

# PHP, zwischen HTML, SHTML und PERL

Sebastian Krüger

October 4, 2000

## Inhaltsverzeichnis

<b>1</b>	<b>Dynamische Webseiten</b>	<b>2</b>
<b>2</b>	<b>Was ist PHP?</b>	<b>2</b>
2.1	HTML . . . . .	2
2.2	SHTML . . . . .	3
2.3	Perl . . . . .	3
2.4	PHP . . . . .	4
<b>3</b>	<b>Geschichte</b>	<b>4</b>
3.1	Statistik . . . . .	5
<b>4</b>	<b>Möglichkeiten</b>	<b>5</b>
4.1	Datenbanken . . . . .	5
4.2	Filefunktionen . . . . .	6
4.3	Vieles anderes... . . . .	6
<b>5</b>	<b>Programmierung</b>	<b>6</b>
5.1	Hello World . . . . .	6
5.2	Kommentare . . . . .	6
5.3	Seperation der Anweisungen . . . . .	7
5.4	Variablen und Typen . . . . .	7
5.5	Operatoren . . . . .	7
5.6	Kontrollstrukturen . . . . .	8
5.7	Funktionen . . . . .	8

1	DYNAMISCHE WEBSEITEN	2
6	<b>BDC und Infotage - Seiten</b>	9
6.1	Die BeLUG - Seiten	9
6.2	Inhalt und Form - Trennung	12
6.3	Die Infotage-Seiten	15
7	<b>Weitere Informationen</b>	16
7.1	Websourcen	16
7.2	Dieser Vortrag	16

# 1 Dynamische Webseiten

Gründe für dynamischen Webseiten sind:

- Corporate Identity - alle Seiten haben das gleiche Layout aber verschiedene Inhalte
- Trennung von Inhalt und Form und damit bessere Administrierbarkeit von Inhalten
- Nutzung externer Informationsquellen wie Datenbanken, andere Webseiten u.ä.
- Applications Server - für beliebige Clients

PHP ist eine von vielen Möglichkeiten dies umzusetzen.

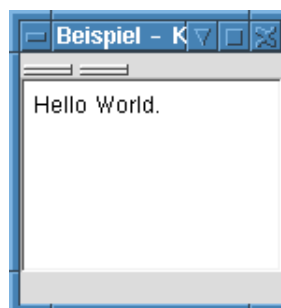
## 2 Was ist PHP?

### 2.1 HTML

Der HTML-Code (HyperText Markup Language) wird vom Webserver zum Browser geschickt.

HTML ist das, was der Browser versteht.

```
1 <html>
2   <head>
3     <title>Beispiel</title>
4   </head>
5   <body>
6     Hello World.
7   </body>
8 </html>
```



## 2.2 SHTML

SHTML (Serverparsed HTML) ist eine einfache Möglichkeit dynamische Elemente in die HTML-Seite zu bringen.

```

1 <html>
2   <head>
3     <title>Beispiel</title>
4   </head>
5   <body>
6     Hello World.
7   <hr>
8   <!--#config timefmt="%d.%m.%Y" -->
9   Letzte &Auml;nderung: <!--#flastmod file="bsp2.shtml" -->
10  </body>
11 </html>

```



Befehle: include, exec, if elif else endif, echo, flastmod, fsize, config, printenv, set. Keine Schleifen, rekursives Includen nicht erlaubt.

## 2.3 Perl

Perl ist eine Möglichkeit die Webseiten zu dynamisieren. Dazu gibt es zwei Formen:

- als CGI-Script oder
- als embedded Perl in HTML-Dateien

```

1 #!/usr/bin/perl
2
3 use CGI qw:/standard /;
4
5 print header,
6     start_html( "Beispiel" ),
7     "Hello world!",
8     end_html();
9
10 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
11 <HTML><HEAD><TITLE>Beispiel</TITLE>
12 </HEAD><BODY>Hello world!</BODY></HTML>

```

Perl kann alles, auf Dutzenden Wegen. Bei CGI-Programmen wird für jede Ausgabe "print" benutzt. Für kleine Probleme zu aufwendig.

## 2.4 PHP

PHP (PHP: Hypertext Preprocessor) ist ähnlich wie embedded Perl eine vollständige Scriptsprache die in HTML eingebettet wird.

```
1 <html>
2   <head>
3     <title>Beispiel</title>
4   </head>
5   <body>
6     <? for($i=1; $i<=3; $i++){ ?>
7       Hello World. <br>
8     <? } ?>
9   </body>
10 </html>
```



Die PHP-Codeteile wird dabei von `<?>` und `?>` eingeschlossen.

Ein Modul des Apache interpretiert nun bei entsprechender Einstellung den PHP-Code der Seiten mit ".php3"-Endung und schickt das Ergebnis dem fragenden Browser.

## 3 Geschichte

RASMUS LERDORF entwickelte im Herbst 1994 PHP für seine private Homepage.

Anfang 1995 war die erste Version für andere unter dem Namen "Personal Home Page Tools" verfügbar. Damals bestand PHP aus einem einfachen Parser der einigen Macros verstand.

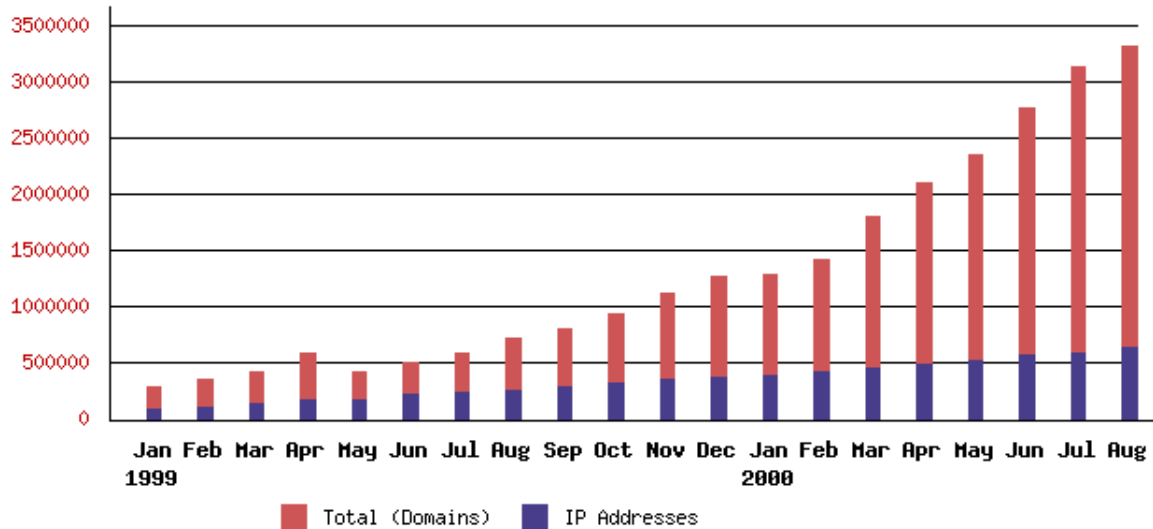
Der Parser wurde Mitte 1995 neu geschrieben zu PHP/FI Version 2. FI steht für "Form Interpreter". Außerdem kam zu den Tools eine mSQL-Anbindung. Danach stieg die Nutzerzahl erheblich, so dass sie Mitte 1997 bei ca. 50 000 lag.

1997 wurde das ganze Paket überarbeitet zu PHP3 (mit inzwischen gewachsener Entwicklergemeinde, vor allem von Zeev Suraski und Andi Gutmans). Es läuft nun unter diversen Webservern, beim Apache als "native Server Modul".

Seit dem 22. Mai 2000 gibt es PHP4 mit neuen Kern (Zend-Engine), Server Abstractions Layer und vielen anderen neuen Features.

### 3.1 Statistik

Die Nutzung von PHP auf Webseiten:



Quelle: Netcraft: <http://www.netcraft.com>

August 2000: 3314634 Domains auf 637746 IP Adressen

## 4 Möglichkeiten

PHP-Seiten können ziemlich dasselbe wie CGI-Skripte. Der wohl wichtigste Vorteil von dynamischen Webseiten ist die Möglichkeit Inhalt und Form zu trennen. Unterstützt wird dies durch diverse Features.

### 4.1 Datenbanken

Außer ODBC Support stellt PHP spezielle Funktionen für diverse Datenbanken Zugriffsfunktionen zu Verfügung.

Adabas D	Empress	Interbase	mSQL	Oracle	Solid	Unix dbm
dBase	Filepro	Informix	MySQL	PostgreSQL	Sybase	Velocis

### 4.2 Filefunktionen

`require()` und `include()` wirken ähnlich wie `#include` in C.

Funktionen wie `fopen`, `fget`, `sprintf`...

### 4.3 Vieles anderes...

GIFs erzeugen, HTTP-Authentification, Cookies, File Uploads, `fopen` über `http` und `ftp`, PDF-Erzeugung...

## 5 Programmierung

### 5.1 Hello World

```
1 <html>
2   <head>
3     <title> Hi ho </title>
4   </head>
5   <body>
6     <? print("Hello World"); ?>
7   </body>
8 </html>
```

Als Anfang nimmt man sich eine HTML - Datei und ersetzt Text durch PHP-Code. Der Code wird dabei durch `<?>` und `?>` vom HTML getrennt. Alternativ können die Codeteile auch durch `<?php>` und `?>`, `<%>` und `%>` oder `<script language="php">` und `</script>` ersetzt werden.

### 5.2 Kommentare

PHP unterstützt C, C++, und Shell - Style Kommentare.

```
1 <?
2
3 /* Dies ist ein C-Style Kommentar,
4    mehrzeilig */
5
6 // Dies ist ein C++ Kommentar (einzeilig)
7
8 # Dies ist ein Shell Kommentar (einzeilig)
9
10 ?>
```

### 5.3 Separation der Anweisungen

Die Anweisungen werden mit `;` getrennt. `?>` erspart das Semicolon.

## 5.4 Variablen und Typen

Sämtliche Variablen beginnen mit `$`. PHP unterstützt folgende Typen:

**floating-point numbers** (real,double,float) `$a = 1.2e3`

**integer** `$b = 42`

**string** `$c = "42"`

**arrays** `$d[2] = 3`

**object** Es gibt Classes und Objects...

Zur Typumwandlung braucht man Funktionen wie `intval()` (macht aus (u.a.) Strings Integer): `$b` und `$c` sind verschieden, aber `$b` und `intval($c)` sind gleich.

Möglich ist auch sogenanntes "Type casting": `$wert = (double) 10;` .  
Damit kann der Typ gewählt werden.

Funktionen zur Typerkennung: `is_int()`, `is_real()`, ...

Spezielle Funktion um Arrays zu bilden:

```
$a = array( "A", "B", array( "a" => 34, "b" => "C" ))
```

`phpinfo()` gibt alle vordefinierte Variablen sowie weitere Infos aus.

## 5.5 Operatoren

Unterstützt werden die üblichen Operatoren:

**Aritmetische:** `+`, `-`, `*`, `/`, `%`

**String:** `.`

**Zuweisung:** `=`, `.=`, `+=`, ...

**Bit:** `&`, `|`, `^`, `~`, `<<`, `>>`

**Vergleich:** `==`, `!=`, `<`, `>`, `<=`, `>=`, `===`

**Logische:** `and`, `or`, `xor`, `!`, `&&`, `||`

**Inc- und Decrement:** `++`, `--`

## 5.6 Kontrollstrukturen

- `if (expr) statm elseif (expr) statm else statm;`
- `while (expr) statm;`
- `do statm while (expr);`
- `for(expr1;expr2;expr3) statm;`
- `foreach(array_expr as $value) statm;`
- `switch($value) { case $v1 : statm; case $v2 : statm; ... }`

Alternative Schreibweisen: `if(expr): statm endif; .`

## 5.7 Funktionen

### Eigene Funktionen

Allgemeine Form:

```
1 function foo ($arg_1, $arg_2, ..., $arg_n)
2 {
3     echo "Example function.\n";
4     return $retval;
5 }
```

Beispiel:

```
1 $lastmodified = 0;
2
3 function dsetlastmodified ($dateiname)
4 {
5     $dateidate = filemtime($dateiname);
6     global $lastmodified;
7     if ($dateidate > $lastmodified) { $lastmodified = $dateidate; };
8 };
```

### Weitere wichtige Funktionen

**MySQL:** Datenbankzugriffsfunktionen wie:

```
1 $conn = mysql_connect($server,$user,$pass);
2 $result = mysql_query($conn,$sqlstring);
3 $wert = mysql_result($result,$zeile,$spalte);
```



**Regular Expressions:** Stringerkennung und Veränderung:

```
1 $myarray = split("%20",$QUERY_STRING,5);
2 $bool = preg_match("/\w+@\w+/", $email);
3 $neu_string = preg_replace("/\xE4/", "&auml;", $alt_string);
```

**Streams:** Öffnen anderer Files und Programme:

```
1 $EMAIL = popen("mail -s \" ".$betreff." \" ".$email, "w");
2 fputs($EMAIL, $text);
3 pclose($EMAIL);
4 $FH1 = fopen("/home/sk/file.txt", "r");
5 $FH2 = fopen("http://www.php.net/", "r");
6 $FH3 = fopen("ftp://user:password@ftp.example.de/", "w");
```

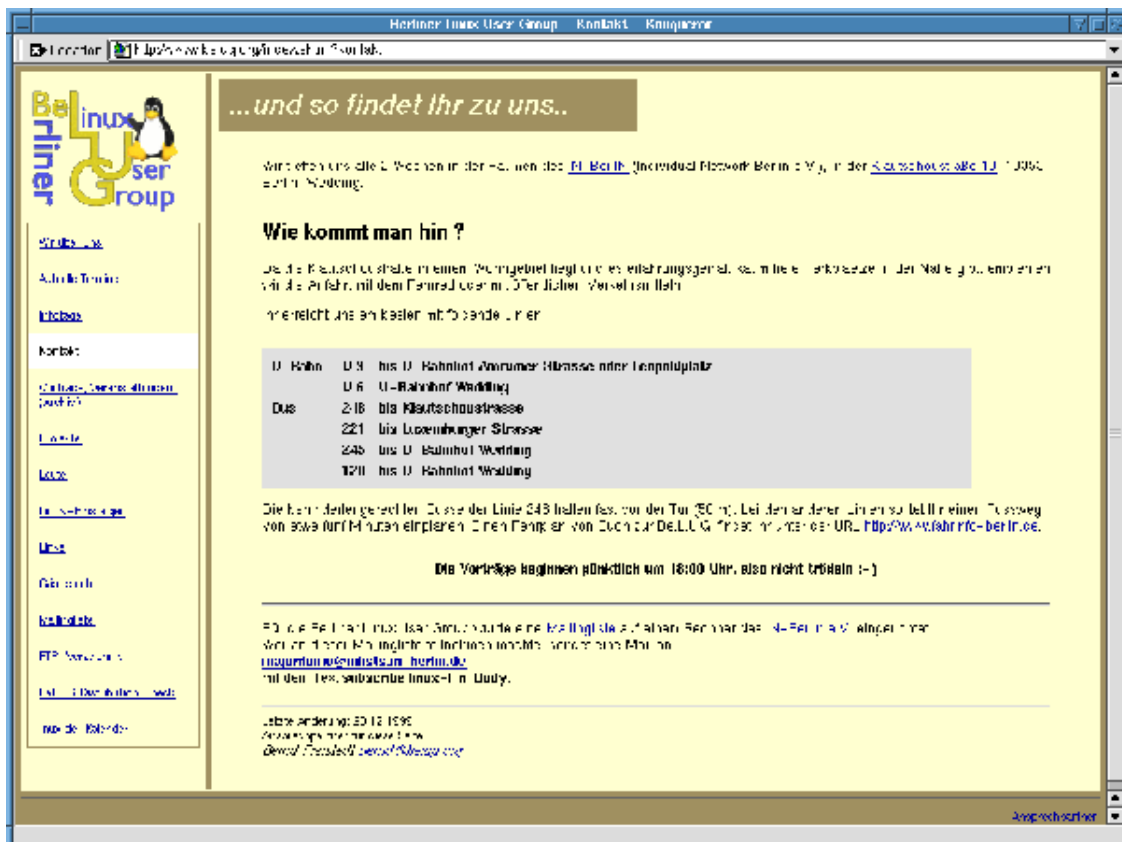
**Diverses:** Erzeugung und Ändern von GIF, JPEG, PNG und SWF; Erzeugung von PDFs;  
 Parsen von XML-Dateien; Verschicken von Mails

## 6 BDC und Infotage - Seiten

### 6.1 Die BeLUG - Seiten

	Dateien	Grösse	Kommentar	Code
BeLUG-Hauptseite	index.shtml	15115	772	14343
	incl/body_color	86		86
	incl/d_color	10		10
	incl/l_color	10		10
	incl/logo_color	10		10
	incl/w_color	10		10
	Summe:	15241	772	14469
BDC-Seiten	index.php3	6033	1515	4518
	index.dat	5813	1606	4207
	Summe:	11846	3121	8725

Beide Seiten enthalten 14 Menüpunkte und zum Webbrowser werden von beiden Systemen **6080 ± 20** Byte übermittelt.





## 6.2 Inhalt und Form - Trennung

Die Datei `index.shtml`. 1. Ausschnitt: Titelwahl und Anzeige

```

1 <title>
2 Berliner Linux User Group -
3 <!--#if expr="\$QUERY_STRING_UNESCAPED\" = \"about\" -->
4     Wir &uuml;ber uns
5 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"kontakt\" -->
6     Kontakt
7 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"projekte\" -->
8     Projekte, News &amp; Oldies
9 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"vortrag\" -->
10    Votr&auml;ge, Veranstaltungen Archiv
11 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"leute\" -->
12    BeLUG-Personendatenbank
13 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"einsteiger\" -->
14    Linux-Einsteiger
15 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"links\" -->
16    Links
17 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"gaestebuch\" -->
18    G&auml;stebuch
19 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"mailinglist\" -->
20    Mailing-Liste
21 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"bdc\" -->
22    BeLUG Distribution Check
23 <!--#else -->
24    Aktuelle Termine
25 <!--#endif -->
26 </title>

```

2. Ausschnitt: Einbinden des Seitentextes

```

1 <!--#elif expr="\$QUERY_STRING_UNESCAPED\" = \"bdc\" -->
2 <!--#include file=\"./bdc/bdci.shtml\"-->
3 <hr>
4 <font size="2">
5 Letzte &Auml;nderung: <!--#flastmod file=\"./bdc/bdci.shtml\"--><br>
6 <!--#include file="incl/maintain_bdc"-->
7 </font>

```

Diese 7 Zeilen kommen für jeden möglichen Menüpunkt einmal vor, d.h. hier ca. 14 mal. Die ganze Datei ist zu finden unter <http://www.belug.org/index.txt>.

Die Datei `index.dat`. Ausschnitt:

```

1 $default= "aktuelles";
2

```

```

3 $menupunkte = array(
4     ...
5     array(
6         "name"           => "aktuelles",
7         "indexmenu"     => "Aktuelles",
8         "ueberschrift"  => "Aktuelles",
9         "bdcimenu"      => "",
10        "incl"           => "aktuelles.php3",
11        "link"           => "index.php3"
12    ),
13    ...
14    array(
15        "name"           => "mailinglist",
16        "indexmenu"     => "Mailingliste",
17        "ueberschrift"  => "bdc-1: Die BDC-Mailingliste",
18        "bdcimenu"      => "Mailingliste",
19        "incl"           => "mailinglist.html",
20        "link"           => "index.php3?mailinglist"
21    ),
22    array(
23        "name"           => "tools",
24        "indexmenu"     => "Tools",
25        "ueberschrift"  => "Test Tools",
26        "bdcimenu"      => "Tools",
27        "incl"           => "tools/tools.php3",
28        "link"           => "index.php3?tools"
29    )
30 );

```

Ein Array beschreibt einen Menüpunkte. Weitere Einstellungen wie Farben, Logo, Titel befinden sich am Anfang der Datei.

Die Datei `index.php3`.

1. Ausschnitt: Einlesen der Daten, Menüpunktwahl

```

1  <?
2  #####
3  #
4  # Laden der Daten
5  #
6
7  include("index.dat");
8
9  #####
10 #
11 # $QUERY_STRING_ARRAY erstellen (aus Uebergabeparameter)

```

```

12 #
13
14 $QUERY_STRING_ARRAY = split("%20",$QUERY_STRING,2);
15
16 #####
17 #
18 # Sichern, dass $QUERY_STRING_ARRAY[0] ein Menüpunkt ist und
19 # gegebenenfalls auf den default-Wert setzen, berechnen von
20 # $menunumber
21 #
22
23 $bool = 1;
24 $menunumber=0;
25 while($menupunkte[$menunumber][name])
26 {
27     if ($QUERY_STRING_ARRAY[0] == $menupunkte[$menunumber][name])
28     {
29         $bool = 0;
30         break;
31     };
32     if ($default == $menupunkte[$menunumber][name])
33     {
34         $defaultnumber=$menunumber;
35     };
36     $menunumber++;
37 };
38
39 if ($bool)
40 {
41     $QUERY_STRING_ARRAY[0] = $default;
42     $menunumber=$defaultnumber;
43 };
44
45 ?>

```

## 2. Ausschnitt: Titelanzeige

```

1 <title>
2 <? print $Titel." - ".$menupunkte[$menunumber][ueberschrift]; ?>
3 </title>

```

## 3. Ausschnitt: Einbinden des Seitentextes:

```

1 <?
2 include($menupunkte[$menunumber][incl]);
3 dsetlastmodified($menupunkte[$menunumber][incl]);
4 ?>

```

```

5     <hr width=100%>
6     <font size="2"> Letzte &Auml;nderung: <?
7         print date("d.m.Y H:i:s", $lastmodified) ?> </font>

```

### 6.3 Die Infotage-Seiten

Die Seiten für die Infotage unterscheiden sich von den BDC-Seiten *nur* in Farbe und Inhalt. Beide benutzen dieselbe `index.php3` und unterscheiden sich nur in der Datei `index.dat`.

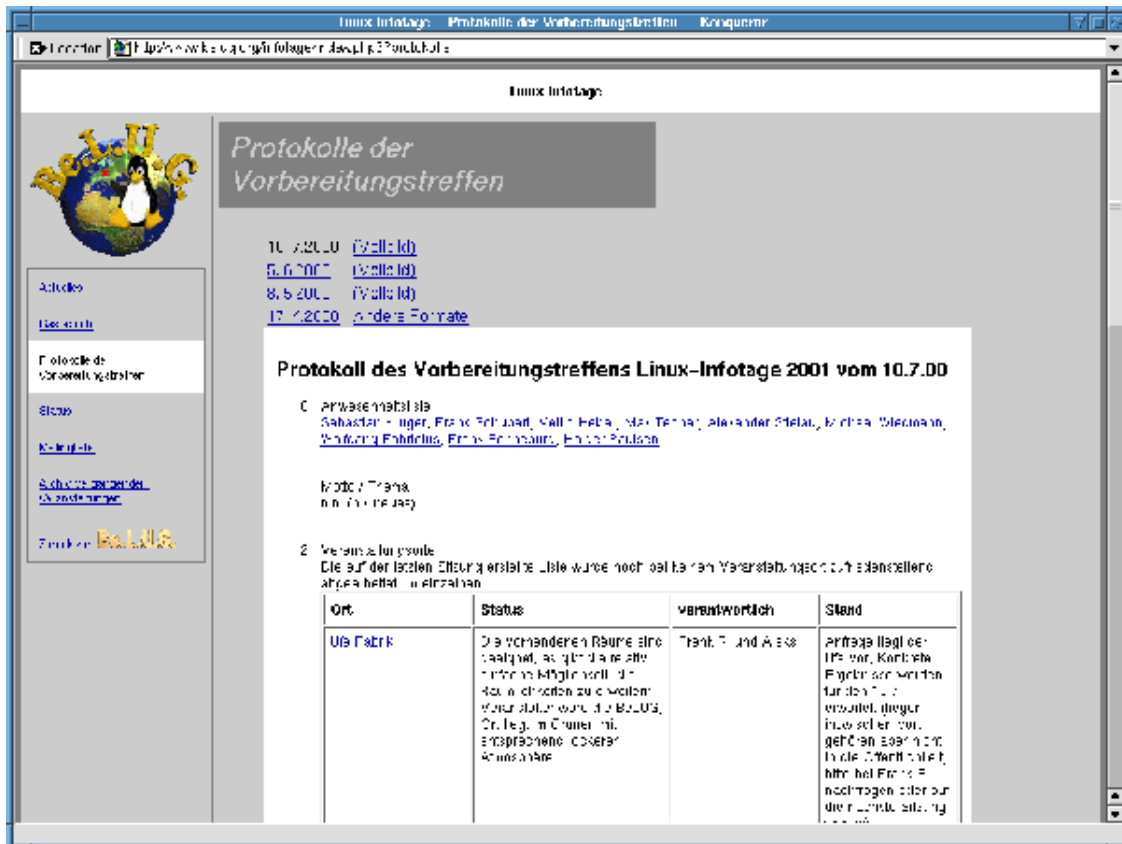
Ausschnitt aus der `index.dat` von den Infotage-Seiten:

```

1  $Titel = "Linux Infotage";
2  $Ueberschrift = "Linux Infotage";
3  $Logo = "/infotage/images/BeLUG.gif";
4  $Logo_groesse = "height=140 width=160";
5
6  $Hintergrund = "#808080";
7  $Zwischengrund = "#cccccc";
8  $F_text = "#000000";
9  $F_link = "#0000a0";
10 $F_vlink = "#0000a0";
11 $F_alink = "#0000ff";

```





## 7 Weitere Informationen

### 7.1 Websourcen

- <http://www.php.net>: Zentrale Weltweite Anlaufstelle
- <http://www.zend.com>: Zend - die Firma der Entwickler hinter PHP
- <http://www.php-center.de>: Deutschsprachige Seite mit News, Links, FAQ, ...
- <http://www.phpwelt.de/provider>: Liste deutscher Provider mit PHP-Unterstützung

Auf allen diesen Seiten findet man weitere Links, spezielle PHP-Skripte und das Manual.

### 7.2 Dieser Vortrag

<http://www.belug.org/~sk> : Hier findet man den Vortrag, Beispiele und Links.

**Kontakt:** [sk@belug.org](mailto:sk@belug.org)