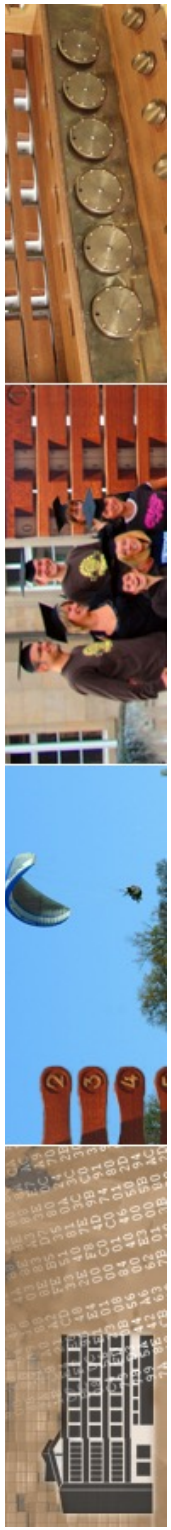


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATHEMATISCH-
NATURWISSENSCHAFTLICHE FAKULTÄT
FACHBEREICH INFORMATIK
INFORMATIONSDIENSTE

Grundlagen der Web-Entwicklung

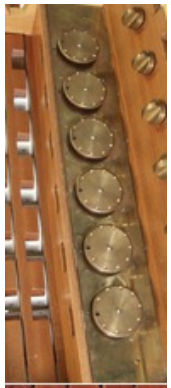
INF3172

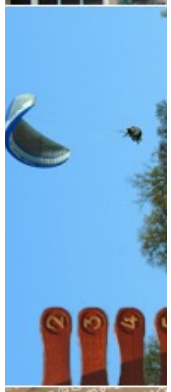
Template-Engines am Beispiel Smarty

Thomas Walter

12.12.2024

Version 1.0

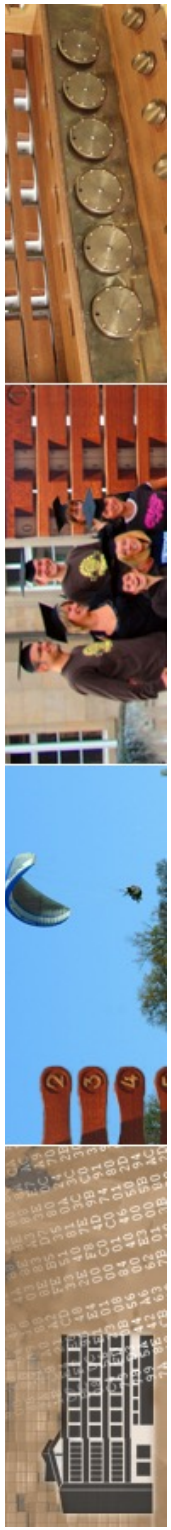





das Weihnachtsrätsel

- am **1. Advent 01.12.2024**
  ab 10.00h:
  das Weihnachtsrätsel!

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

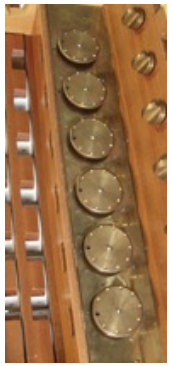


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

das Weihnachtsrätsel

- Einstieg über

  http://134.2.6.167:2412/xmas/

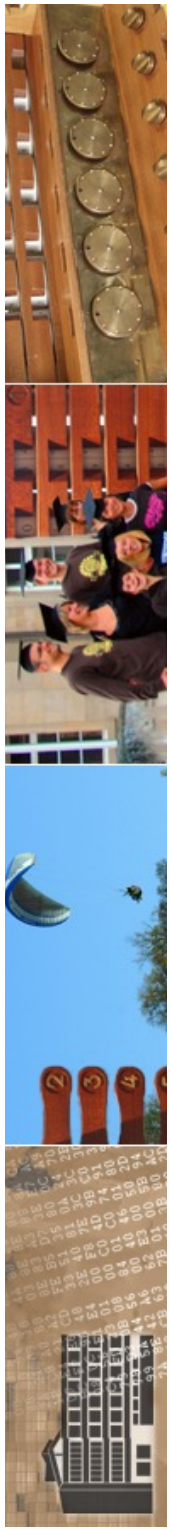


Netzbsicherung über die Feiertage

Studie: Unternehmen an Feiertagen und Wochenenden anfälliger für Cyberangriffe

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

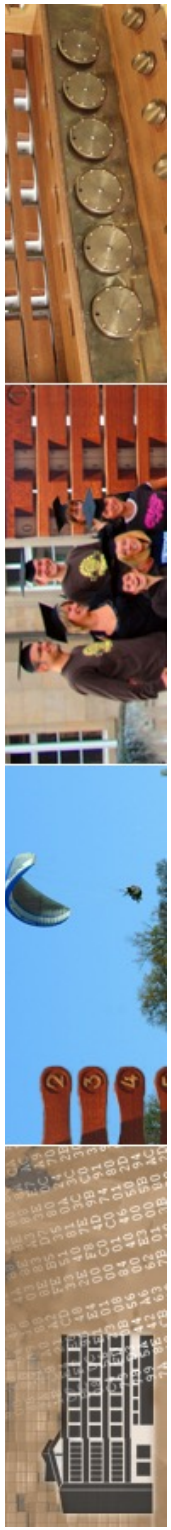
Unternehmen sind an Feiertagen und Wochenenden besonders gefährdet für Cyberangriffe, da das Sicherheitspersonal in dieser Zeit oft reduziert ist. Das bestätigt jetzt auch eine neue Studie zu Ransomware-Angriffen von Semperis, einem Anbieter im Bereich identitätsbasierter Cyber-Resilienz. Demnach wurden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



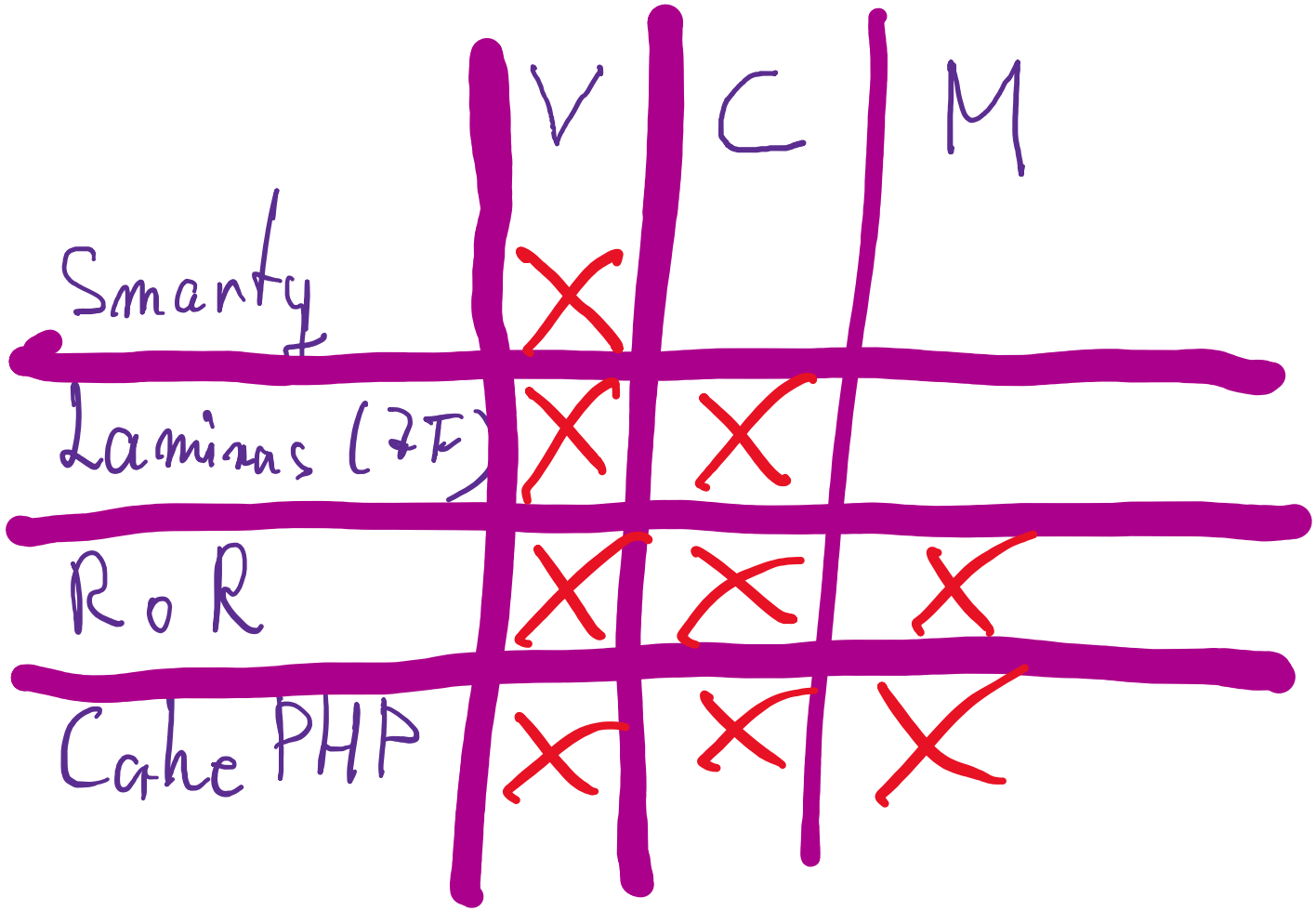
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

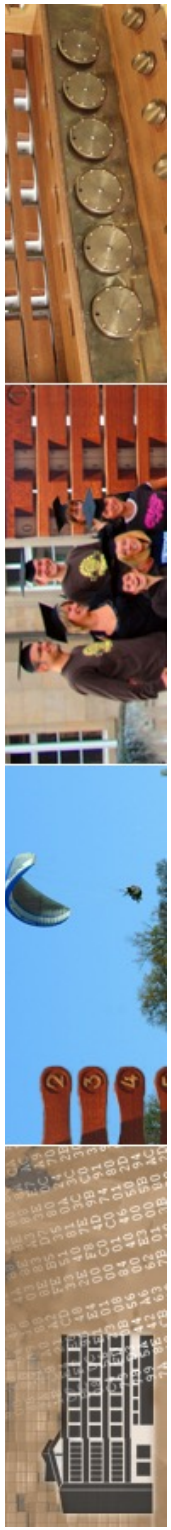
Frameworks

- Framework: *wiederverwertbares* Softwaresystem mit bereits implementierter, genereller Funktionalität
  - Spezialisierung führt zu konkreter Anwendung
  - Framework *setzt Architektur um*
  - Framework folgt Design Pattern
  - Beispiele
    - Node.js
    - ZF: Laminas (Zend Framework)
    - Ruby on Rails
    - JEE
    - Angular
    - dojo



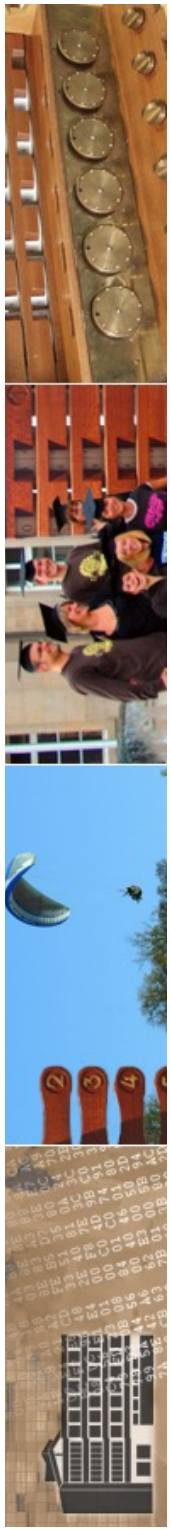
Frameworks (serverseitig) und MVC





Template Engine

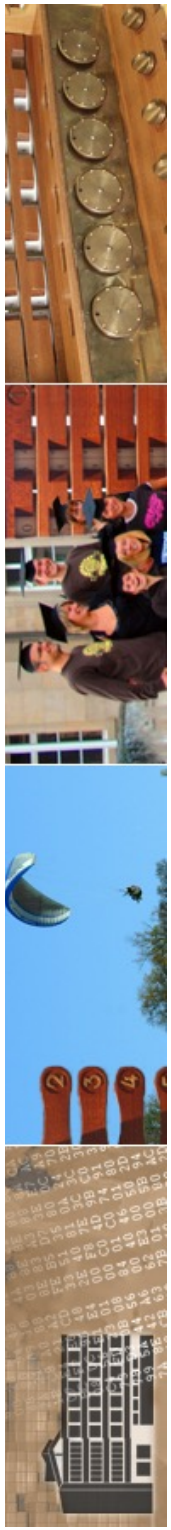
- Software, welche in Vorlagen (den Templates) Platzhalter mit konkretem Inhalt ausfüllt
- in Templates soll *keine* Business-Logik enthalten sein!
- in MVC-Paradigma: Abspaltung der View durch Template-Engine



Bewertung

- durch Template-Engine *saubere Trennung* der View vom Programm-Code
- Designer unabhängig vom Programmierer, kann sich besser einbringen

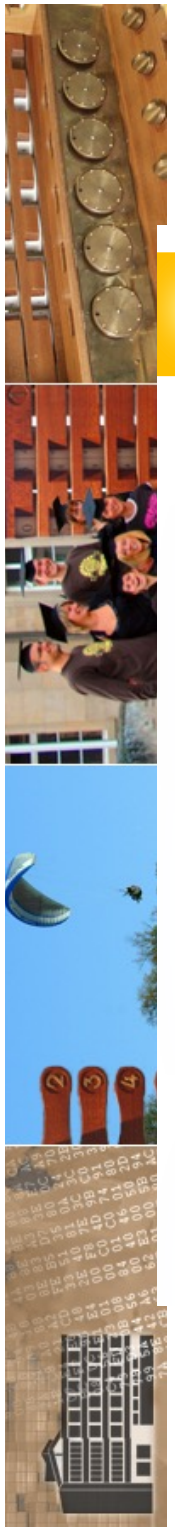
- zusätzlicher Overhead
- zusätzliche Strukturen
- Performance



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Beispiele

- Beispiele für Template Engines
  - Smarty
    - www.smarty.net
  - Fluid/FLOW3
  - alle weiteren serverseitigen Frameworks in dieser Veranstaltung haben eine Art von Template Engine: CakePHP, Ruby on Rails, Laminas-(ZEND)-Framework
  - zahlreiche weitere



smarty
TEMPLATE ENGINE

Smarty

Home
What is Smarty?
Why use it?
Use Cases and Work Flow
Syntax Comparison
Template Inheritance
Best Practices
Crash Course

Resources

Download
Quick Install

Smarty 5.4.2, 4.5.5 Released! Nov 21, 2024

5.4.2
4.5.5

Smarty 5.4.1 Released! Aug 29, 2024

5.4.1

v5.4.0, v4.5.4 released! Aug 14, 2024

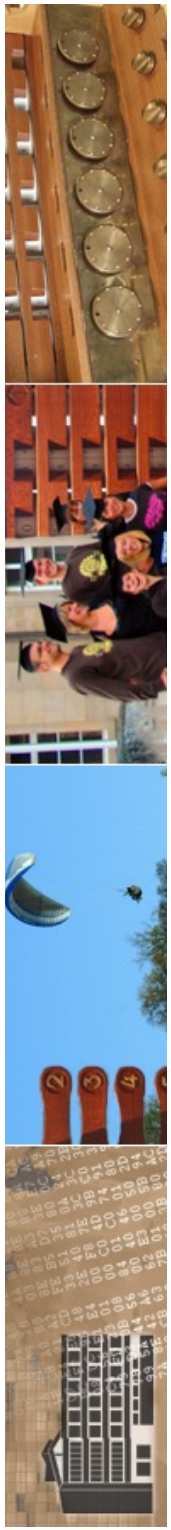
5.4.0
4.5.4

PARTNERS

Compare Swedish casinos online on casivo.se

Sponsors [info]

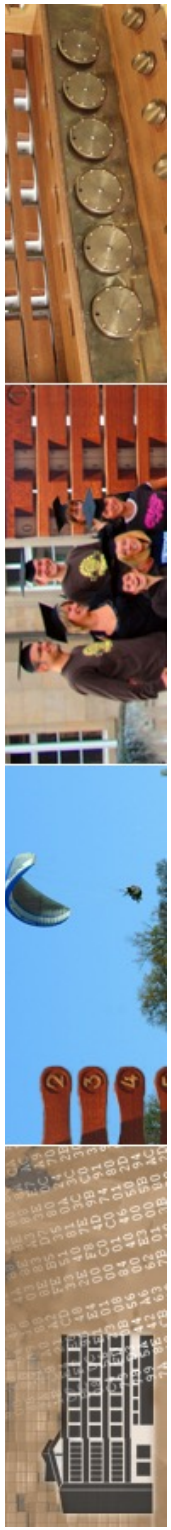
pelisivut.com
nettikasino.fi
nettikasinot.org
Buzzoid
Twicsy
sure.bet
Play online casino
casasapuestasdeportivas.es
rahapelit
casinoonlineespaña.es
kasinot
comparador de casinos
casinosonlineespaña.es
Bookmakers
BSV DevCon
Netticasinot
Intetics - Software Development Company
VPS Server



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Einleitung Smarty

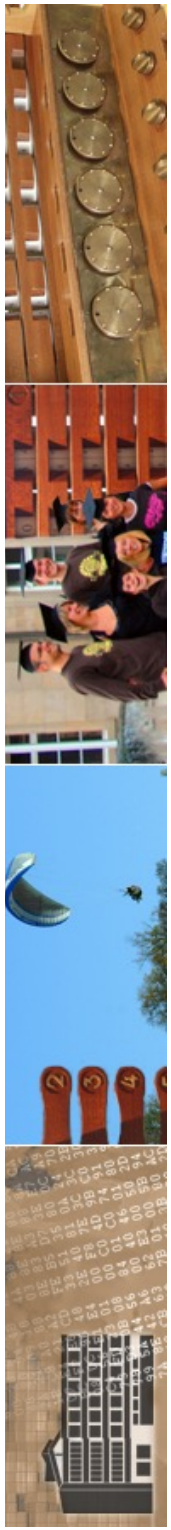
- Smarty ist eine „kompilierende Template-Engine“ für PHP
- Ziel: Trennung der in PHP geschriebenen Anwendungslogik von der Sicht/Formatierung
  - keine Applikationslogik im Template, keine Präsentationslogik in der PHP-Anwendung



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Separation of presentation from application code

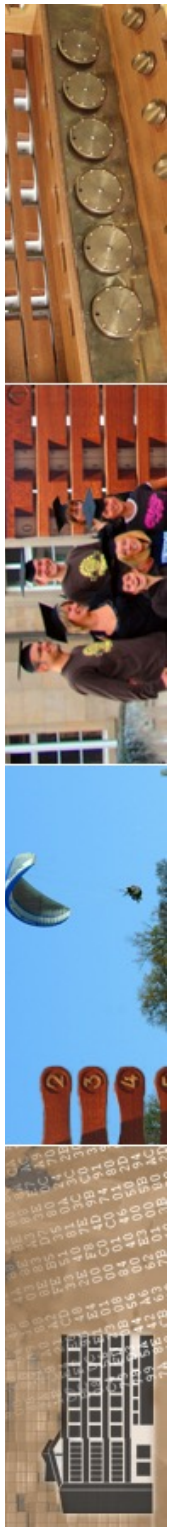
- This means templates can certainly contain logic under the condition that it is for presentation only. Things such as including other templates, alternating table row colors, upper-casing a variable, looping over an array of data and rendering it are examples of presentation logic.
- This does not mean however that Smarty forces a separation of business and presentation logic. Smarty has no knowledge of which is which, so placing business logic in the template is your own doing.
- Also, if you desire *no* logic in your templates you certainly can do so by boiling the content down to text and variables only.



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Vorteile von Smarty

- Smarty ist
  - sehr performant
  - nur einmaliges Compilieren der Templates (wenn unverändert)
  - if/elsif/else/endif -Konstrukte im Template
  - unterstützt Caching
  - erweiterbar durch PlugIn-Architektur



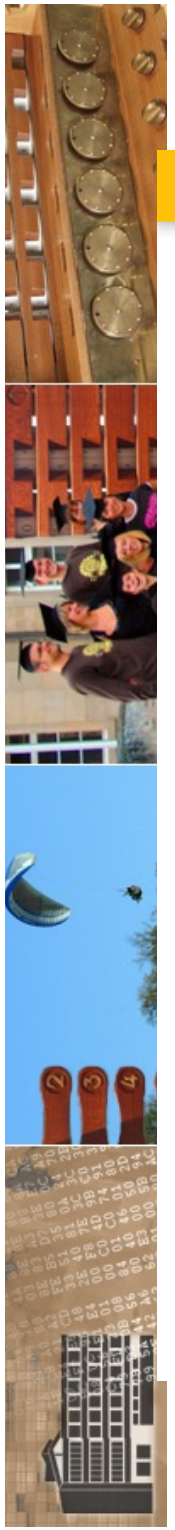
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Resourcen

- zentral:

  http://www.smarty.net/

- darüber hinaus zahlreiche Ressourcen im Web
  - auch gute Dokumentation
    (Monte Ohrt, Andrei Zmievski)



Smarty Documentation

Smarty is a template engine for PHP, facilitating the separation of presentation (HTML/CSS) from application logic.

It allows you to write **templates**, using **variables**, **modifiers**, **functions** and **comments**, like this:

```
<h1>{$title|escape}</h1>

<p>
    The number of pixels is: {math equation="x * y" x=$height y=$width}.
</p>
```

When this template is rendered, with the value "Hello world" for the variable $title, 640 for $width, and 480 for $height, the result is:

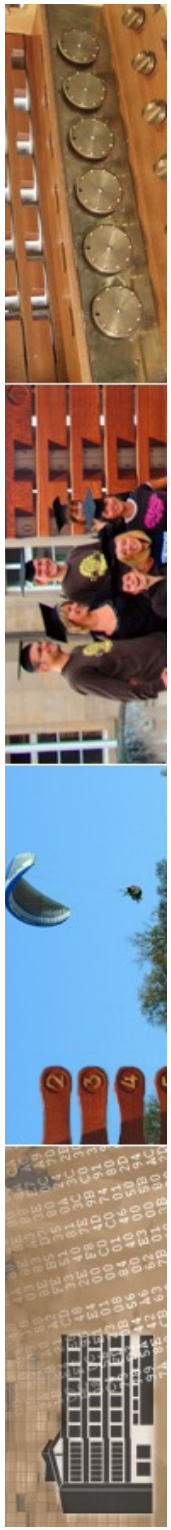
```
<h1>Hello world</h1>

<p>
    The number of pixels is: 307200.
</p>
```

Getting Started

- Getting Started
- Philosophy - or "Why do I need a template engine?"
- Features - or "Why do I want Smarty?"

Table of contents

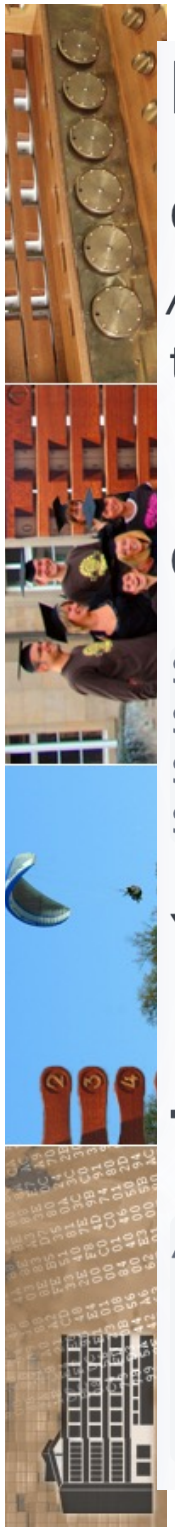
Getting Started
Help
Source code



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Installation

- die Grundinstallation ist einfach:
  - Herunterladen von Smarty (aktuell sind 5.4.2. und 4.5.5)
  - Entpacken
  - Kopieren des Smarty-Verzeichnisses an geeignete Stelle, etwa **`PHPDIR/smarty`**
  - Anpassen der Direktive **`include_path`** in php.ini
  - aktuelles/passendes PHP vorausgesetzt



Install Library Files

Copy the Smarty library files to your system. In our example, we place them in */usr/local/lib/php/Smarty/*. If you are using FTP/sFTP for server access, unzip the Smarty files locally and upload them to the proper directory.

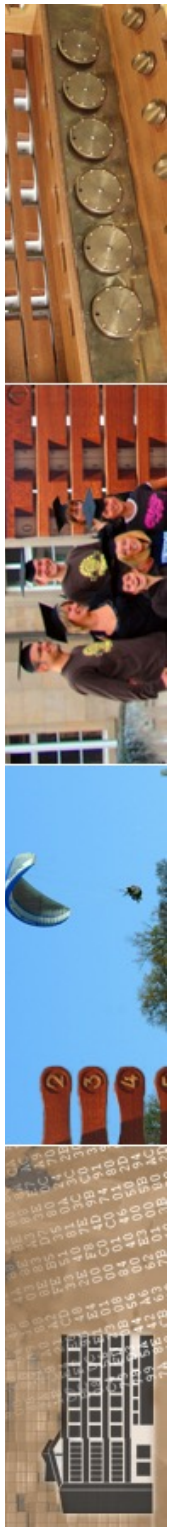
command line

```
$> cd YOUR_DOWNLOAD_DIR
$> gtar -zxvf Smarty-3.0.tar.gz
$> mkdir /usr/local/lib/php/Smarty
$> cp -r Smarty-3.0/libs/* /usr/local/lib/php/Smarty
```

You should now have the following file structure:

file structure

```
/usr/local/lib/php/Smarty/
    debug.tpl
    plugins/
    Smarty.class.php
    sysplugins/
```



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Installation

- oder:
  Composer (wieder)

A Dependency Manager for PHP

Installation

Smarty versions 3.1.11 or later can be installed with Composer.

To get the latest stable version of Smarty use:

```
composer require smarty/smarty
```

To get the latest, unreleased version, use:

```
composer require smarty/smarty:dev-master
```

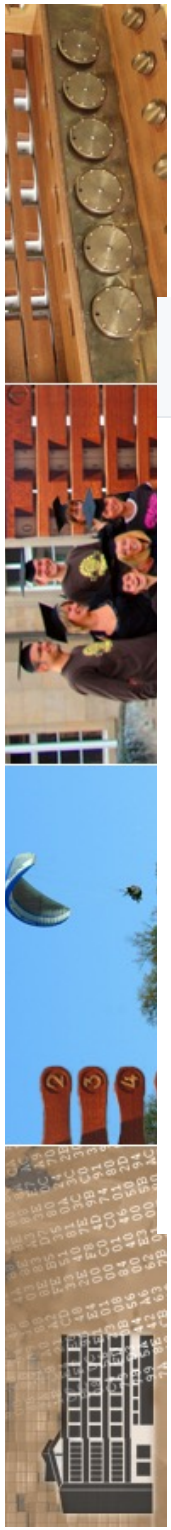
To get the previous stable version of Smarty, Smarty 3, use:

```
composer require smarty/smarty:^3
```

Here's how you create an instance of Smarty in your PHP scripts:

```
<?php

require 'vendor/autoload.php';
$smarty = new Smarty();
```

Now that the library files are in place, it's time to setup the Smarty directories for your application.



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

smarty-php / smarty Public

Notifications Fork

Code Issues 33 Pull requests 7 Discussions Actions Projects Wiki Security 5 Insights

master 24 branches 58 tags Go to file Code

wisskid Allow dereferencing of non-objects accross all supported PHP vers... a34ee98 last month 4,892 commits

.github/workflows	PHP8.2 compatibility (#775)	last month
demo	fix PHP 8.1 deprecation notices in demo/plugins/cacheresource.pdo....	last year
docs	PHP8.2 compatibility (#775)	last month
lexer	Dropped remaining references to removed PHP-support in Smarty 4 ...	3 months ago
libs	Allow dereferencing of non-objects accross all supported PHP versio...	last month
tests	Allow dereferencing of non-objects accross all supported PHP versio...	last month
utilities	PHP8.2 compatibility (#775)	last month
.gitattributes	Include docs en demo in the releases.	3 months ago
.gitignore	add local testrunners for all supported PHP versions using docker. (#...	5 months ago
CHANGELOG.md	Allow dereferencing of non-objects accross all supported PHP versio...	last month
LICENSE	Clarify correct LGPL version.	2 years ago
README.md	PHP8.2 compatibility (#775)	last month
SECURITY.md	Update SECURITY.md	4 months ago
composer.json	changed homepage links in composer.json	last year
docker-compose.yml	Simplify test running to support all phpunit cmdline options	last month

About

Smarty is a template engine for PHP, facilitating the separation of presentation (HTML/CSS) from application logic.

php smarty

Readme
View license
Security policy
2.1k stars
179 watching
670 forks

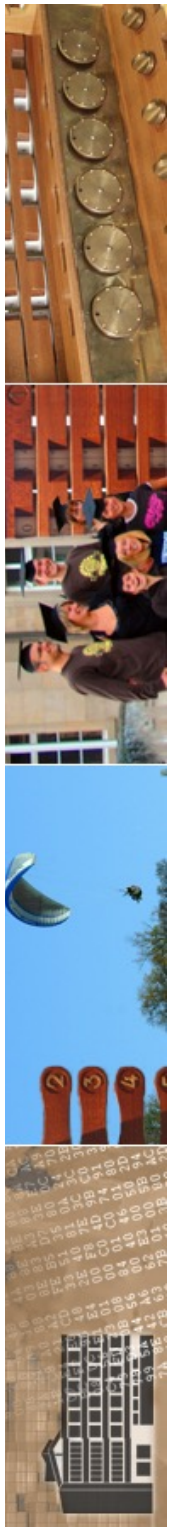
Releases 38

v4.3.0 Latest
last month

+ 37 releases

Packages

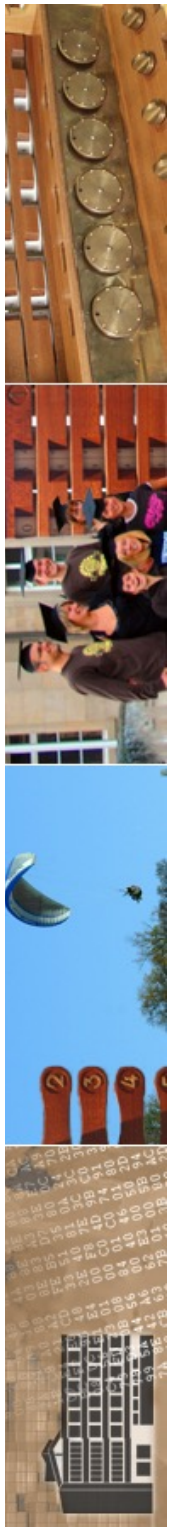
No packages published



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

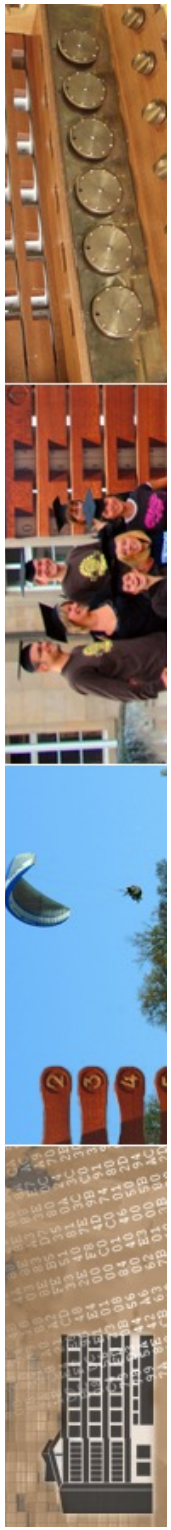
Verzeichnisstruktur

- Smarty sollte unbedingt mit einer vernünftigen Verteilung auf Verzeichnisse betrieben werden
  - spezielle Verzeichnisse für compilierte Templates, Templates, Konfigurationen, ...
  - nur wenig unterhalb von „htdocs“



erste Schritte

- Beispielanwendung mit Smarty:
  - Smarty-PHP setzt eine Variable „message“
  - Template fügt den Wert der Variablen ein und verwendet unser CSS



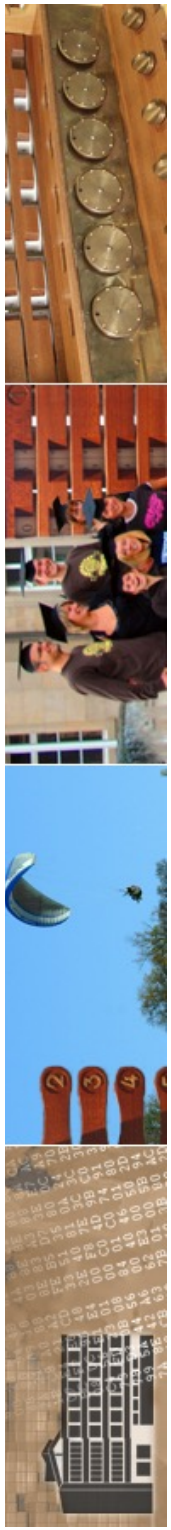
PHP - hellosmarty.php - Eclipse SDK

File Edit Navigate Search Project PHP/Apache Run Latex Window Help

EuroCalc2.java | Euro4.java | hellosmarty.php | hellosmarty.tpl | 58

```
1  <?php
2  /*
3   * Created on 15.12.2005
4   *
5   * Fortgeschrittene Programmierung fuer das Internet
6   *
7   * Beispiel fuer Smarty
8   */
9  ?>
10
11
12 <?php
13 require 'Smarty.class.php';
14 $smarty = new Smarty;
15
16 $smarty->template_dir="D:\\www\\webst2\\smarty\\templates";
17
18 $smarty->assign("message","Hello Webst2<BR>here's Smarty'");
19
20 $smarty->display('hellosmarty.tpl');
21 ?>
```

Writable | Smart Insert | 1 : 1



PHP - hellosmarty.tpl - Eclipse SDK

File Edit Navigate Search Project PHP/Apache Run Latex Window Help

EuroCalc2.java | hellosmarty.php | hellosmarty.tpl | smarty.lib.php | 59

```
1  {* webst2: Smarty-Template *}
2
3  <HTML>
4      <HEAD>
5          <TITLE>webst2: HelloWorld mit Smarty</TITLE>
6          <link rel="stylesheet" type="text/css" href="/css/webst.css">
7      </HEAD>
8      <BODY>
9          <HR>
10         <CENTER><H2>
11             {$message}
12         </H2></CENTER>
13         <HR>
14     </BODY>
15 </HTML>
```

Writable | Insert | 1 : 1

webst2: HelloWorld mit Smarty - Mozilla Firefox

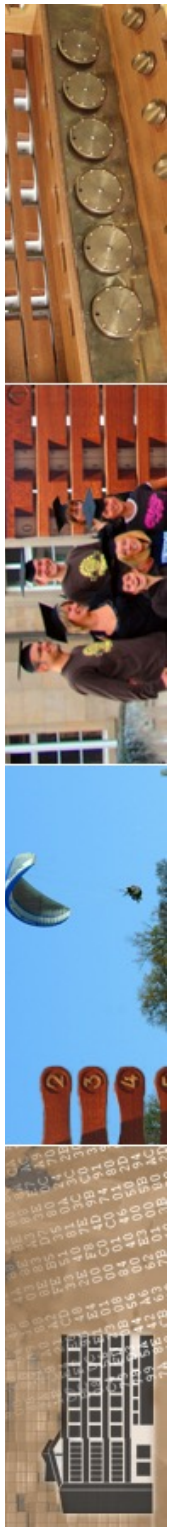
Datei Bearbeiten Ansicht Gehe Lesezeichen Extras Hilfe

http://127.0.0.1/webst2/smarty/hellosmarty.php

FH Kaiserslautern: web... | Hochschul-Informations... | Smarty : Template Engine | webst2: HelloWorld ... | 216 Farben (Netscape)

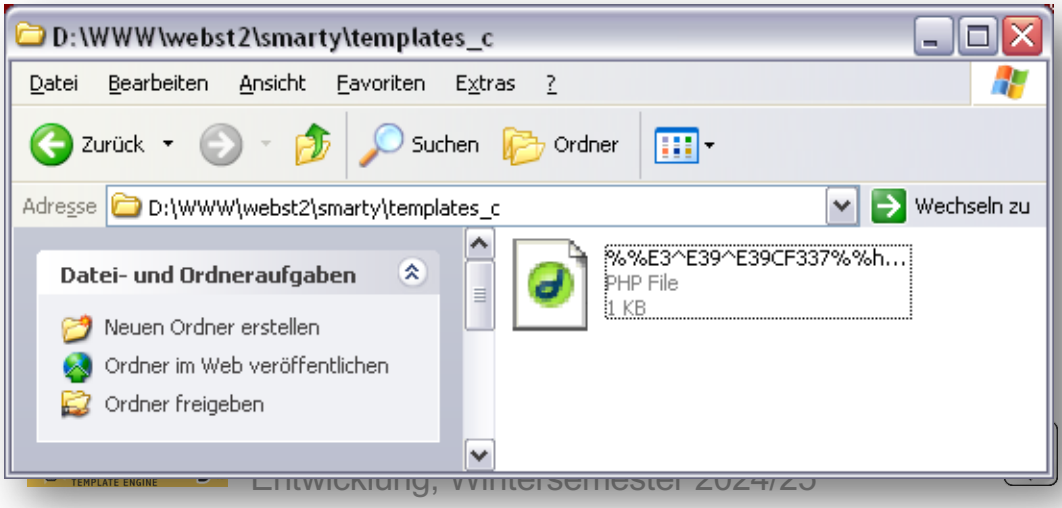
Hello Webst2
here's Smarty!

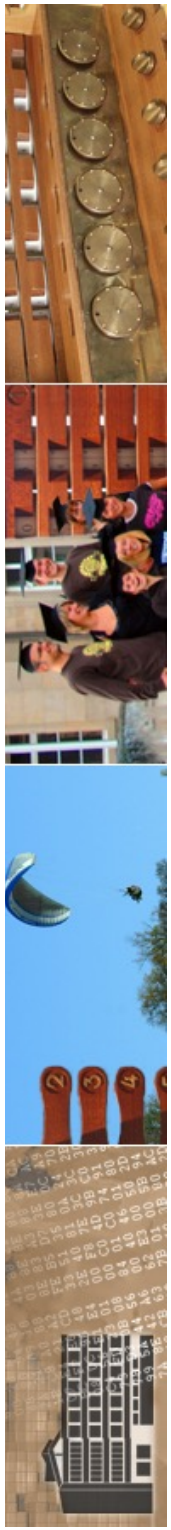
Fertig

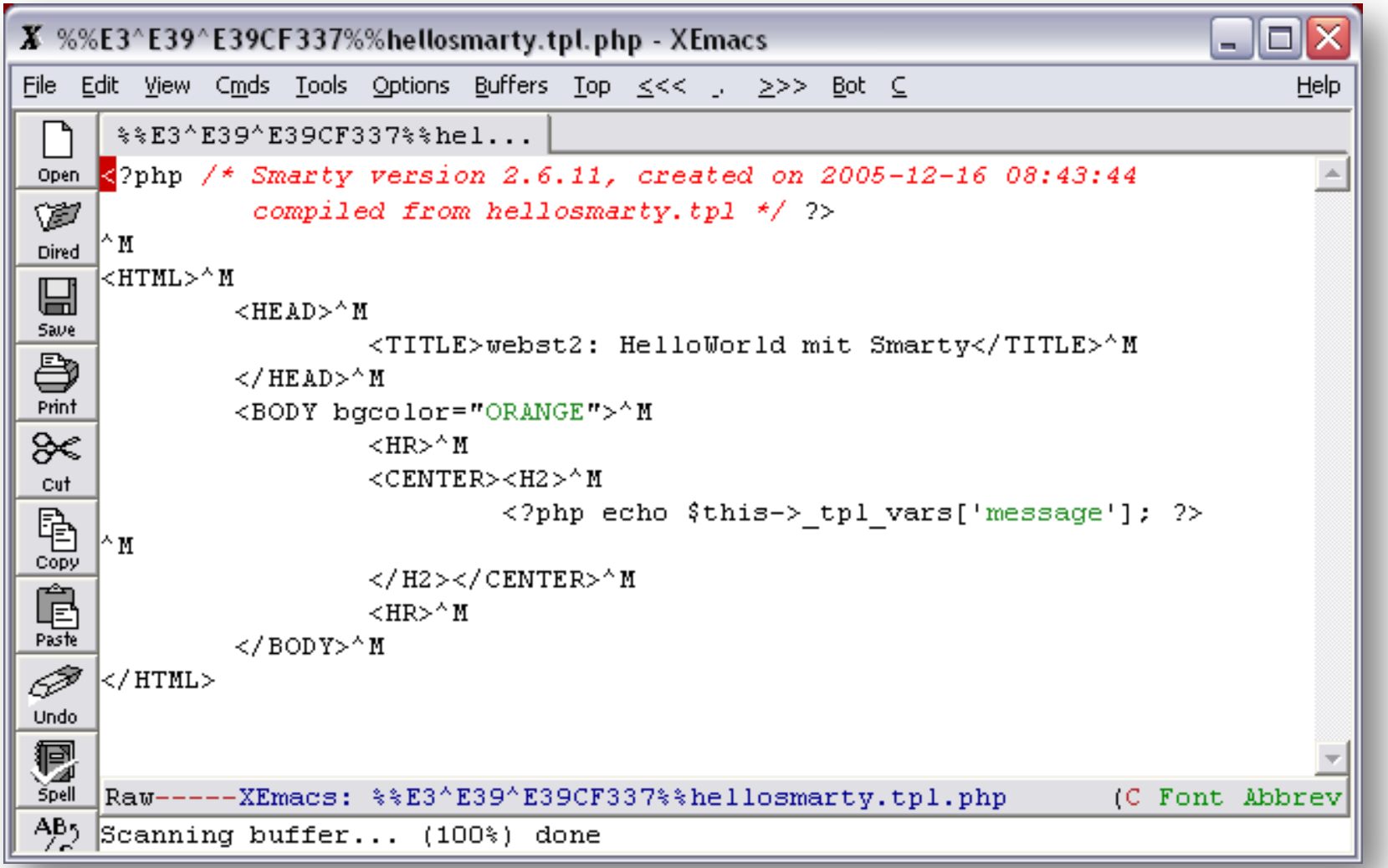


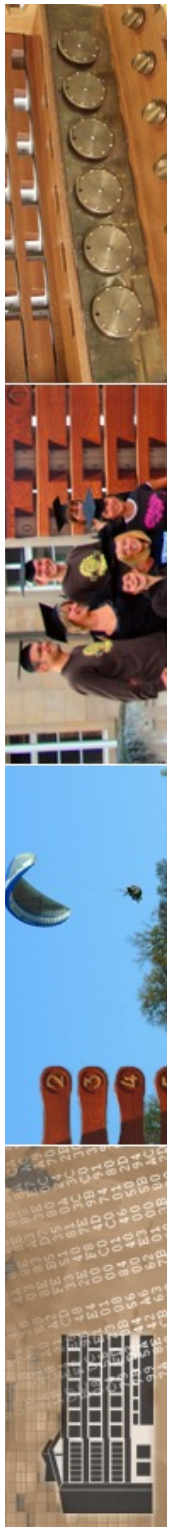
beim Ausführen...

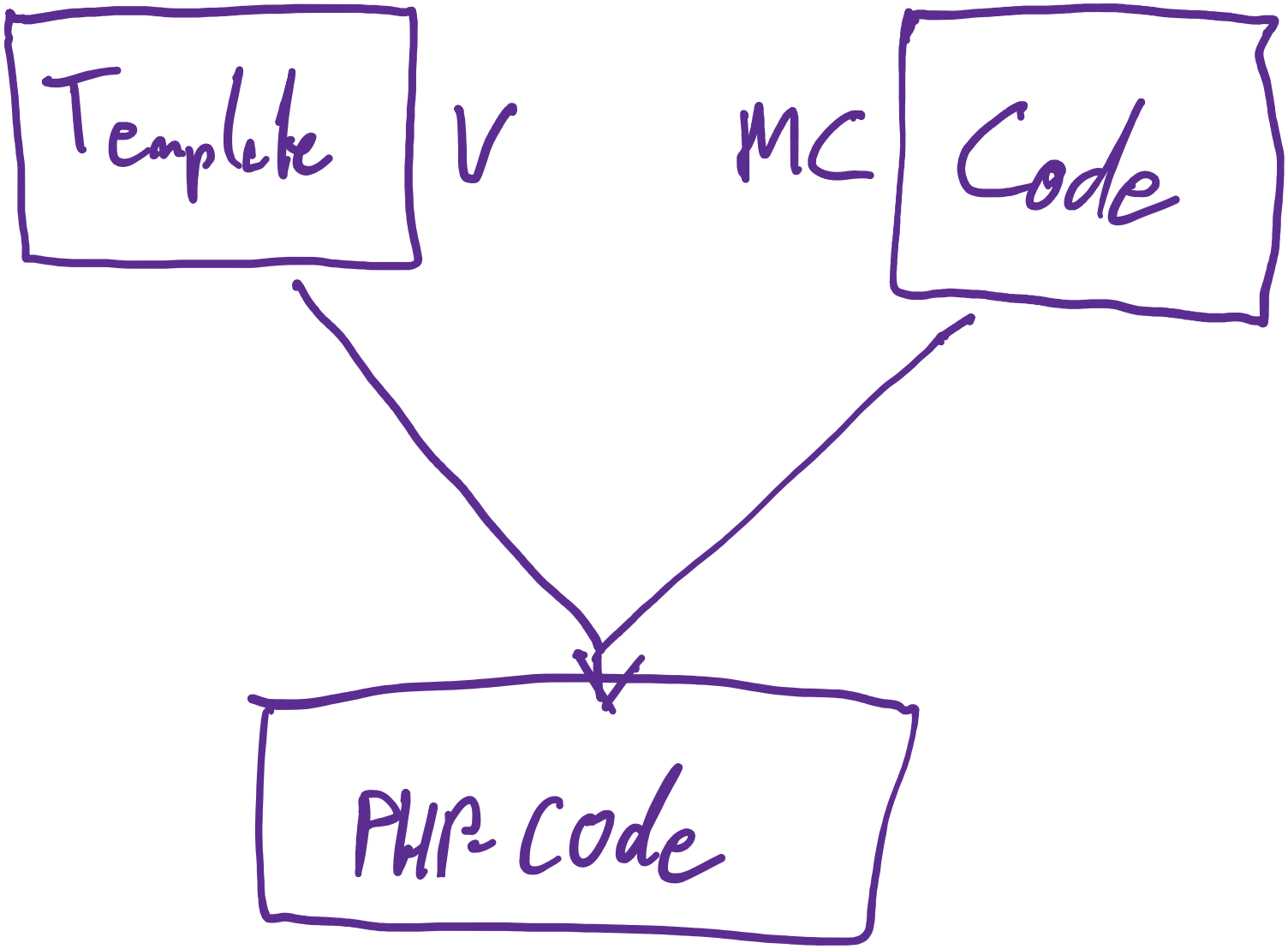
- ...wird das Template „compiliert“ zu reinem PHP
- großer Performance-Vorteil, da keine Neucompilation, solange sich das Template nicht geändert hat

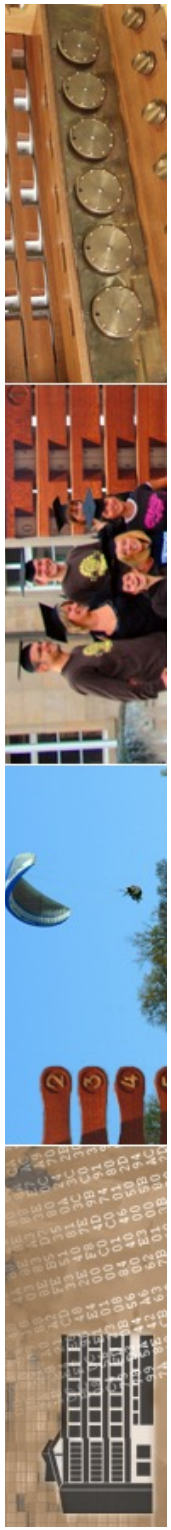





%%E3^E39^E39CF337%%hellosmarty.tpl.php - XEmacs
File Edit View Cmds Tools Options Buffers Top <<< . >>> Bot C Help
Open
Dired
Save
Print
Cut
Copy
Paste
Undo
Spell
%%E3^E39^E39CF337%%hel...
<?php /* Smarty version 2.6.11, created on 2005-12-16 08:43:44
compiled from hellosmarty.tpl */ ?>
^M
<HTML>^M
<HEAD>^M
<TITLE>webst2: HelloWorld mit Smarty</TITLE>^M
</HEAD>^M
<BODY bgcolor="ORANGE">^M
<HR>^M
<CENTER><H2>^M
<?php echo $this->_tpl_vars['message']; ?>
^M
</H2></CENTER>^M
<HR>^M
</BODY>^M
</HTML>
Raw-----XEmacs: %%E3^E39^E39CF337%%hellosmarty.tpl.php (C Font Abbrev
Scanning buffer... (100%) done



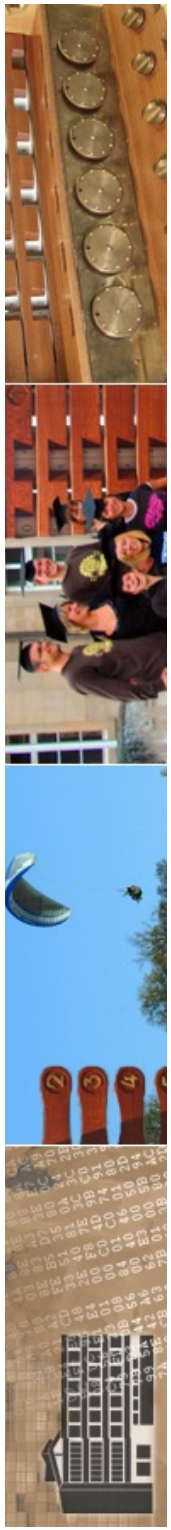




EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Konfiguration über Datei

- zahlreiche Smarty-Parameter können sinnvollerweise über eine zentrale include-Datei verwaltet werden



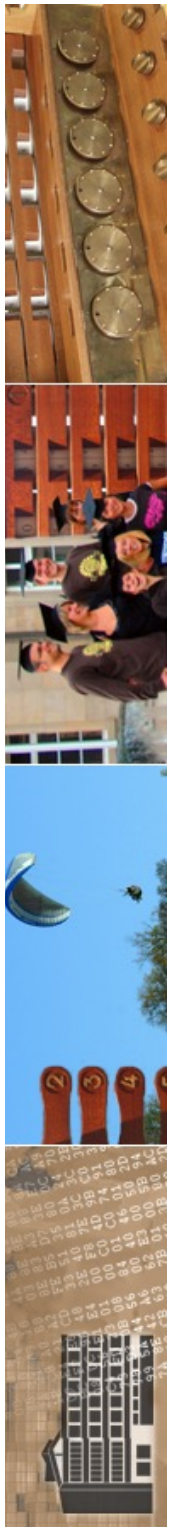
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Smarty programmieren

- Grundlegendes:

  Smarty-Code innerhalb von Templates beginnt standardmäßig mit den Tags

  ```
  {     <smarty-code>     }
  ```

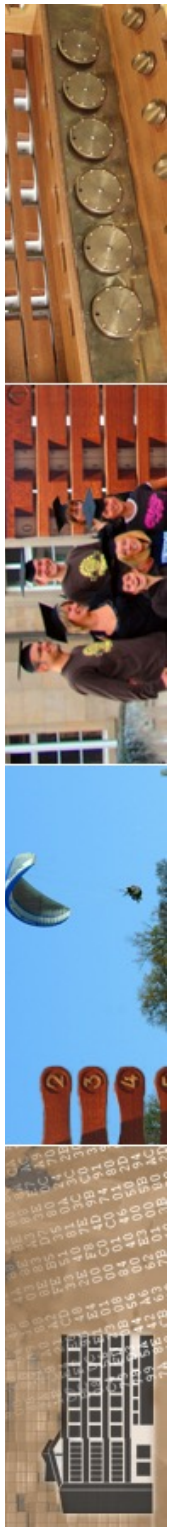


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Kommentare

- Kommentare in Smarty-Templates:

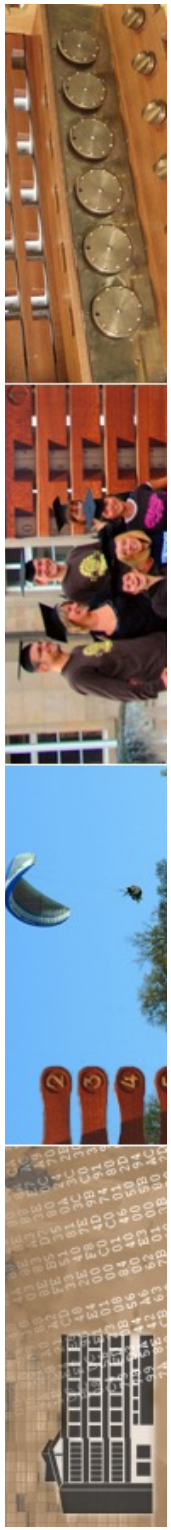
```
{
   * Kommetar im Smarty-Template  *
}
```



Variablen

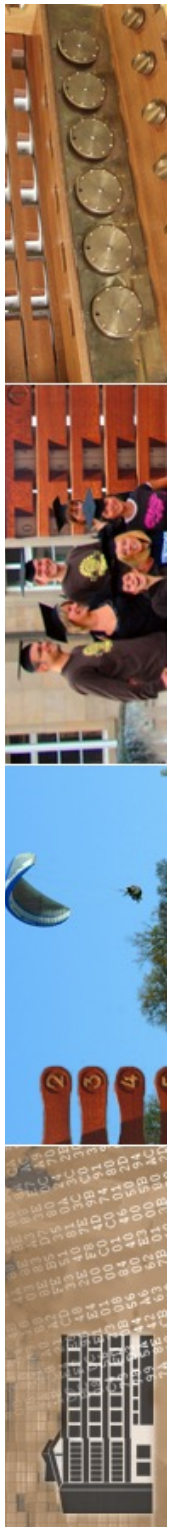
- Variablenbezeichner beginnen mit **`$`**
- Arrays werden mit **`[i]`** indiziert
- assoziative Arrays werden $assArray.key indiziert
- Objektattribut: **`$referenz->attribut`**
- Objektmethode: **`$referenz->methode()`**

- Systemvariable: **`#PATH#`**



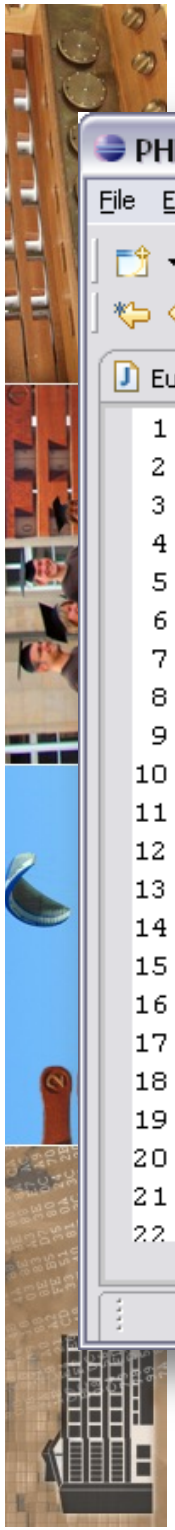
Übergabe von Variablenwerten aus PHP-Script (I)

- mittels
  - `assign(variablenname, wert);`
- wird in PHP eine Variable für das Template bereitgestellt
- diese Variable hat im Template den Bezeichner **`$variablenname`** und den Wert **`wert`**



Übergabe von Variablenwerten aus PHP-Script (II)

- analog für PHP-Arrays:
  - `assign(arrayname, array(0,1,2,"drei"));`
- und für assoziative Arrays in PHP:
  - `assign(hashname,array(key1 => value1, key2 => value2);`



PHP - variablen.php - Eclipse SDK

File Edit Navigate Search Project PHP/Apache Run Latex Window Help

EuroCalc2.java | hellosmarty.php | hellosmarty.tpl | smarty.lib.php | variablen.php | variablen.tpl | 59

```
1  <?php
2  /*
3   * Created on 16.12.2005
4   *
5   * Fortgeschrittene Programmierung fuer das Internet
6   *
7   * Variablen in Smarty
8   */
9
10 require 'Smarty.class.php';
11
12 $smarty = new Smarty;
13 $smarty->template_dir="D:\\www\\webst2\\smarty\\templates";
14
15 $smarty->assign("lokaleVariable","drei");
16 $smarty->assign("anArray",array(0,1,2,"drei"));
17 $smarty->assign("aHash",array("eins" => 1, "zwei" => "zwei", "drei" => "zweiundvierzig"));
18
19 $smarty->display('variablen.tpl');
20
21 ?>
22
```

Writable | Smart Insert | 1 : 1

PHP - variablen.tpl - Eclipse SDK

File Edit Navigate Search Project PHP/Apache Run Latex Window Help

hellosmarty.php | hellosmarty.tpl | variablen.php | variablen.tpl

```
1  {* webst2: Smarty-Template zu Script "variablen.php" *}
2
3  <HTML>
4      <HEAD>
5          <TITLE>webst2: Variablen mit Smarty</TITLE>
6          <link rel="stylesheet" type="text/css" href="/css/webst.css">
7      </HEAD>
8      <BODY>
9          <HR>
10         <CENTER><H2>
11             Variablen mit Smarty
12         </H2></CENTER>
13         <H4><CODE>
14             lokaleVariable = { $lokaleVariable } <BR>
15             anArray[0] = { $anArray[0] } <BR>
16             anArray[3] = { $anArray[3] } <BR>
17             aHash[eins] = { $aHash.eins } <BR>
18             aHash[zwei] = { $aHash.zwei } <BR>
19             aHash[$lokaleVariable] = { $aHash.$lokaleVariable } <BR>
20         </CODE></H4>
21
22
23         <HR>
24     </BODY>
25 </HTML>
```

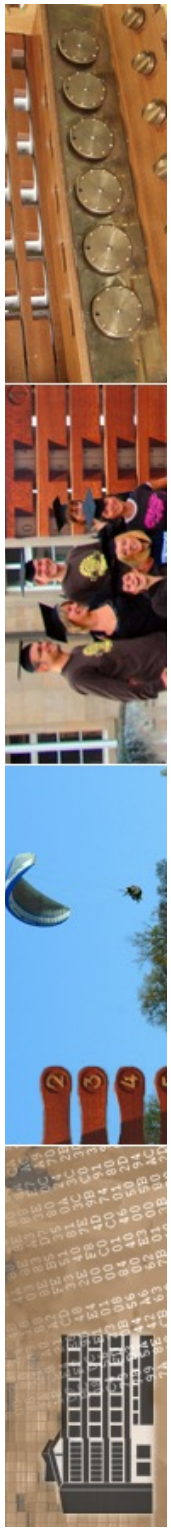
Writable | Insert | 1 : 1

.0.0.1/webst2/smarty/variablen.php

plate Engine | webst2: Variablen m... | 216 Farben (Netscape)

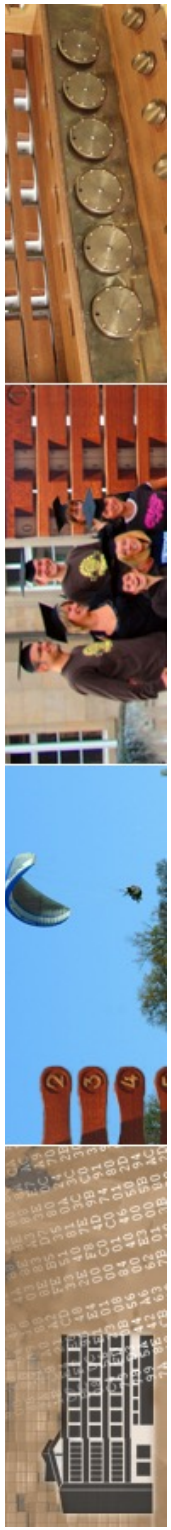
Variablen mit Smarty

lokaleVariable = drei
anArray[0] = 0
anArray[3] = drei
aHash[eins] = 1
aHash[zwei] = zwei
aHash[$lokaleVariable] = zweiundvierzig



die Referenz `$smarty`

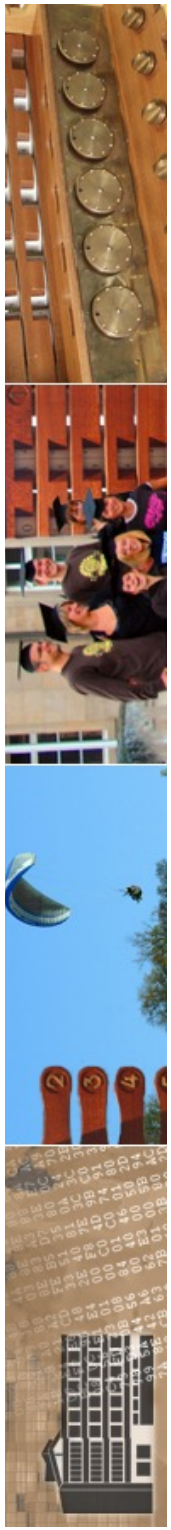
- die Variable $smarty referenziert auf ein Objekt, welches wesentliche Informationen zur Webapplikation speichert, etwa
  - `$smarty.server.SERVER_NAME`
  - `$smarty.env.PATH`
  - `$smarty.session.id`
  - `$smarty.cookies.benutzer`
  - `$smarty.version`
  - `$smarty.now`



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

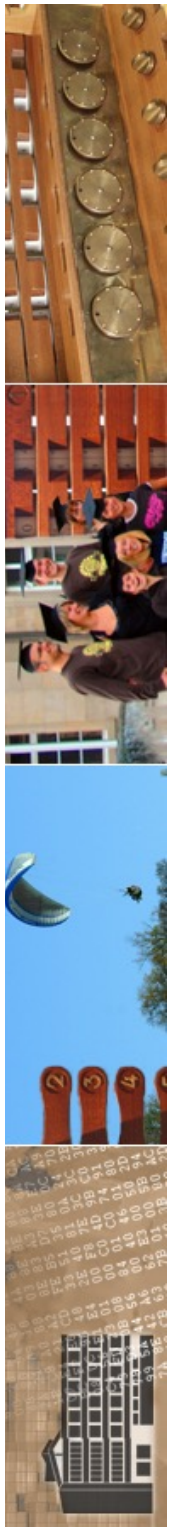
Smarty Funktionen

- Smarty verfügt über einige nützliche Funktionen, etwa
  - `lower`
  - `upper`
  - `nl2br`
  - `replace`
  - `regex_replace`
  - `...`



Laden einer Konfiguration

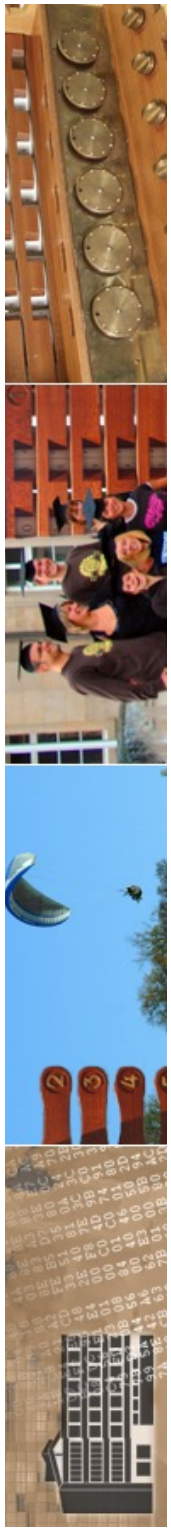
- die Direktive **`config_load`** lädt eine Konfiguration in Smarty
- zwingender Parameter:
  - **`file`**



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Kontrollstrukturen

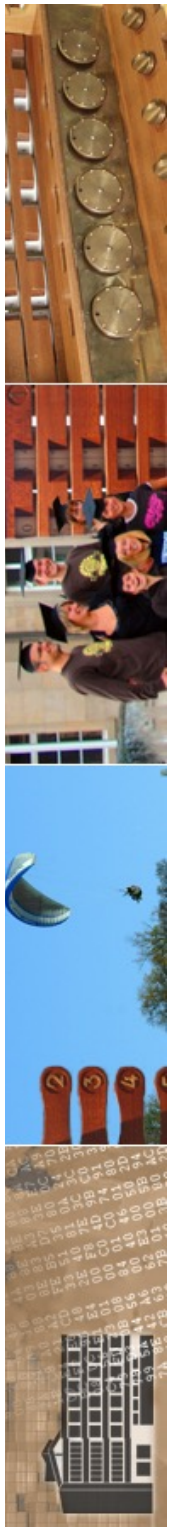
- Smarty stellt in den Templates auch Kontrollstrukturen bereit
  - Verzweigungen
  - Schleifen



Verzweigung

- Smarty stellt die übliche wenn-dann-sonst-Struktur mit elseif zur Verfügung
- Syntax:

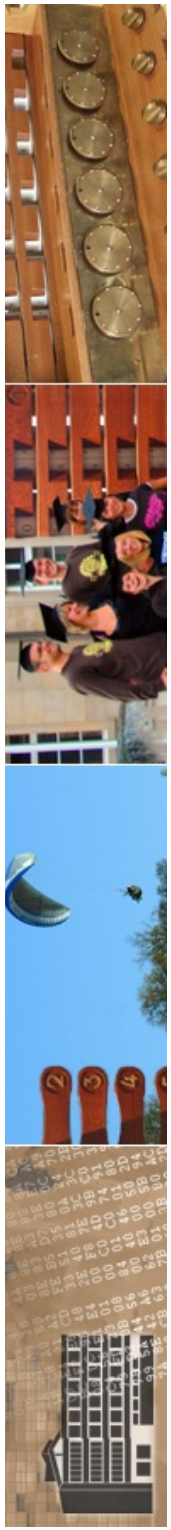
```
– {if $a == 1}
     a hat Wert 1
  {elseif $a == 2}
     a hat Wert 2
  {else}
     a hat anderen Wert
  {/if}
```



Schleifen

- foreach-Schleife durchläuft Array
- Syntax:
  - ...

```
{foreach from=$anArray item=i}
    Wert = <B>{$i}</B>
{/foreach}
```



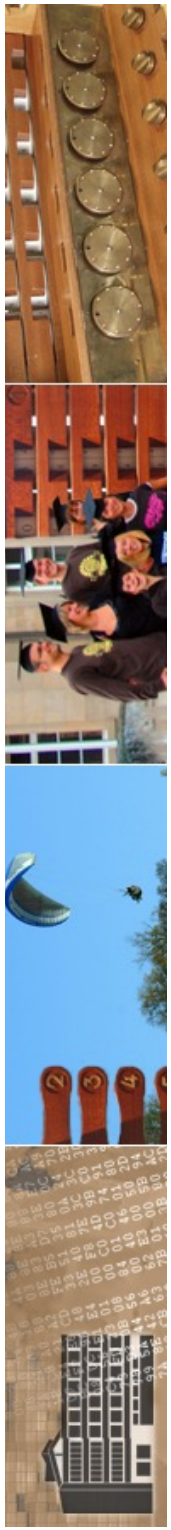
PHP - schleifen.php - Eclipse SDK

File Edit Navigate Search Project PHP/Apache Run Latex Window Help

variablen.tpl | variablen.php | debugging.php | schleifen.php | schleifen.tpl | »63

```
1  <?php
2  /*
3   * Created on 16.12.2005
4   *
5   * Fortgeschrittene Programmierung fuer das Internet
6   *
7   * Beispiel fuer Kontrollstrukturen in Smarty: Schleifen
8   */
9
10 require 'Smarty.class.php';
11
12 $smarty = new Smarty;
13 $smarty->template_dir="D:\\www\\webst2\\smarty\\templates";
14
15 $myArray = array(0,1,2,"drei");
16
17 $smarty->assign("anArray", $myArray);
18
19 $smarty->display('schleifen.tpl');
20
21 ?>
```

Writable | Smart Insert | 1 : 1



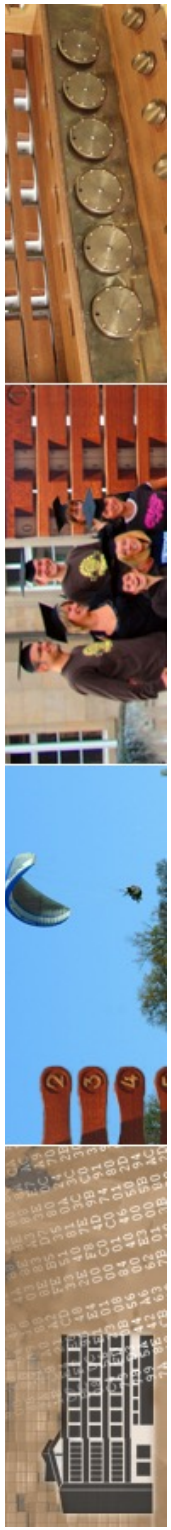
PHP - schleifen.tpl - Eclipse SDK

File Edit Navigate Search Project PHP/Apache Run Latex Window Help

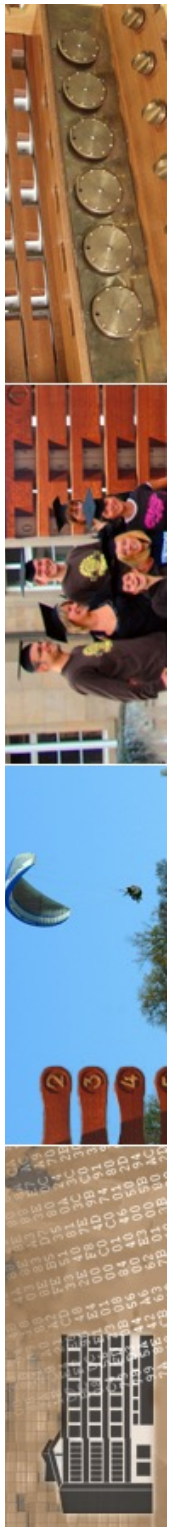
variablen.tpl | variablen.php | debugging.php | schleifen.php | schleifen.tpl

```
1  {* webst2: Beispiel fuer Kontrollstrukturen in Smarty: Schleifen *}
2
3  <HTML>
4      <HEAD>
5          <TITLE>webst2: Variablen mit Smarty</TITLE>
6          <link rel="stylesheet" type="text/css" href="/css/webst.css">
7      </HEAD>
8      <BODY>
9          <HR>
10         <CENTER><H2>
11             Schleifen mit Smarty
12         </H2>
13         <H4>
14             { foreach from=$anArray item=i}
15                 Wert = <B>{$i}</B><BR>
16             {/foreach}
17
18         </H4>
19         </CENTER>
20         <HR>
21     </BODY>
22 </HTML>
```

Writable | Insert | 1 : 1




webst2: Variablen mit Smarty - Mozilla Firefox
Datei Bearbeiten Ansicht Gehe Lesezeichen Extras Hilfe
http://127.0.0.1/webst2/smarty/schleifen.php
Schleifen mit Smarty
Wert = 0
Wert = 1
Wert = 2
Wert = drei
Fertig

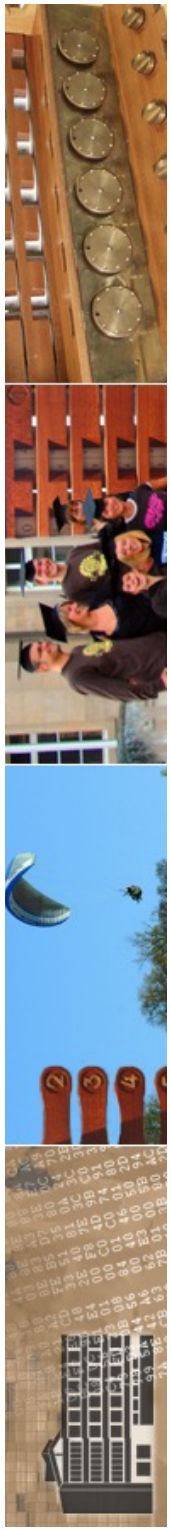


Importieren

- nützlich ist der Import von Dateien in die Template-Datei:

```
{  include file="footer.tpl"  }
```

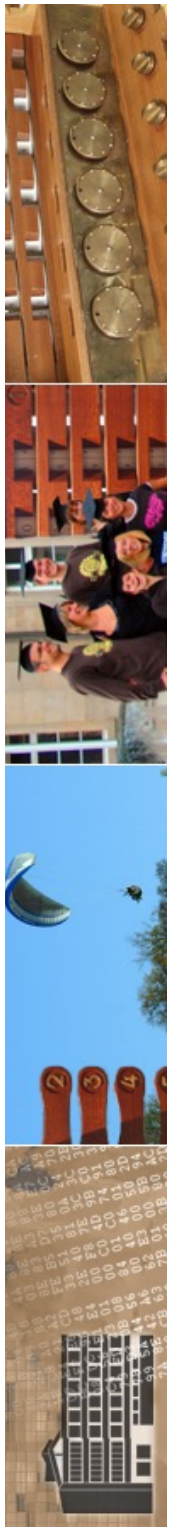
- neben include gibt es noch **`insert`**, nur findet hier kein Caching statt



Wertzuweisung innerhalb eines Templates

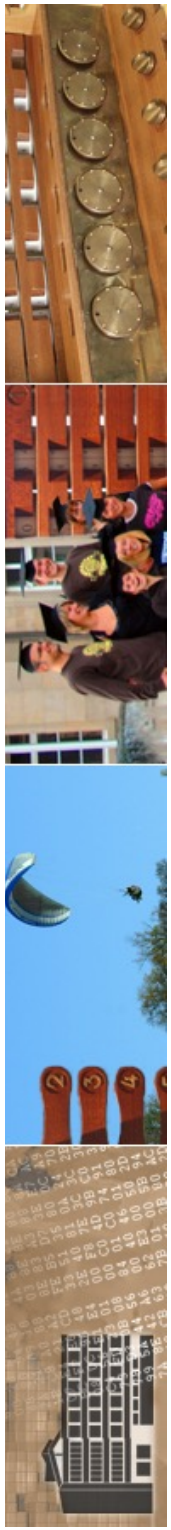
- auch innerhalb eines Templates kann einer Smarty-Variablen ein Wert zugewiesen werden:

```
{
 assign var="name" value="value"
}
```



die Anweisung `fetch`

- mittels `{ fetch file="datei" }` kann mit den Protokollen
  - http
  - ftp
  - (Zugriff auf eine Datei auf dem lokalen Filesystem)
- ein Dokument angefordert werden



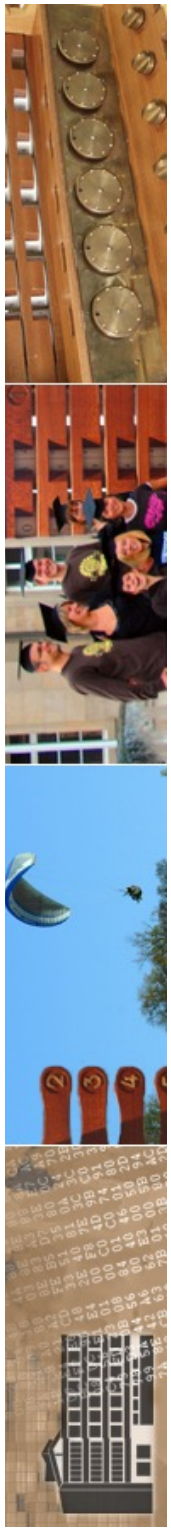
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

HTML-Output

- ähnlich wie die HTML-Ausgabefunktionen des PERL-Moduls CGI.pm bzw. Python-Modul CGI kann mit Smarty-Funktionen HTML-Code einfach erzeugt werden

- es steht eine Vielzahl derartiger Funktionen für die jeweiligen HTML-Elemente zur Verfügung

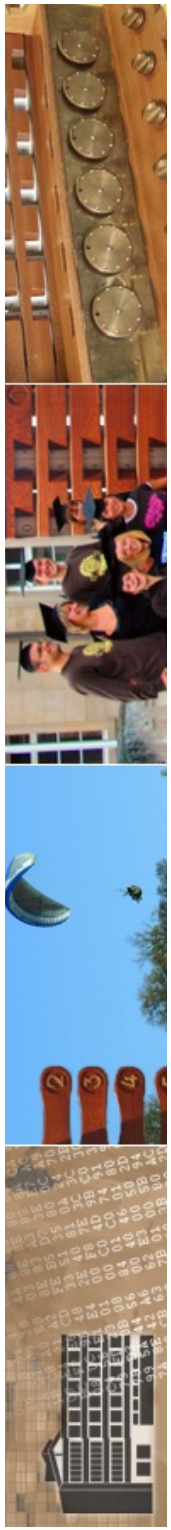
  – Funktionen **`html_xxx`**



Konfigurationsdateien

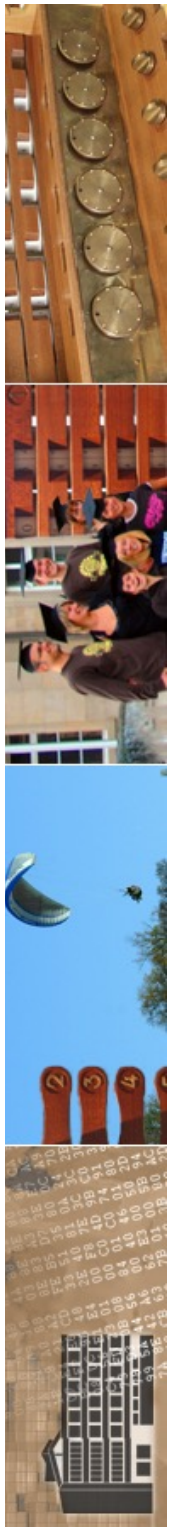
- mittels Konfigurations-Dateien können elegant Template-Variablen übersichtlich gesetzt werden

- wichtige Konfigurationen:
  - SMARTY_DIR
  - $template_dir
    - sinnvoller weise *nicht* unterhalb von „htdocs“
  - $cache_dir
  - ...



Caching

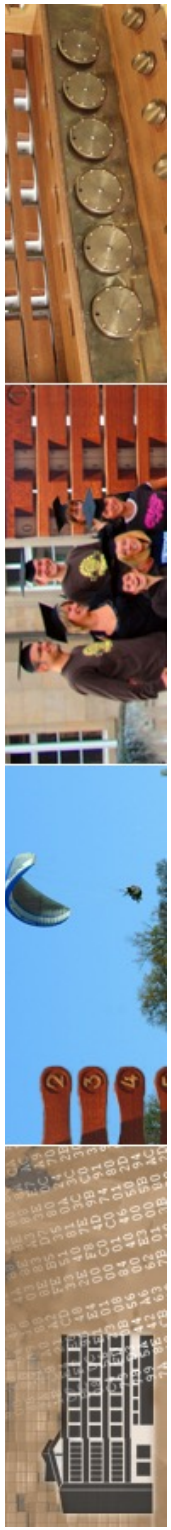
- Smarty hat umfassende Möglichkeiten zur Unterstützung von Caching
- bezieht sich auf die Methoden
  - `fetch()`
  - `display()`
- Steuerung über Variable $caching



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Plugins

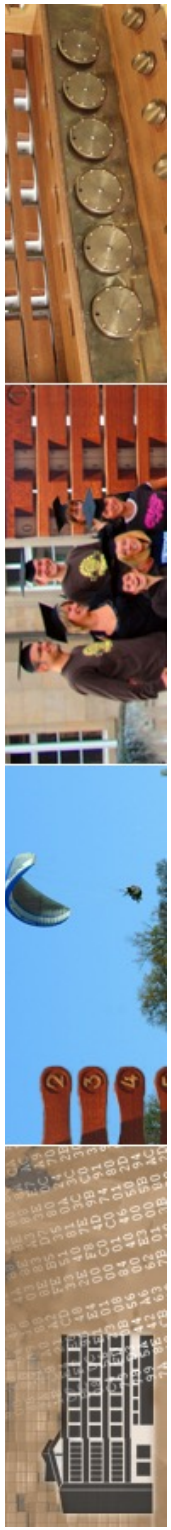
- Smarty kann durch Plugins erweitert werden
- diese sind in PHP geschrieben



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Smarty 3

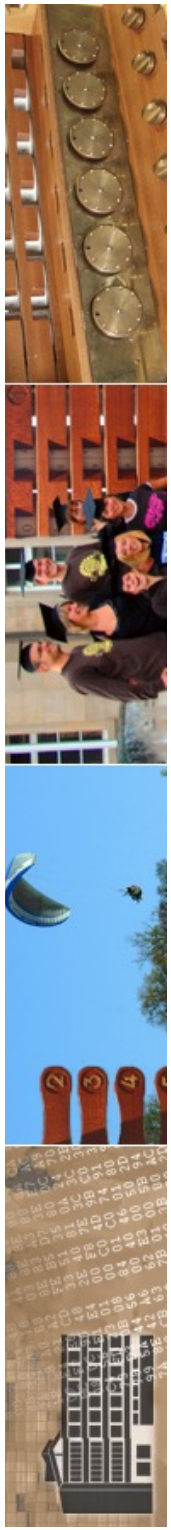
- benötigt PHP5 oder PHP7
- neuer Template-Parser erlaubt u.a. einfache Berechnungen und rekursive Funktionen im Template
- neues Smarty Data Object: `Smarty_Data`
- Vererbung von Templates
- Funktionsdefinition und -aufruf im Template
- weitgehend rückwärtskompatibel zu Smarty 2



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Diskussion

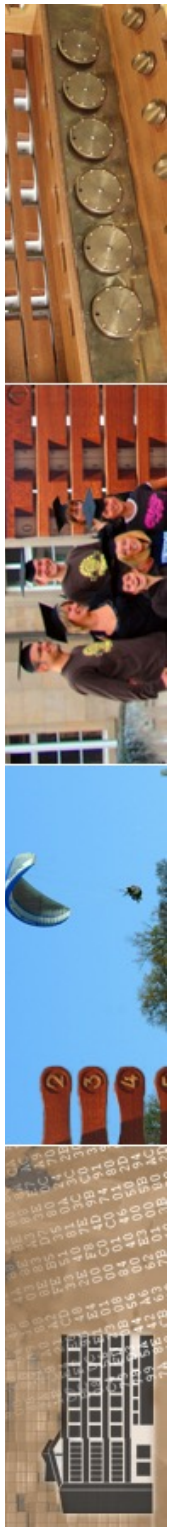
- Vorteile von Smarty
  - modulare Entwicklung durch Konfigurationsfiles, die leicht zu warten sind
  - Sicherheit: Templates enthalten kein PHP, so dass ein reiner Template-Designer keinen „Schaden“ anrichten kann
  - technische Vorteile
    - Caching, Debugging, Compiling, ...



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Diskussion: STL

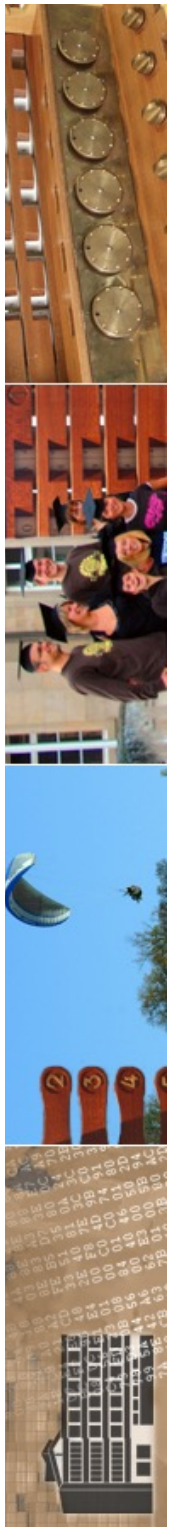
- Smarty Template Language (STL)
  - sehr eingeschränkte Sprache im Template
  - Nachteil: Weitere Sprache notwendig
  - Vorteile
    - leicht zu lernen
    - Erzwingt geradezu die Trennung View-Controller



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Ziel

- Trennung von Anwendungscode von Darstellung
  - Designer können den Anwendungscode nicht beschädigen
  - Fehlerbehandlung modular
  - klare Rollentrennung
  - der Designer braucht im wesentlichen nur HTML-Kenntnisse, um die Anwendung auf seine CI anzupassen

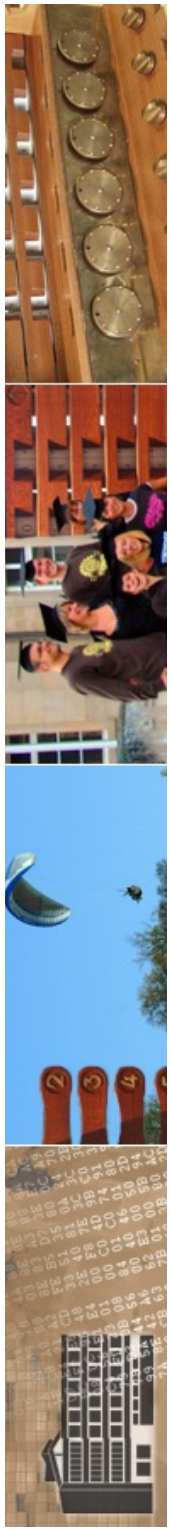


EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

What Smarty is Not

Smarty is not an application development framework. Smarty is not an MVC. Smarty is not an alternative to Laravel, Symfony, CodeIgniter, or any of the other application development frameworks for PHP.

Smarty is a template engine, and works as the (V)iew component of your application. Smarty can easily be coupled to any of the engines listed above as the view component. No different than any other software, Smarty has a learning curve. Smarty does not guarantee good application design or proper separation of presentation, this still needs to be addressed by a competent developer and web designer.



…und nun…

- haben wir ein einfaches Framework für die Komponente View kennen gelernt

- als nächstes: das Laminas-(ZEND-)Framework als zentrales Beispiel für PHP-Frameworks