



Grundlagen Internet-Technologien

INF3171

Cookies & Sessions

Version 1.0

01.07.2013





aktuelles





Erweiterungen

- wir betrachten zwei Erweiterungen:
 - **Personalisierung** der Web-Verbindung durch Cookies & Sessions
 - **AJAX**: Kombination von Client- und Server-Techniken





Individualisierung im Web

- die klassische Web-Site ist **zustandslos**
- wesentliche Eigenschaft
 - die Website kennt keine Vorgeschichte
- ...für viele Anwendungen ist dies nicht sinnvoll





Business-Applications

- Business-Applications brauchen ganz typisch einen Zustand wie
 - angemeldet
 - Rolle und Rechte
- trivial-Ansatz: hidden fields im HTML-Formular
 - Übungsaufgabe zum Ratespiel im Web





hidden fields (I)

- HTML-Formulare können über **versteckte Felder** verfügen
 - nicht sichtbar (werden nicht angezeigt)
 - werden aber ganz konventionell als key-value-Paare übertragen
 - Übertragung mit HTTP-Methoden **GET** und **POST**



hidden fields (II)

- Vorteile der Technik:
 - absolut unabhängig von Clientsoftware/-konfiguration
 - Client wird nicht belastet
 - reine HTML-Technik

- Nachteile der Technik:
 - „anstrengend“, da nach dem ersten Formular alle Formulare dynamisch generiert werden und alle „verstecketen“ Felder immer übertragen werden müssen
 - **im HTML-Source-Code sichtbar: "strg-u"**





bessere Ansätze

- Cookies und Sessions zur Personalisierung von Websites

– Beispiel:

neuer Browser, Besuch von amazon.de und dort suchen nach bestimmten Artikel

Browser schließen

neuer Besuch von Amazon: es werden passende Artikel angeboten





Cookies

- Zusatzinformationen werden *lokal auf dem Client abgelegt* und müssen so nicht jedesmal übertragen werden
 - Erweiterung des ursprünglichen HTTP-Protokolls 0.9





Cookie-Informationen

Suchen

Folgende Cookies sind auf Ihrem Computer gespeichert:

Website	Cookie-Name
typo3forum.net	
u-003-swsrf01.uni-tuebingen.de	
uimserv.net	
unicast.com	
uni-tuebingen.de	
uni-tuebingen.de	_utmc
uni-tuebingen.de	_utmz
uni-tuebingen.de	_utma
uni-tuebingen.de	fe_typo_user
urbanspoon.com	
usatoday.com	

Name: fe_typo_user
 Inhalt: 7a8ac474ebf6965093ee3924a4d6c024
 Host: www.uni-tuebingen.de
 Pfad: /
 Senden für: Jeden Verbindungstyp
 Gültig bis: am Ende der Sitzung

Schnellsuche

- heise.ivvbox.de
- hitbox.com
- jobs.perl.com
- mysql.com
- news.netcraft.com
- nexen.net
- paehl.com
- pecl.php.net
- perl.com
 - __qca: 49f44fa8-d839d-4aba2-6c12e
 - CP: null*
- perl.org
- portal.opera.com
- qooxdoo.org
- quantserve.com
- search.cpan.org
- sourceforge.net
- sun.com
- www.heise.de





Prinzip des Cookies

- ein Cookie ist ein **key-value-Paar**
- hat eine begrenzte Haltbarkeit
- ist Bestandteil des HTTP-Protokolls
- **wird im HTTP-Header übertragen**
- einfache Textdatei auf dem Client
- jedes Cookie gehört zu einer Website
- alle Browser speichern ihre Cookies getrennt
- für den jeweiligen Browser: **alle Browser-Tabs und Browser-Fenster haben die gleichen Cookies**
 - DOM: Cookie gilt für navigator-Objekt



Datenstruktur des Cookies

- name, value (durch "=" getrennt)
- version
- expires
- max-age
- domain
- path
- port
- comment
- commenturl
- secure
- discard





Cookies und Sicherheit

- keine wirkliche Gefährdung des Clients
 - aber: "Ausspionieren" des Clients
(auch von fremden Sites)
- einfache Manipulation des Cookies



- Gefahr für Diensteanbieter



Cookies in PHP

- Prinzipiell: Cookie ist Bestandteil von HTTP
 - Cookie wird *vor* dem Dokumentenbeginn `<HTML>` gesetzt
- PHP-Methode `setcookie (...)` setzt Cookie in PHP
 - Argument: key-value-Paar
 - „klassisches PHP“: Variable mit Bezeichner `$key` und Belegung `value` zur Verfügung
- Zugriff auf Cookies über das PHP-Superglobal `$_Cookie`





Syntax setcookie

- `bool setcookie (string name [, string value [, int expire [, string path [, string domain [, bool secure]]]])`
- name, value, path klar
- expire: Unix-timestamp, wann Cookie ungültig wird
- domain: (Sub-)Domainen, die das Cookie benutzen können
- secure: wenn true, dann Cookie-Informationen nur über https übertragbar





Beispiel: Zähler, Dauer

The screenshot shows a web browser window with the following content:

Grundlagen Internet-Technologien
Cookies mit PHP

Zähler: 13
Startzeitpunkt: 1308680240
Dauer der Session in Sekunden: 14



```
PHP - InternetTechnologien PHP/cookie.php - Eclipse
File Edit Navigate Search Project PHP/Apache Run Window Help
Quantum ... Database ... PHP Perl
fcgi.pl helloworld.php globalevariable.php MySQLi_student.php cookie.php x
1 <?php
2 /*
3  * Grundlagen Internet-Technologien
4  *
5  * Beispiel fuer Cookies mit PHP:
6  * - Cookie "start" merkt sich den Zeitpunkt des Seitenzugriffs
7  * - Cookie "count" die Zugriffe vom gleichen Client
8  * - wenn count nicht gesetzt ist, werden zunaechst beide gesetzt
9  * - alle Cookies bekommen eine "Lebensdauer" von 600 Sekunden
10 * - Verwendung des Superglobals $_COOKIE
11 */
12
13 // Setzen des Cookies
14 if(!isset($_COOKIE['start'])) {
15     $count = 1;
16     $start = time();
17     setcookie("start", $start, time()+600);          // setzen des Cookie "start"
18 } else
19     $count = $_COOKIE['count'] + 1;          // Wert (value) des Cookies 'count' + 1
20
21 setcookie("count", $count, time()+600);          // Setzen des count-Cookies
22 ?>
23 <html>
24     <head>
25         <title>Grundlagen Internet-Technologien: Cookies mit PHP</title>
26         <link rel="stylesheet" type="text/css" href="/css/webkompendium.css">
27         <link rel="shortcut icon" href="/favicon.ico">
28     </head>
29     <body>
30         <center>
31             <HR><H2>Grundlagen Internet-Technologien</H2><H3>Cookies mit PHP</H3><HR><H4>
32             Z&auml;hler:
33             <?php if (isset($_COOKIE['count'])) echo $_COOKIE['count']; else echo "0"; ?>
34             <BR>Startzeitpunkt:
35             <?php if (isset($_COOKIE['start'])) echo $_COOKIE['start']; else echo "0" ?>
36             <BR>Dauer der Session in Sekunden:
37             <?php
38                 if (isset($_COOKIE['start'])) $duration = time() - $_COOKIE['start'];
39                 else $duration = "0";
40                 echo "$duration";
41             ?>
42             </H4><HR></center>
43     </body>
44 </html>
```





Cookies in anderen Sprachen

- Perl: Modul CGI::Cookie

```
- my $cgi = new CGI;
  my $cookie = cgi->cookie(-name=>'name',
                          -value=>'wert');
```

- Python: Modul Cookie

- Ruby: ähnlich Perl





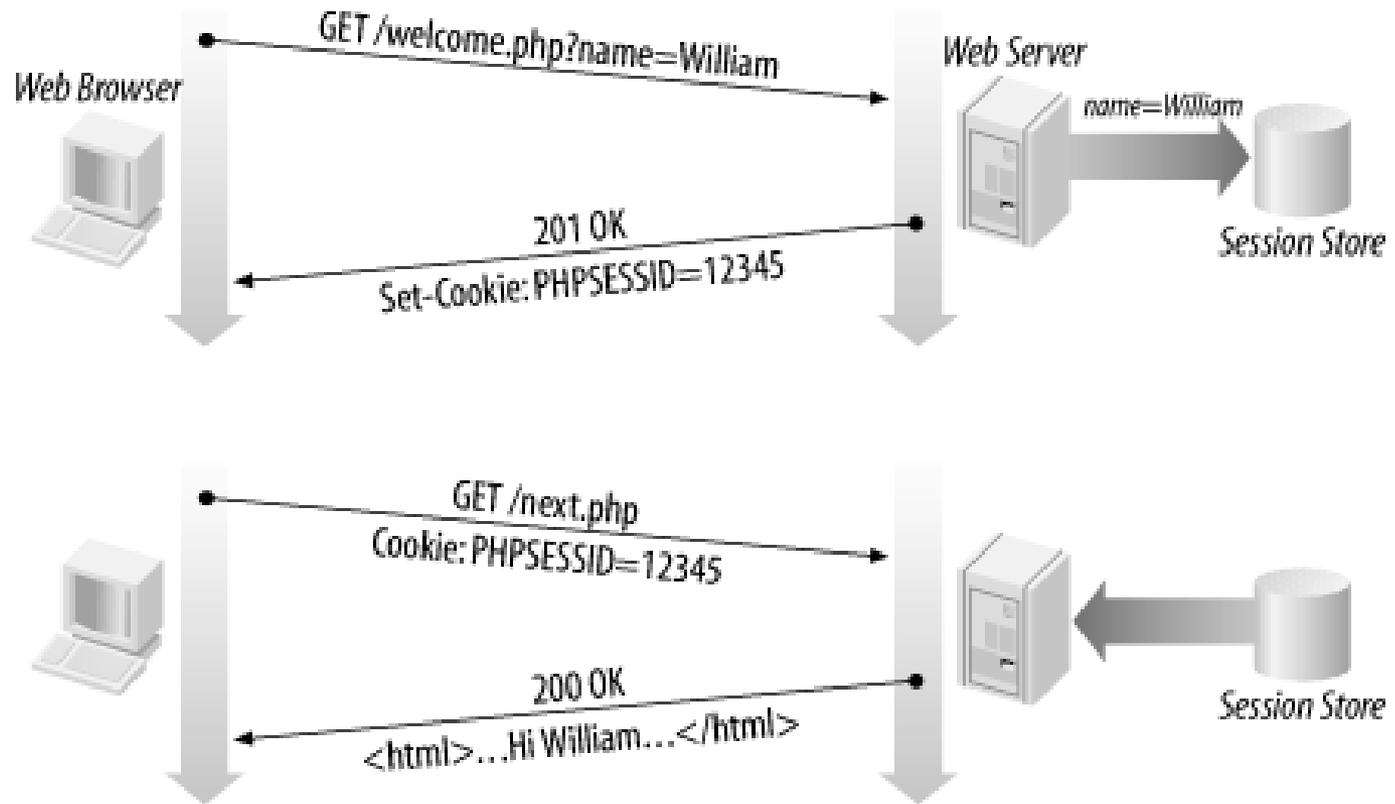
Sessions

- Cookie: Daten liegen auf dem Client
- Session: **Daten liegen auf dem Server**, werden über eine Session-ID der Browserinstanz zugeordnet
 - Kennzeichnung der Session über Session-ID
 - **diese Session-ID ist dann typischerweise ein Cookie auf dem Client**





Prinzip





Sessions in PHP

- die eigentlichen Session-Daten liegen auf dem Webserver
 - üblicherweise in /tmp, konfigurierbar über `session.save_path`
 - Daten sind dort sicher
- neue Session nur möglich vor der ersten Ausgabe
- Beginn einer Session:
 - `session_start()`
- Session-Daten in Suberglobal
 - `$_SESSION`
- Ende einer Session:
 - `session_destroy()`





```

1  <?php
2  /*
3   * Grundlagen Internet-Technologien
4   *
5   * Sessions in PHP: Anlegen einer Session
6   */
7
8   // das fuehrende "@" unterdrueckt moegliche Fehlermeldungen
9
10  @session_start();
11
12  $_SESSION["s_userName"]      = "Thomas Walter";
13  $_SESSION["s_userPermissions"] = "keine :-(";
14  ?>
15  <HTML>
16  <HEAD>
17      <TITLE>Grundlagen Internet-Technologien: Sessions mit PHP</TITLE>
18      <link rel="stylesheet" type="text/css" href="/css/webkompendium.css">
19      <link rel="shortcut icon" href="/css/favicon.ico">
20  </HEAD>
21  <BODY>
22      <CENTER>
23          <HR><H2>Grundlagen Internet-Technologien</H2>
24          <H3>Sessions in PHP</H3>
25          <HR><H4>Session in PHP gestartet!</H4><HR>
26      </CENTER>
27  </BODY>
28 </HTML>
    
```





```

PHP - InternetTechnologien PHP/session2.php - Eclipse
File Edit Navigate Search Project PHP/Apache Run Window Help
Quantum ... Database ... PHP Perl >>
fcgi.pl MySQLi_student.php cookie.php session1.php session2.php »_2
1 <?php
2 /*
3  * Grundlagen Internet-Technologien
4  *
5  * Sessions in PHP: Verwendung der Session
6  */
7
8     @session_start();
9
10    $name = $_SESSION['s_userName'];
11    $recht = $_SESSION['s_userPermissions'];
12
13    // wird in kein session_unregister() oder
14    // session_destroy() ausgeführt, bleiben die Daten erhalten!
15    ?>
16
17 <HTML>
18     <HEAD>
19         <TITLE>Grundlagen Internet-Technologien: Sessions mit PHP</TITLE>
20         <link rel="stylesheet" type="text/css" href="/css/webkompendium.css">
21         <link rel="shortcut icon" href="/css/favicon.ico">
22     </HEAD>
23     <BODY>
24         <CENTER>
25             <HR><H2>Grundlagen Internet-Technologien</H2>
26             <H3>Sessions in PHP</H3>
27             <HR><H4>Hallo <?php echo $name; ?><BR>
28             Ihre Rechte: <?php echo $recht; ?></H4><HR>
29         </CENTER>
30     </BODY>
31 </HTML>
    
```



Grundlagen Internet-1 x

127.0.0.1:82/session1.php

Grundlagen Internet-Technologien

Sessions in PHP

Session in PHP gestartet!

Grundlagen Internet-1 x

127.0.0.1:82/session2.php

Grundlagen Internet-Technologien

Sessions in PHP

Hallo Thomas Walter
Ihre Rechte: keine :-)





AT
N

```

PHP - InternetTechnologien PHP/session3.php - Eclipse
File Edit Navigate Search Project PHP/Apache Run Window Help
fcgi.pl MySQLi_student.php cookie.php session1.php session2.php session3.php
1 <?php
2 /*
3  * Grundlagen Internet-Technologien
4  *
5  * Sessions in PHP
6  */
7
8 @session_start();
9
10 if (!$_SESSION['start']) {
11     $_SESSION['start'] = time();
12     $_SESSION['count'] = 0;
13 }
14
15 ?>
16
17 <HTML>
18     <HEAD>
19         <TITLE>Grundlagen Internet-Technologien: Sessions mit PHP</TITLE>
20         <link rel="stylesheet" type="text/css" href="/css/webkompendium.css">
21         <link rel="shortcut icon" href="/css/favicon.ico">
22     </HEAD>
23     <BODY>
24         <CENTER>
25             <HR><H2>Grundlagen Internet-Technologien</H2>
26             <H3>Sessions in PHP</H3>
27             <HR><H4>Anzahl der Zugriffe: <?php echo ++$_SESSION['count']; ?><BR>
28             Dauer der Session: <?php echo (time() - $_SESSION['start']); ?><BR>
29             Session-ID: <?php echo session_id(); ?> </H4><HR>
30         </CENTER>
31     </BODY>
32 </HTML>
    
```

Writable

Smart Insert



The screenshot shows a web browser window with the following content:

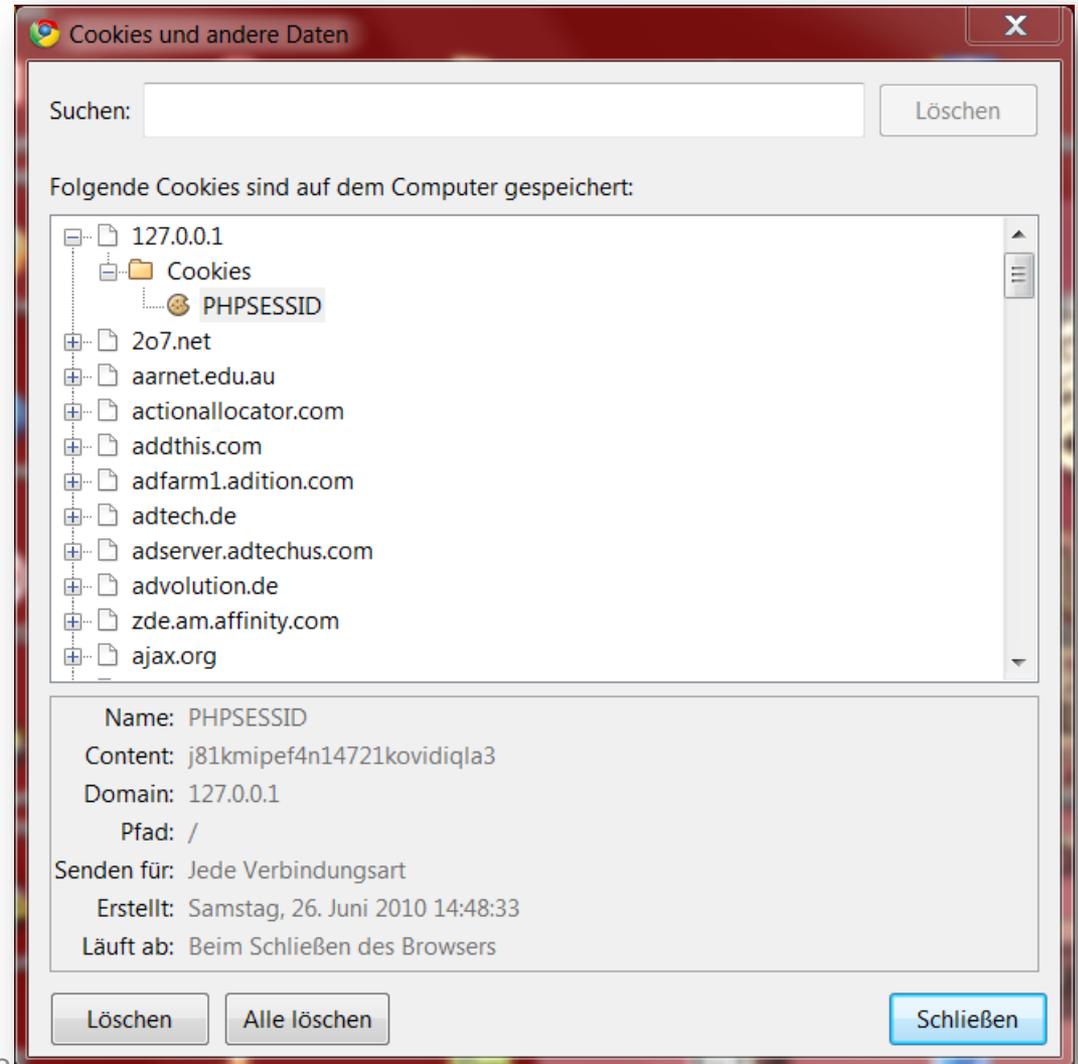
- Tab: Grundlagen Internet-1 x
- Address bar: 127.0.0.1:82/session3.php
- Page title: Grundlagen Internet-Technologien
- Section title: Sessions in PHP
- Statistics:
 - Anzahl der Zugriffe: 13
 - Dauer der Session: 21
 - Session-ID: dj0r2j51j16m851o9ldkglss36





die Session-ID

- ist ein Cookie
- mit dem Namen **PHPSESSID**





PHP-Sessions ohne Cookies

- es gibt auch Alternativen ohne Cookies:
 - Session-ID an die URL anfügen:
`Session->url()`
 - wieder als verstecktes Formular-Feld:
`Session->hidden()`





Sessions in Perl

- es gibt mehrere (viele) Möglichkeiten, Sessions in PERL (CGI) zu implementieren
 - Session-Funktionalität über verschiedene PERL-Module
 - Suche in CPAN





Modul CGI-Session

- aktuelle Version: 3.95
- Autor: Sherzod Ruzmetov
- die Session-Informationen können serverseitig wie schon in PHP verschieden abgelegt werden
 - File (default)
 - Database
 - ...
- Modul CGI::Session ist davon unabhängig, spezifischer Treiber wird dazugeladen wie bei DBI





CGI::Session::File

- Standard-Fall: Session-Information in Datei auf Server
- notwendiger Parameter: Verzeichnis für die Ablage der Datei
 - `use CGI::Session;`
 - `$session = new CGI::Session("driver:File, undef, {Directory=>'C:/temp'});`
- jede Session in separater Datei
 - Defaultname: `cgisess_%s` mit `%s` : session-id





Clientseitig: Cookie

- wie in PHP wird Clientseitig die session-ID im Cookie abgelegt:
 - `$cookie = $cgi->cookie (CGISESSIONID => $session->id) ;`
 - `print $cgi->header (-cookie=>$cookie) ;`





...damit...

- verliert die Website die Eigenschaft "Zustandslos"
- wesentliche Erweiterung unserer bisherigen Fähigkeiten und Möglichkeiten!





...und nun...

- haben wir zwei wesentliche Erweiterungen für die Web-Programmierung kennen gelernt:
Cookies & Sessions
- als nächstes:
Ajax

