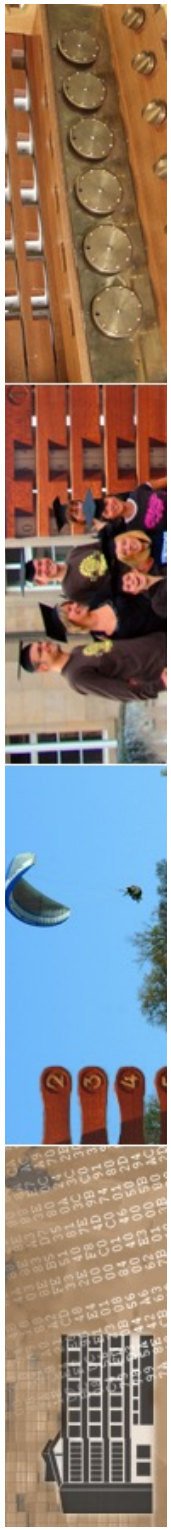
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Ergebnis



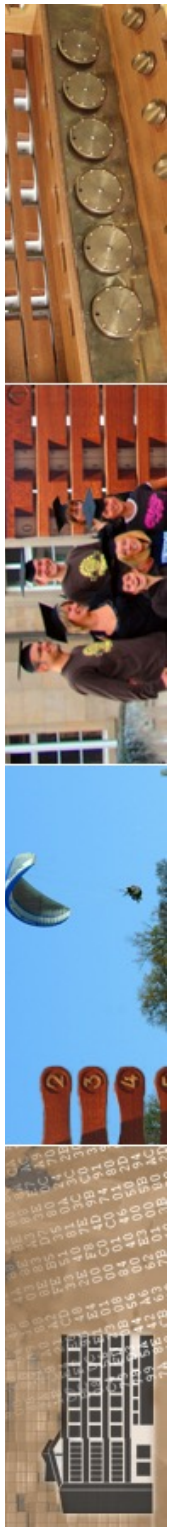
Zeitformate

- SSI stellt einfache Möglichkeit zur Zeitformatierung bereit:
  - `<!--#config timefmt="%d. %b %Y, %H.%mh" -->`
    `Today is <!--#echo var="DATE_LOCAL" -->`
  - Bedeutung:
    d: Tag, Y: Jahr b: Monat (Kurzform) H: Stunde, m: Minute



weitere Konfigurationen

- mit **`config`** kann neben dem Zeitformat noch festgelegt werden:
  - **`ermsg`** : die Fehlermeldung, die der Client erhält
  - **`sizefmt`** : bei Angabe einer Dateigröße das Format der Anzeige (Kb oder Mb)



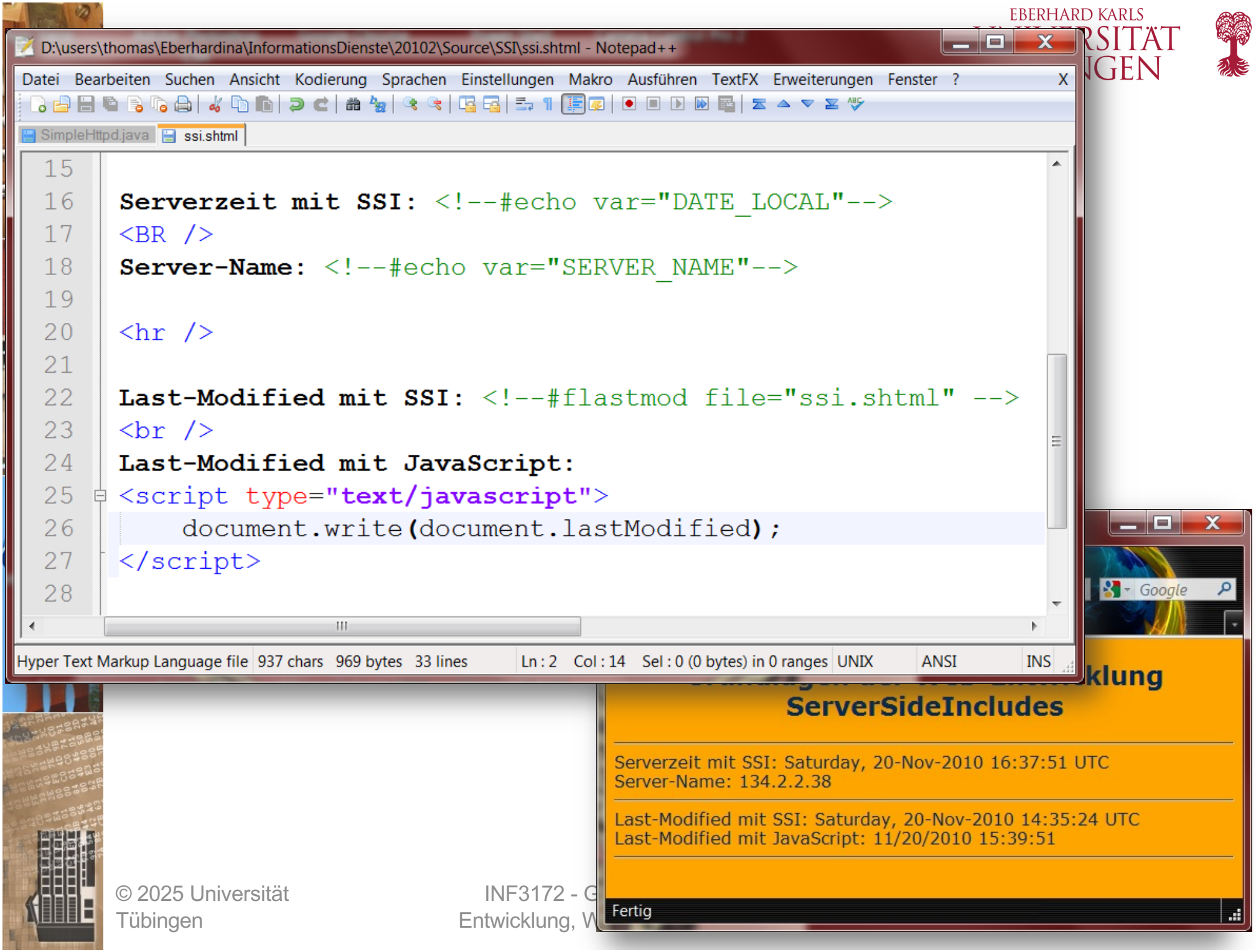
last modified

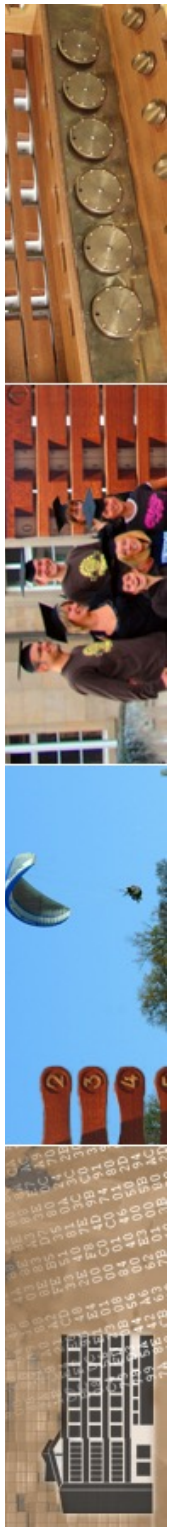
- Ersetzen des JavaScripts für “LastModified” durch SSI:

```
<!--#flastmod file="index.html" -->
```

...damit LastModified auch *ohne* JavaScript!

  - die Zeitformatierung gilt auch hier

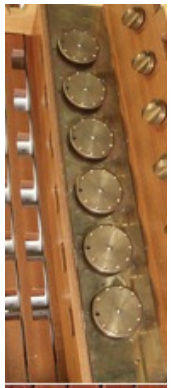




Einfügen einer footer-Datei

- alle Seiten sollen einheutlichen Abschluss bekommen
- es wäre sehr ungeschickt, diesen in jeder einzelnen HTML-Datei zu codieren
- -> Einfügen einer HTML-Datei

```
<!--#include virtual="/footer.html" -->
```



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

D:\users\thomas\Eberhardina\InformationsDienste\20102\Source\SSI\footer.html - Notepad++

Datei Bearbeiten Suchen Ansicht Kodierung Sprachen Einstellungen Makro Ausführen TextFX
Erweiterungen Fenster ?

SimpleHttpd.java | ssi.shtml | footer.html

```
1  <font size="-2">
2      Grundlagen der Web-Entwicklung
3      <BR />
4      (c) Universit&auml;t T&uuml;bingen, 2010
5  </font>
```

H 102 chars 110 bytes 5 lines | Ln : 1 Col : 1 Sel : 0 (0 bytes) in 0 ranges | Dos\Windo ANSI | INS

Grundlagen der Web-Entwicklung - SSI

Datei Bearbeiten Ansicht Chronik Lese

http://134

Grundlagen der Web-Entwic...

Grundlagen der ServerSi

Serverzeit mit SSI: Saturday, 20-
Server-Name: 134.2.2.38

Last-Modified mit SSI: Saturday,

Grundlagen der Web-Entwicklung
(c) Universität Tübingen, 2010

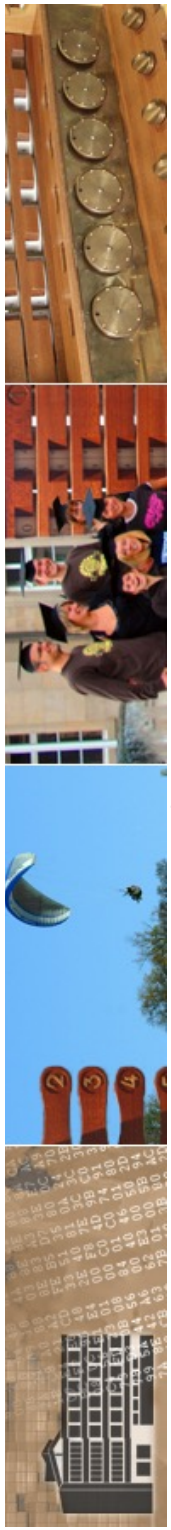
Fertig

Datei Bearbeiten Suchen Ansicht Kodierung Sprachen Einstellungen Makro Ausführen TextFX Erweiterungen Fenster ? X

SimpleHttpd.java | ssi.shtml | footer.html

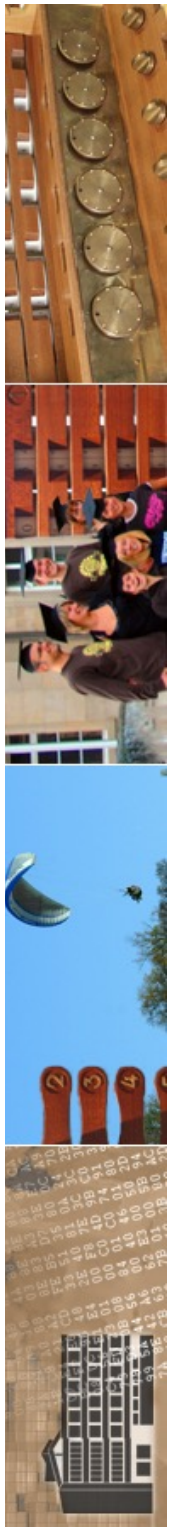
```
15
16          Serverzeit mit SSI: <!--#echo var="DATE_LOCAL"-->
17          <BR />
18          Server-Name: <!--#echo var="SERVER_NAME"-->
19
20          <hr />
21
22          Last-Modified mit SSI: <!--#flastmod file="ssi.shtm
23          <hr />
24
25          <!--#include virtual="footer.html"-->
26      </body>
27  </html>
```

Hyper Text Markup Language file | 849 chars 876 bytes 28 lines | Ln : 14 Col : 1 Sel : 0 (0 bytes) in 0 ranges | UNIX | ANSI | INS



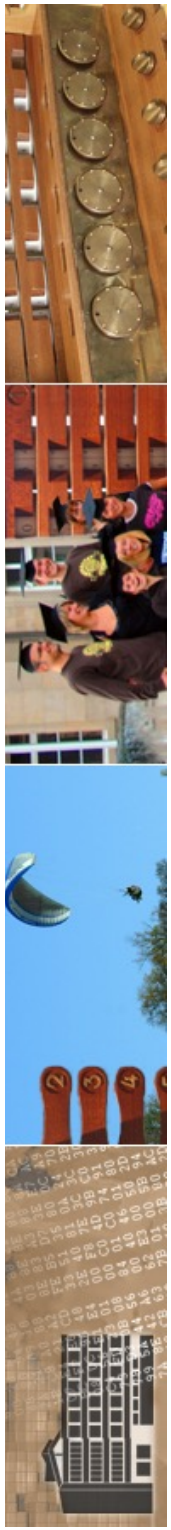
Kombination

- in der »includeten«-Datei kann selbst auch wieder SSI-Code enthalten sein, der dann entprechend ausgeführt wird
- »geschachtelte« SSIs sind möglich!
  - Beispiel: Serverzeit in footer.html



Ausführen eines Programms

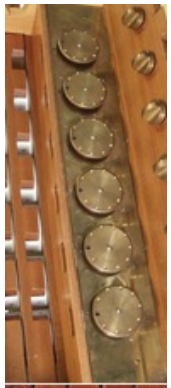
- SSI kann direkt ein ausführbares Programm auf dem Server starten
  - Anweisung
  - `<!--#exec cmd="ls -alp"  -->`
- damit etwa auch Python-Scripte mit Shebang-Zeile einbindbar

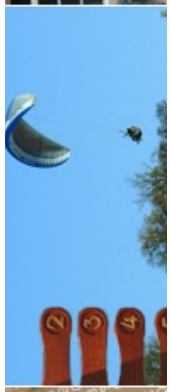


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Ausführen eines cgi-Scriptes

- mittels **`#exec cmd=`**… wird ein auf der Shell ausführbares Programm ausgeführt
- Alternative:
  - **`#exec cgi=`**…
    führt dierkt ein vollwertiges CGI aus



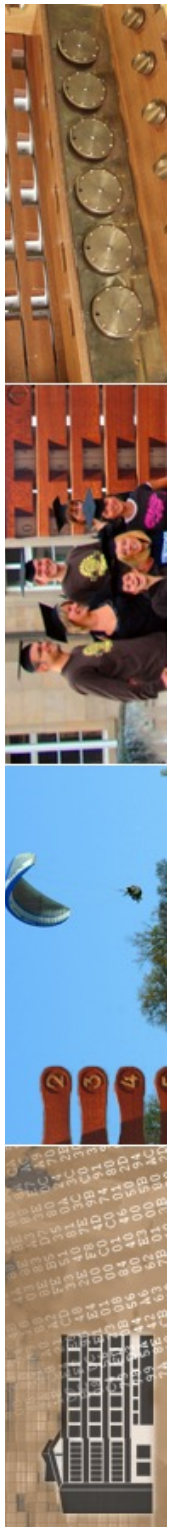



Was gibt es noch?

- noch ein paar »Kleinigkeiten«:
  - einfache (typenlose) Variablen

```
<!--#set var="cost"  value="€100"  -->
```

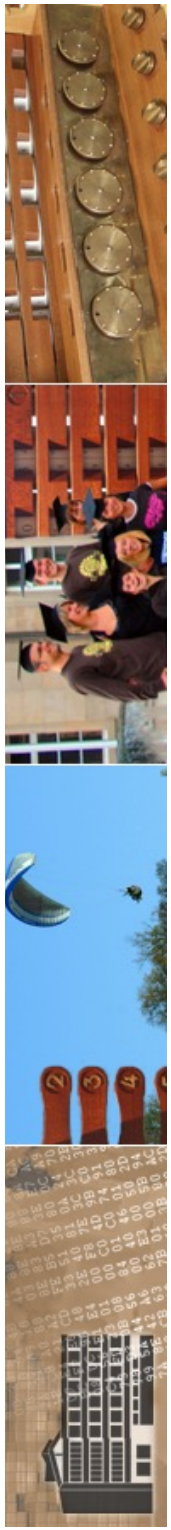
  - Vergleichsoperationen (if - else)
  - Abfrage Browser-Typ
  - Ausgabe der Größe einer Datei: fsize



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

SSI-Standardvariablen

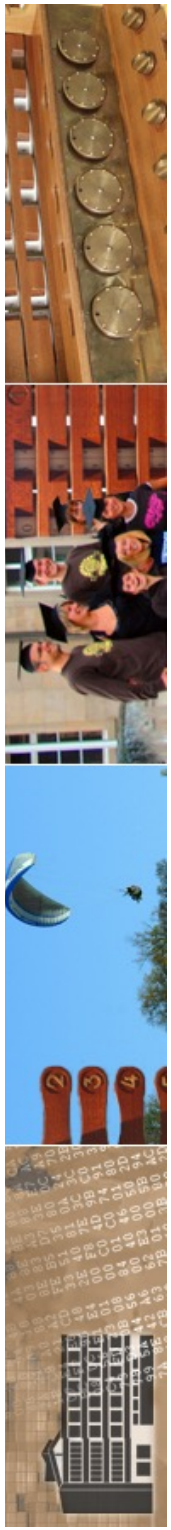
- `DATE_GMT`
- `DATE_LOCAL`
- `DOCUMENT_NAME`
- `DOCUMENT_URI`
- `LAST_MODIFIED`
- `HTTP_USER_AGENT`
- `HTTP_REFERER`
- `SERVER_NAME`
- `SERVER_SOFTWARE`
- `REMOTE_ADDR`



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

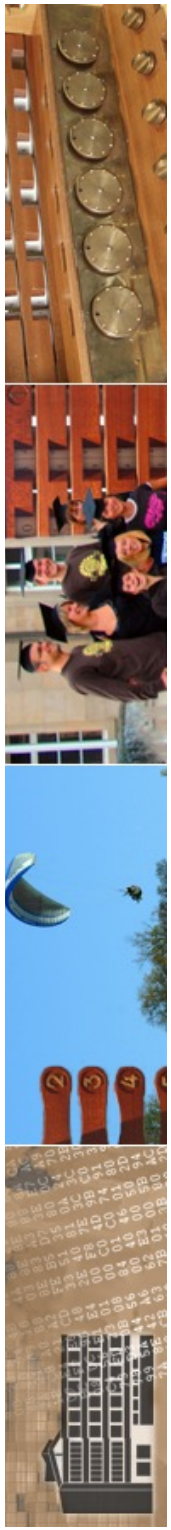
SSI Commands

- `include`
- `config`
- `echo`
- `fsize`
- `flastmod`
- `exec`
- `set`



Beispielanwendung

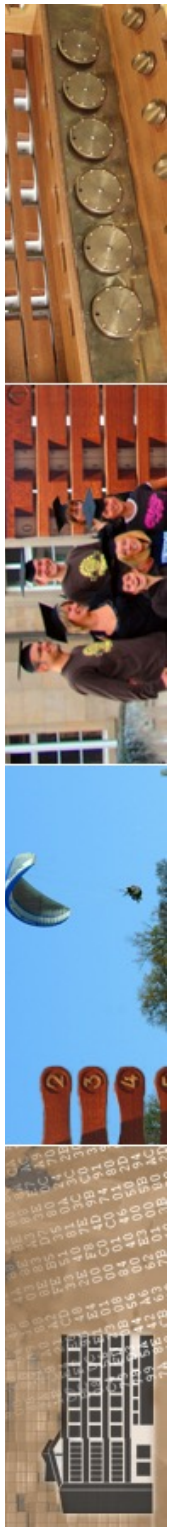
- Site mit beim Neuladen austauschenden Bildern (Zufallsazswahl)
  - über SSI, welches PERL-Script zur Zufallsberechnung der Bilder enthält
- modularer Seitenaufbau
  - Auslagerung des Menüs in Datei, die von SSI importiert wird



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

eingeschränkte Ausführung

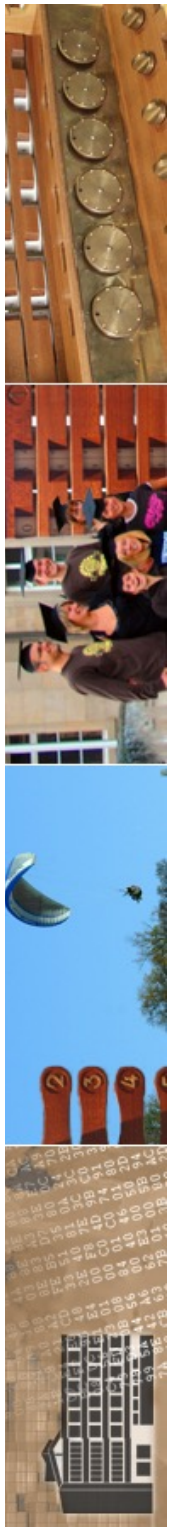
- besondere Apache-Direktive
  - **`IncludesNOEXEC`**
- diese erlaubt SSI *ohne* **`#include`** und **`#exec`**



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Zusammenfassung SSI

– „SSI is certainly not a replacement for CGI, or other technologies used for generating dynamic web pages. But it is a great way to add small amounts of dynamic content to pages, without doing a lot of extra work.“



…und nun…

- haben wir zwei spezielle Apache-Module genauer kennen gelernt: fastCGI und Includes

- als nächstes: Content-Management-Systeme, insbesondere TYPO3