

Dokumentation zur Datenbank auf pciwww.unizh.ch

mit MySQL und EPL

von

Rolf Pfister und Richard Gadola

Phys.Chem.Institut der Universität Zürich

17. Juni 2002

Inhaltsverzeichnis

1	Einleitung	- 1 -
1.1	Start der Homepage	- 1 -
1.2	Administrations-Seite	- 1 -
1.3	Programmierung	- 1 -
1.4	Dokumentation	- 1 -
2	Sourcen	- 2 -
2.1	aktuelle Quelldateien	- 2 -
2.2	alte Seiten	- 2 -
3	Datenbank	- 3 -
3.1	ERD	- 3 -
3.2	Tabellenbeschreibungen	- 4 -
4	Backup	- 11 -
4.1	Backup der Mysql-Datenbank	- 11 -
4.2	Alternatives Backup der Mysql-Datenbank	- 11 -
4.3	Mysql-Daten von Hand anschauen	- 11 -
5	Statistik mit Counter	- 12 -
6	Perl und EPL	- 13 -
6.1	Datenbankabfragen	- 14 -
6.2	Ganze Tabelle abfragen	- 15 -
6.3	Eingabe von Textfeldern	- 16 -
6.4	Bilder in Datenbank speichern und benutzen	- 17 -
7	Perl-Subroutinen	- 19 -
8	Chemikalienkartei	- 22 -
9	Probleme und Fallstricke	- 23 -
10	Fehler und Verbesserungsvorschläge	- 24 -
10.1	im öffentlichen Bereich	- 24 -
10.2	im Administrationsbereich	- 24 -
11	Literatur und Internetadressen	- 25 -

1 Einleitung

1.1 Start der Homepage

Wenn die Seite “<http://pciwww.unizh.ch/>” aufgerufen wird, wird `index.html` gestartet, welches dann mit einem Javascript auf `index.epl` umgeleitet wird.

Von dieser Homepage können die folgenden Bereiche angewählt werden:

- Home
- Forschung
- Vorlesungen
- Mitarbeiter
- Publikationen
- News
- Links

Dies ist der allgemein zugängliche Teil.

1.2 Administrations-Seite

Um Änderungen an der Datenbank vorzunehmen wird die Seite “<http://pciwww.unizh.ch/admin/>” aufgerufen. Diese Seite ist aus Sicherheitsgründen nur direkt von der Uni her aufrufbar. Zudem braucht man Benutzername und Passwort.

1.3 Programmierung

Fast alles ist mit Embperl realisiert. Dies ist Perl das in HTML eingebunden ist.

Grundlagen dazu: <http://perl.apache.org/embperl/>

Normales perl wird noch in `cgi-bin` verwendet um Bilder zu speichern und abzufragen. Siehe Kapitel “Bilder”.

1.4 Dokumentation

Diese Dokumentation ist gespeichert unter:

`r.pfister@pcicompaq3000:~/latex/datenbank/`

Es ist in \LaTeX geschrieben. Das “`makefile`” sagt wie man es übersetzen muss.

2 Sourcen

2.1 aktuelle Quelldateien

Die aktuellen Dateien sind im Verzeichnis “/usr/local/apache/htdocs/” auf pciwww.unizh.ch zu finden. Das heisst wenn man zum Beispiel “http://pciwww.unizh.ch/index.epl” eingibt, wird die Datei “pciwww.unizh.ch:/usr/local/apache/htdocs/index.epl” aufgerufen.

CGI-Scripts (zum Beispiel um Bilder von der Datenbank zu lesen) müssen im Verzeichnis “/usr/local/apache/bin” abgelegt werden.

Die Quelldateien von Richard Gadola befinden sich direkt in “/usr/local/apache/htdocs/”.

Die Quelldateien von Rolf befinden sich in “home/pfister/rolf/pci/”. Es gibt dort ein “makefile”, so dass man nur “make” zu tippen braucht um allfällig korrigierte Dateien an die richtigen Stellen zu kopieren.

Die Dateien mit der Endung **.eplh** sind Include-Dateien für epl (eigene Erfindung). Alles was zwischen [#include name.eplh #] und [#endinclude#] steht wird von “eplinclude” (ein C-Programm) durch den Inhalt von name.eplh ersetzt. Um “eplinclude” auf alle epl-Dateien anzuwenden sollte man nach Änderungen in eplh-Dateien “make include” machen.

Die Quellen für den Admin-Bereich sind in “home/pfister/rolf/pci/admin/”. Für Unter-Bereiche gibt es dort noch die Unterverzeichnisse “forsch/”, “link/” und “pass/” (immer mit makefile).

Die Quellen für die Statistik sind in “home/pfister/rolf/pci/counter/”.

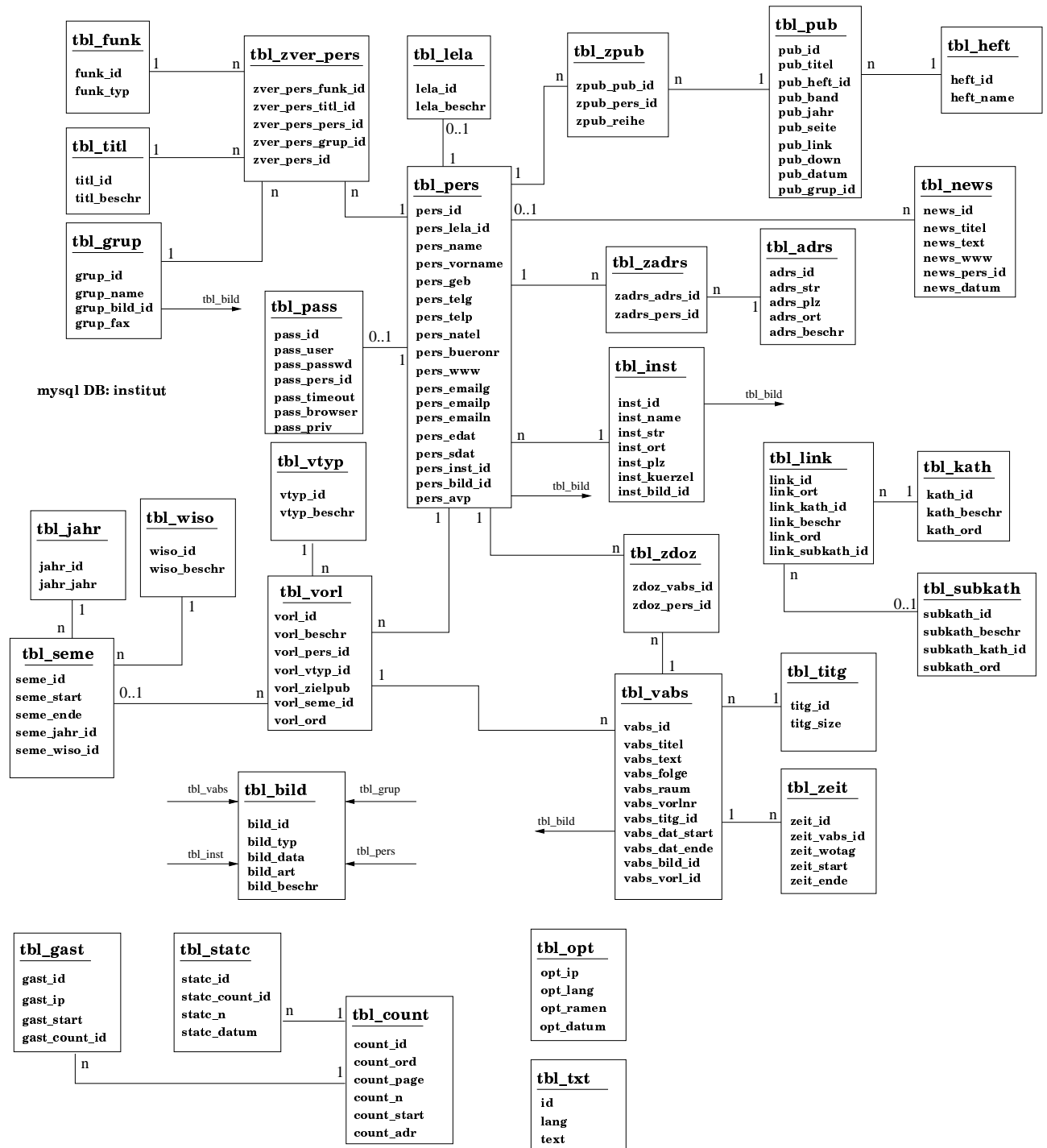
2.2 alte Seiten

Einige alte Seiten sind noch in “/usr/local/apache/htdocs/pci/” zu finden. Dabei ist mindestens “pfister/formelsammlung.epl” noch aktuell.

Auf “/usr/local/apache/htdocs/pci/seeger/” ist noch die alte (aktuelle) Homepage von Prof.Seeger zu finden. Sie wird von Dirk Kurzbuch verwaltet. Es gibt deshalb den Softlink “/home/kurzbuch/public_html/” wo er seine Sachen per ftp hineinkopiert.

3 Datenbank

3.1 ERD



3.2 Tabellenbeschreibungen

dual

Feldname	Beschreibung	Typ
X	Minimale Tabelle zur Abfrage einer Funktion die eigentlich keine Tabelle brauchen würde (z.B. SYSDATE()). Einziger Eintrag: 'Y'	String

tbl_funk

Feldname	Beschreibung	Typ
funk_id	Primärschlüssel	Integer
funk_typ	Typ der Funktion. Z.B. Administrator, Instituts Vorsteher, Koordinator usw.	String

tbl_titl

Feldname	Beschreibung	Typ
titl_id	Primärschlüssel	Integer
titl_beschreibung	Titel die eine Person bekleidet, z.B. Doktorant, Professor usw.	String

tbl_grup

Feldname	Beschreibung	Typ
grup_id	Primärschlüssel	Integer
grup_name	Name der Gruppe. Z.B. Huber, Fischer usw.	String
grup_bild_id	Gruppenlogo	Integer
grup_fax	FAX-Nummer	String

tbl_pers

Feldname	Beschreibung	Typ
pers_id	Primärschlüssel	Integer
pers_lela_id	Fremdschlüssel auf tbl_lela	Integer
pers_name	Name der Person	String
pers_vorname	Vorname der Person	String
pers_geb	Geburtsdatum der Person	Datum
pers_telg	Telefon Geschäft	String
pers_telp	Private Telefonnummer	String
pers_natel	Natelnummer	String
pers_bueronr	Büronummer, z.B. Y34-K-24	String
pers_www	Zusätzliche Homepage von dieser Person	String
pers_emailg	Email Adresse vom Geschäft	String
pers_emailp	Private Email Adresse	String
pers_emailn	Email Adresse nach dem Arbeitsverhältnisses. Kann gebraucht werden um späteren kontakt aufzunehmen	String
pers_edat	Datum der Kündigung	Datum
pers_sdat	Datum der Einstellung	Datum
pers_inst_id	Fremdschlüssel auf Tabelle tbl_inst	Integer
pers_bild_id	Bild der Person	Integer
pers_avp	Abkürzung des Vornamens für Publikationen	String

tbl_lela

Feldname	Beschreibung	Typ
lela_id	Primärschlüssel	Integer
lela_beschr	Der Lebenslauf	String

tbl_pass

Feldname	Beschreibung	Typ
pass_id	Primärschlüssel	Integer
pass_user	Loginname von einem Benutzer. Dieser kann sich vom Tatsächlichen Namen unterscheiden	String
pass_passwd	verschlüsseltes Passwort zum Einloggen in den gesicherten Bereich	String
pass_pers_id	Fremdschlüssel auf die Tabelle tbl_pers.	Integer
pass_timeout	Wird gespeichert wann die Session geöffnet wurde. Wird benötigt um einen Timeout zu realisieren	Integer
pass_browser	Hier wird die IP-Adresse des Browser gespeichert. Wird benötigt um den User zu identifizieren	String
pass_priv	Privilegien für www-Administration	Integer

tbl_inst

Feldname	Beschreibung	Typ
inst_id	Primärschlüssel	Integer
inst_name	Name des Instituts	String
inst_str	Strsse des Instituts	String
inst_ort	Ort des Instituts	String
inst_plz	Postleitzahl des Instituts	String
inst_kuerzel	Der Kürzel des Instituts	String

tbl_vorl

Feldname	Beschreibung	Typ
vorl_id	Primärschlüssel	Integer
vorl_beschr	Beschreibung der Vorlesung im Ganzen	String
vorl_pers_id	Femdschlüssel zur Person. Die Kontaktperson für die gesampte Vorlesung	Integer
vorl_vtyp_id	Fremdschlüssel zur Tabelle tbl_vtyp	Integer
vorl_zielpub	Zielpublikum dieser Vorlesung	String
vorl_ seme_id	Fremdschlüssel zur Tabelle tbl_ seme	Integer
vorl_ord	Ordnungsnummer zum Sortieren z.B. in forschung.epl	Integer

tbl_vtyp

Feldname	Beschreibung	Typ
vtyp_id	Primärschlüssel	Integer
vtyp_beschr	Beschreibung der Vorlesung. Z.B. Seminar, Vorlesung usw.	Integer

tbl_ seme

Feldname	Beschreibung	Typ
seme_id	Primärschlüssel	Integer
seme_start	Start des Semesters	Datum
seme_ende	Ende des Semesters	Datum
seme_jahr_id	Fremdschlüssel auf die Tabelle tbl_jahr	Integer
seme_wiso_id	Fremdschlüssel auf die Tabelle tbl_wiso	Integer

tbl_jahr

Feldname	Beschreibung	Typ
jahr_id	Primärschlüssel	Integer
jahr_jahr	Jahreszahl	Integer

tbl_wiso

Feldname	Beschreibung	Typ
wiso_id	Primärschlüssel	Integer
wiso_beschr	Winter oder Sommersemester	String

tbl_vabs

Feldname	Beschreibung	Typ
vabs_id	Primärschlüssel	Integer
vabs_titel	Titel des Abschnittes	String
vabs_text	Beschreibung des Abschnittes	String
vabs_folge	Die Reihenfolge der Abschnitte	Integer
vabs_raum	Der Raum, in der die Vorlesung statt findet	String
vabs_vorlnr	Die Vorlesungsnummer	String
vabs_titg_id	Fremdschlüssel auf die Tabelle tbl_titg	Integer
vabs_dat_start	Start der Vorlesung	Datum
vabs_dat_ende	Ende der Vorlesung	Datum

tbl_zvabs

Feldname	Beschreibung	Typ
zvabs_vabs_id	Fremdschlüssel auf die Tabelle tbl_vabs	Integer
zvabs_vorl_id	Fremdschlüssel auf die Tabelle tbl_vorl	Integer

tbl_titg

Feldname	Beschreibung	Typ
titg_id	Primärschlüssel	Integer
titg_size	Titelgrösse	String

tbl_zeit

Feldname	Beschreibung	Typ
zeit_id	primärschlüssel	Integer
zeit_vabs_id	Fremdschlüssel auf die Tabelle tbl_vabs	Integer
zeit_wotag	Wochentag der Vorlesung	String
zeit_start	Startzeit	time
zeit_ende	Endzeit	time

tbl_zdoz

Feldname	Beschreibung	Typ
zdoz_vabs_id	Fremdschlüssel auf die Tabelle tbl_vabs	Integer
zdoz_pers_id	Fremdschlüssel auf die Tabelle tbl_pers	Integer

tbl_pub

Feldname	Beschreibung	Typ
pub_id	Primärschlüssel	Integer
pub_titel	Titel der Publication	String
pub_heft_id	Fremdschlüssel auf die Tabelle tbl_heft	Integer
pub_band	In welchem Band ist die Publication erschienen	String
pub_jahr	In welchem Jahr ist die Publication erschienen	String
pub_seite	Auf welcher Seite in diesem Band ist die Publication erschienen	String
pub_link	Evt. einen Link auf die Zeitschrift	String
pub_down	Evt. einen Link zum downloaden der Publikation	String
pub_datum	Erscheinungsdatum der Publikation	Datum
pub_grup_id	Gruppe unter der die Publikation aufgeführt werden soll	Integer

tbl_heft

Feldname	Beschreibung	Typ
heft_id	Primärschlüssel	Integer
heft_name	Name der Zeitschrift, in der die Publikation veröffentlicht wurde	String

tbl_zpub

Feldname	Beschreibung	Typ
zpub_pub_id	Fremdschlüssel auf die Tabelle tbl_pub	Integer
zpub_pers_id	Fremdschlüssel auf die Tabelle tbl_pers	Integer
zpub_reihe	Sortierreihenfolge	Integer

tbl_fors

Feldname	Beschreibung	Typ
fors_id	Primärschlüssel	Integer
fors_beschr	Beschreibung der Forschungsaufgaben	String

tbl_zfors

Feldname	Beschreibung	Typ
zfors_fors_id	Fremdschlüssel auf die Tabelle tbl_fors	Integer
zfors_pers_id	Fremdschlüssel auf die Tabelle tbl_pers	Integer

tbl_adrs

Feldname	Beschreibung	Typ
adrs_id	Primärschlüssel	Integer
adrs_str	Private Strasse	String
adrs_plz	Private PLZ	String
adrs_ort	Privater Ort	String
adrs_beschr	Beschreibung der Art der Privaten Adresse. Z.B. Ferienwohnung etc.	String

tbl_zadrs

Feldname	Beschreibung	Typ
zadrs_adrs_id	Fremdschlüssel auf die Tabelle tbl_adrs	Integer
zadrs_pers_id	Fremdschlüssel auf die Tabelle tbl_pers	Integer

tbl_link

Feldname	Beschreibung	Typ
link_id	Primärschlüssel	Integer
link_ort	Html-Adresse des Lins	String
link_kath_id	Fremdschlüssel auf Hauptkategorie	Integer
link_beschr	Beschreibung des Links	String
link_ord	Ordnungsnummer für Reihenfolge bei Darstellung	Integer
link_subkath_id	Fremdschlüssel auf Unterkategorie	Integer

tbl_kath

Feldname	Beschreibung	Typ
kath_id	Primärschlüssel	Integer
kath_beschr	Beschreibung der Kategorie. Z.B. Chemie, Schweiz usw.	String
kath_ord	Ordnungsnummer für Reihenfolge bei Darstellung	Integer

tbl_subkath

Feldname	Beschreibung	Typ
subkath_id	Primärschlüssel	Integer
subkath_beschr	Beschreibung der Kategorie	String
subkath_ord	Ordnungsnummer für Reihenfolge bei Darstellung	Integer
subkath_kath_id	übergeordnete Hauptkategorie	Integer

tbl_count

Feldname	Beschreibung	Typ
count_id	Primärschlüssel	Integer
count_ord	Ordnungsnummer, Reihenfolge wie im Java-Applet erscheint	Integer
count_page	Name der Seite die gezählt werden soll	String
count_n	Zählerstand	Integer
count_start	Startdatum	Datum
count_adr	Adresse der zu zählenden Seite	String

tbl_statc

Feldname	Beschreibung	Typ
statc_id	Primärschlüssel	Integer
statc_count_id	Fremdschlüssel auf den Zähler	Integer
statc_n	Zählerstand wie er am entsprechenden Datum war	Integer
statc_datum	Datum des Zählerstands	Datum

tbl_gast

Feldname	Beschreibung	Typ
gast_id	Primärschlüssel	Integer
gast_ip	IP-Adresse des Besuchers	String
gast_start	Startzeit des Besuchs	String
gast_count_id	Fremdschlüssel auf den Zähler der besuchten Seite	Integer

tbl_opt

Feldname	Beschreibung	Typ
opt_ip	IP-Adresse des Besuchers (Primärschlüssel)	String
opt_lang	Sprache (bisher nur "de" oder "en")	String
opt_rahmen	Rahmen anzeigen ("on" oder "off", bisher noch nicht benutzt)	String
opt_datum	Datum, mehr als 1 Tag alte Einträge werden gelöscht	Datum

tbl_txt

Feldname	Beschreibung	Typ
id	Primärschlüssel zusammen mit lang (gleiche Texte in verschiedenen Sprachen haben die gleiche id)	Integer
lang	Sprache (z. B. "de" oder "en")	String
text	Der Text in der entsprechenden Sprache	String

4 Backup

Mit folgendem crontab-Eintrag (für Änderungen als root “crontab -e” aufrufen) wird jeden Tag um 22:30 ein Backup der mysql-Datenbankeinträge gemacht, tar-Archive von /home und /usr/local/apache erzeugt, und in “pciwwwbackup/” sowie (mit smbmount) auf pcicompaq3000 gespeichert:

```
30 22 * * * /root/backupDB
```

Weitere Einträge in crontab aktualisieren jeweils die Statistik-Counter:

```
55 23 * * * /usr/bin/perl /usr/local/apache/htdocs/pqi/counter/statistik.pl
```

```
0 1 * * * /usr/local/apache/htdocs/pqi/counter/stat
```

```
5 1 * * * /usr/local/apache/htdocs/pqi/counter/hustat.pl
```

4.1 Backup der Mysql-Datenbank

```
mysqldump institut >institutDB.txt -uroot -hpciwww -ppasswort  
(in backupDB werden zudem auch die Datensätze bilder und mysql so gesichert)
```

Restore: (!! erst teilweise getestet !!)

```
mysql -uroot -hpciwww -p institut <backupfile
```

4.2 Alternatives Backup der Mysql-Datenbank

```
cd /;tar zcf var_mysql.tar.gz var/mysql
```

erfolgreich getestetes Restore:

```
cd /;tar zxf var_mysql.tar.gz
```

4.3 Mysql-Daten von Hand anschauen

Hat eigentlich nichts mit dem Backup zu tun, die anzugebenden Optionen sind aber ähnlich.

Tabellendefinition anschauen:

```
mysqlshow -uroot -hpciwww -ppasswort institut tbl_pers %
```

Interaktiv mit der Datenbank arbeiten:

```
mysql -uroot -hpciwww -p
```

```
mysql> use institut
```

Die andere Variante ist der Aufruf von “kmysqladmin”.

5 Statistik mit Counter

Das Java-Applet das vorher schon zusammen mit dem Odometer verwendet wurde, wird vorläufig weiter verwendet. Aber statt dem Odometer wird jetzt alles mit der Datenbank gemacht.

Die Counter sind in der Tabelle `tbl_count` zu finden. Der eigentliche Zähler ist in `count_n`, der Name der Seite in `count_page`, das Erstellungsdatum des Counters in `count_start`, und die vom Java-Applet erwartete Reihenfolge wird in `count_ord` definiert.

In `tbl_statc` werden täglich die aktuellen Counterwerte gesammelt (mit `cron`).

Ab und zu (ca. alle Jahre) sollten dort einige alte Werte gelöscht werden. (Je ein Wert an jedem Monatsanfang und Monatsende behalten). Dazu gibt es das Perl-Programm

“`/home/pfister/rolf/pci/counter/statistik_reset.pl`”. Es gibt dort auch ein “Liesmich”.

Eine Liste aller Werte von `tbl_statc` kann man bequem mit

“`http://pciwww.unizh.ch/pci/counter/getcount.epl?id=0`” erhalten. (Im Browser mit “save as” als Source speichern ergibt eine Ascii-Datei).

Beispieleiten

“`http://pciwww.unizh.ch/pci/pfister/html.epl`”

“`http://pciwww.unizh.ch/pci/counter/statistic.epl`”

6 Perl und EPL

Perl ist sehr ähnlich zu c. Einer der wichtigsten Unterschiede ist, dass in Perl alle einfachen Variablen mit \$ beginnen (Listen mit @ oder %).

Eine Einführung in Perl ist unter "<http://www.teamone.de/selfhtml/tgc.htm>" zu finden.

Embperl wird in Dateien mit der Endung ".epl" eingefügt. Wichtigste Einschränkung gegenüber normalem Perl: es können keine Dateien verwendet werden (weder lesen noch schreiben).

Einfaches Beispiel:

```
<HTML><HEAD><TITLE>titel</TITLE></HEAD>
[!
  use DBI;
  use CGI qw(:standard);
!]
<BODY>
[- $text="hallo"; -]
  Hier wird Text von einer Perl-Variablen eingefügt: [+ "$text" +]<BR>
[$if($text eq "Hallo")$]
  Bedingung ist wahr.
[$else$]
  Bedingung ist falsch.
[$endif$]
[# dies ist ein Kommentar #]
</BODY>
</HTML>
```

Der Perl-Code wird auf dem Server ausgeführt und erst dann die resultierende HTML-Datei verschickt.

Eine gute Dokumentation zu Embperl ist hier zu finden:

"<http://perl.apache.org/embperl/Embperl.pod.cont.html>"

6.1 Datenbankabfragen

Syntax der wichtigsten SQL-Befehle

```
SELECT a,b FROM tbl_x WHERE .. ORDER BY .. GROUP BY ..
INSERT INTO tbl_x (a,b) VALUES ('v1','v2');
UPDATE tbl_x SET a='v1',b='v2' WHERE .. OR .. ;
DELETE FROM tbl_x WHERE .. AND .. ;
```

Typische Beispiele

```
abfrage("SELECT news_titel,news_text
        FROM tbl_news
        ORDER BY news_datum DESC");
for($j=0;$j<$jmax;$j++) {.. $tit=$feld->[$j][0]; $txt=$feld->[$j][1] ..}

abfragh("SELECT news_titel,news_text,
        DATE_FORMAT(news_datum,'%e.%m.%Y') ndatum
        FROM tbl_news
        WHERE news_id=$newsid");
$tit=$hfeld->{news_titel}; $txt=$hfeld->{news_text}; $dat=$hfeld->{ndatum};

$tit=simpleabfrage("SELECT news_titel FROM tbl_news WHERE news_id=7");

go("INSERT INTO tbl_news (news_titel,news_text,news_datum)
    VALUES ('hallo','kleiner Test',SYSDATE());");
$idmax=simpleabfrage("SELECT news_id FROM tbl_news ORDER BY news_id DESC");
go("DELETE FROM tbl_news WHERE news_titel='hallo';");
```


6.2 Ganze Tabelle abfragen

Es gibt im Embperl die speziellen Variablen \$row und \$col, mit denen sich eine Tabelle erzeugen lässt ohne schon zu wissen wie gross sie wird.

Beispiel1:

```
[- abfrage("SELECT * FROM tbl_news ORDER BY news_datum DESC"); -]
<TABLE BORDER=0>
  <TR><TD>[+ $feld->[$row][$col] +]</TD></TR>
</TABLE>
```

Hier noch eine Variante mit Anpassung der (bekannten) Tabellenbreite und setzen eines Links.

Beispiel2:

```
[-
  abfrage("SELECT DATE_FORMAT(news_datum, '%a %e.%b.%Y') datumd,
            CONCAT('\<A HREF=\"news.ep1?news_id=', news_id, '\>',
                  news_titel, '\</A>')
            FROM tbl_news ORDER BY news_datum DESC");
-]
<TABLE BORDER=0 width=100%>
  <TR>
    <TD width=20%>[+ $feld->[$row][0] +]</TD><TD>[+ $feld->[$row][1] +]</TD>
  </TR>
</TABLE>
```

Nochmal das gleiche mit Verwendung einer perl-Schleife (statt \$row und \$col).

Beispiel3:

```
[-
  abfrage("SELECT DATE_FORMAT(news_datum, '%a %e.%b.%Y') datumd,
            news_id, news_titel
            FROM tbl_news ORDER BY news_datum DESC");
  $j=0;
-]
<TABLE BORDER=0 width=100%>
  [ $while($j<$jmax)$ ]
  <TR>
    <TD width=20%>[+ $feld->[$j][0] +]</TD>
    <TD><A HREF="news.ep1?news_id=[+ $feld->[$j][1] +]">
      [+ $feld->[$j][2] +]</A></TD>
  </TR>
  [- $j++; -]
  [ $endwhile$ ]
</TABLE>
```

6.3 Eingabe von Textfeldern

Wenn man Texte mit beliebigen Zeichen eingeben können soll, dann muss man TEXTAREA verwenden. Vor dem Speichern in der Datenbank muss man zuerst alle \ durch \\ und dann alle ' durch \' ersetzen.

Beispiel aus publi.epl:

```
<table>
  <FORM METHOD="POST" ACTION="publi.epl">
  <INPUT NAME=aktion VALUE=update TYPE=HIDDEN>
  <INPUT NAME=pub_id VALUE=[+ "$pub_id" +] TYPE=HIDDEN>
  <tr>
    <td>Titel</td>
    <td><TEXTAREA NAME="pub_titel" ROWS=2 COLS=60>[+ "$feld1->[0][0]"
      +]</TEXTAREA></td>
  </tr><tr>
...
my $titel=$fdat{pub_titel};
$titel =~ s/\\/\\\\/g; $titel =~ s/\'/\\\'/g;
go("UPDATE tbl_pub
    SET pub_titel='$titel',
...

```

6.4 Bilder in Datenbank speichern und benutzen

Bild in Datenbank speichern

Damit man den Inhalt einer lokalen Datei (z.B. GIF-Bild) mit einem Formular an den Server schicken kann, muss ein cgi-bin-Programm aufgerufen werden.

Beispiel des aufrufenden Programms: **user.epl**

Entsprechender Teil daraus:

```
<FORM ACTION="/cgi-bin/upload.pl" METHOD="POST" ENCTYPE="multipart/form-data">
[-
  abfragh("SELECT CONCAT(pers_name,pers_id,'.gif') bild
          FROM tbl_pers
          WHERE pers_id=$pers_id");
  $bild=lc("pic/$hfeld->{bild}");
-]
<INPUT NAME="ziel" TYPE="HIDDEN" VALUE="[+ $bild +]">
<table>
<tr><td>Datei:</td><td><INPUT TYPE=file NAME="datei" SIZE=20></td>
</tr><tr>
<td></td><td><INPUT TYPE="submit" VALUE="absenden"></td>
</tr>
</table>
</FORM>
```

Beispiel des aufgerufenen Programms: **upload.pl**

Auszug daraus:

```
... # Die Zeilen vor dem eigentlichen Bildinhalt schon eingelesen.
@x = <STDIN>;
$bild="";
$neub="";
for(@x) {$bild=$neub; $neub .= $_;}
# in $bild stehen jetzt die binären Daten, aber mit zusätzlich 0D0A am Schluss
$n=length($bild);
$bild=substr($bild,0,n-2); #die beiden zusätzlichen Bytes entfernen

#1. Alle \ durch \\ ersetzen
#2. Alle " durch \" ersetzen
$bild =~ s/\\/\\/\\/\\/g;
$bild =~ s/"/\\/\\/\\/g;

go("INSERT INTO tbl_bild (bild_gif) VALUES(\"$bild\")");
```

Bild von Datenbank lesen

Um ein Bild aus der Datenbank darzustellen wird ein cgi-bin-Programm gebraucht. Das heisst es muss ein Programm (z.B. Perl-Programm) im Verzeichnis "usr/local/apache/cgi-bin/" gespeichert werden.

Auszug aus dem Beispiel **testbild2.pl**:

```
#!/usr/bin/perl
use DBI;
use Mysql;
#use CGI qw(:standart);
print "Content-Type: image/gif\n\n";
...
abfragh("SELECT bild_gif FROM tbl_bild ORDER BY bild_id DESC");
$bild=$hfeld->{bild_gif};
$dbh->disconnect;
print $bild;
exit(0);
```

Um das Bild in einer html-Datei einzubinden wird dann folgende Zeile gebraucht:

Entsprechendes Beispiel: "http://pciwww.unizh.ch/test/testbild1.html"

7 Perl-Subroutinen

Um das Rad nicht immer wieder neu zu erfinden haben wir einige Perl-Subroutinen programmiert. Der allgemeine Aufbau einer Subroutine in Perl ist:

```
sub name
{
  my ($parameter1,$parameter2) = @_ ;
  my $y=0;
  return $y;
}
```

Die folgenden Subroutinen sind in /home/pfister/subroutinen.epl zu finden.

abfrage(“Befehl”)

Dies macht eine MySQL-Abfrage und speichert die Ergebnistabelle in \$feld, wobei die Anzahl Zeilen in \$jmax zu finden ist. Zugegriffen wird mit \$feld->[\$j][\$i], wobei j die Zeile (0 bis jmax-1) und i die Spalte ist.

Falls kein Fehler aufgetreten ist, hat \$ok einen Wert grösser als 0.

abfragh(“Befehl”)

Hier wird das Ergebnis in \$hfeld gespeichert. Zugegriffen wird mit \$hfeld->{name}, wobei “name” der Spaltenname (oder Alias) ist.

Dies ist aber nur brauchbar wenn das Ergebnis nur eine einzige Zeile ist.

abfragh2(“Befehl”)

Hier wird das Ergebnis in \$hfeld2 gespeichert. Zugegriffen wird mit \$hfeld2[\$j]->{name}.

\$ergebnis=simpleabfrage(“Befehl”)

Hier wird das Ergebnis direkt zurückgegeben.

Dies ist aber nur brauchbar wenn das Ergebnis nur ein einzelner Wert ist.

go(“Befehl”)

Hier wird ein MySQL-Befehl abgesetzt, der keine Rückgabe liefert (also ein UPDATE, DELETE oder INSERT Befehl).

datum("Datum")

Damit wird ein Datum der Form "31.12.2001" in die Form "2001-12-31" umgewandelt, wobei sichergestellt wird dass ein gültiges Datum entsteht.

Sollte vor dem Speichern eines Datums in der Datenbank verwendet werden.

Um ein Datum in der Form "31.12.2001" bei einer SELECT-Abfrage zu erhalten, sollte

```
DATE_FORMAT(datum, '%e.%m.%Y') verwendet werden.
```

profdr(pers_id)

Sucht die Titel einer Person aus der Datenbank und gibt entsprechend "Prof. ", "Dr. ", "Prof. Dr. " oder "" zurück.

checkgast(count_id)

Überprüft ob ein Gast die Seite mit dem entsprechenden Counter schon besucht hat, und macht gegebenenfalls einen Eintrag in tbl_gast. (Wird von der Subroutine counter benutzt).

counter("Countername")

Liest den aktuellen Zähler und erhöht ihn gegebenenfalls. In \$cn steht dann der aktuelle Zählerstand und in \$cs das Startdatum des Zählers.

Typische Anwendung:

```
<A HREF=" ../pci/counter/statistic.epl">Statistik:</A>
```

```
[- counter("Publikationen"); -]
```

```
diese Seite wurde bisher <B>[+ $cn +]</B> mal aufgerufen (seit [+ $cs +]).
```

babel(text1,sprache1,text2,sprache2)

Damit wird ein Text in der richtigen Sprache ausgegeben.

Die Zielsprache ist dabei \$tolang oder, wenn nicht definiert, der vom Benutzer (Identifikation über IP-Adresse) eingestellten Wert der Datenbank (opt_lang aus Tabelle tbl_opt).

Wenn kein zweiter Text angegeben wird, wird in der Datenbank (tbl_txt) nach der richtigen Übersetzung gesucht. Typische Anwendung:

```
[+ babel("hallo Welt", "de", "hello world", "en") +]
```

passcheck(passwd, eingabe)

Wird im Admin-Bereich benötigt um zu vergleichen ob eingabe dem verschlüsselt gespeicherten passwd entspricht. Bei Übereinstimmung wird 1 sonst 0 zurückgegeben.

Typische Anwendung:

```
[$if(passcheck($pass_passwd, $fdat{password}))$]
```

```
Willkommen ...
```

```
[$else$]
```

```
falsches Passwort  
[$endif$]
```

setpass(password)

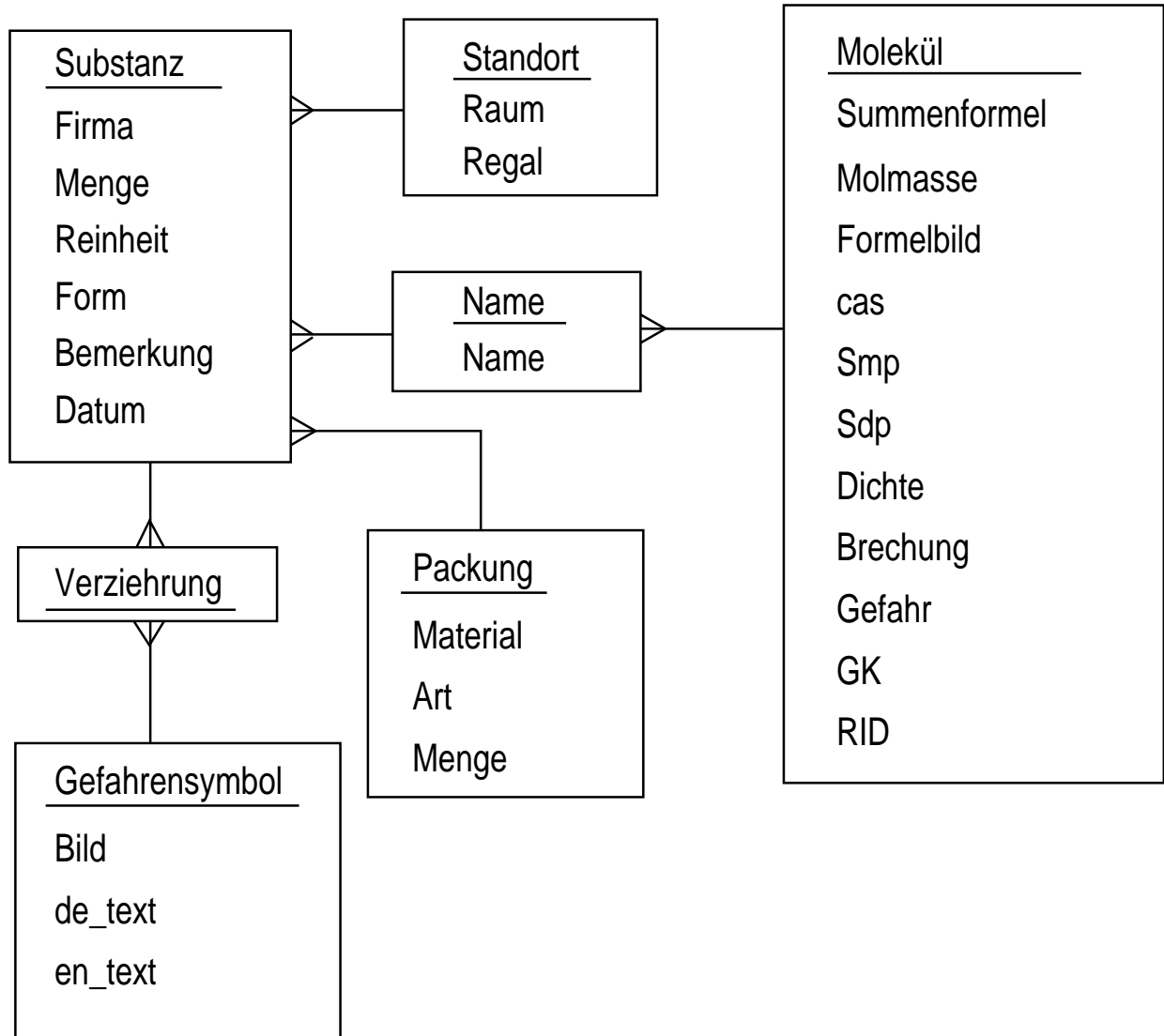
Der Rückgabewert liefert ein verschlüsseltes Passwort. Dabei werden Zufallszahlen verwendet so dass bei mehrmaligem Verschlüsseln des selben Passwortes jedesmal ein anderes verschlüsseltes rauskommt.

Typische Anwendung:

```
my $passwd=setpass($fdat{password});  
go("UPDATE tbl_pass SET pass_passwd='$passwd' WHERE pass_user='$pass_user'");
```

8 Chemikalienkartei

Chemikalien



Abfragen

Startpunkt für die Suchabfrage ist unter Links bei "nur lokal abrufbare Seiten" zu finden.

Aktualisierungen

Startpunkt für Neueinträge und Korrekturen ist unter Links bei "Passwort geschützte Seiten" zu finden.

Beim Einloggen unter "<http://pciwww.unizh.ch/admin>" und Verwendung von "Chemikalien DB" führt noch zur alten Variante von Richard.

9 Probleme und Fallstricke

Hier werden bereits gelöste Probleme behandelt.

- Bei Verwendung des eigentlich eleganten Schlaufenmechanismus im EPL mit \$row und \$col gibt es ein Problem bei mehr als 100. Beispiel zur Umgehung des Fehlers in admin/publiz.epl. Jetzt gibt es aber eine einfachere Behebung des Problems: \$maxrow und \$maxcol auf höheren Wert setzen, oder \$tabmode setzen.
(http://perl.apache.org/embperl/Embperl.pod.6.html#_tabmode)
- In epl-Dateien müssen bei Tabellen TH TR TD immer abgeschlossen werden. Ist etwas störend wenn man aus einer alten html-Datei ein epl macht.
- Wenn man in epl in einer while-Schleife vergisst den Zähler zu erhöhen kann der Server bis zum Absturz ausgebremst werden. Neustart von apache, oder neu booten ist dann angebracht. (Kommando als root: “/sbin/init.d/apache restart” oder “reboot”)
- Bei Textfeldern (TEXTAREA) dürfen ausserhalb der [++] Konstruktion keine Leerstellen oder Zeilenumbrüche verwendet werden. Diese werden sonst in den Text eingefügt. (Beispiel im Kapitel “Eingabe von Textfeldern”)
- Beim Speichern von Text in der Datenbank gibts Probleme wenn einfache Anführungszeichen (') oder Backslashes (\) vorkommen. Diese Zeichen müssen deshalb durch \' und \\ ersetzt werden: \$text =~ s/\\/\\\\/g; \$text =~ s/\'/\\\'/g; (in dieser Reihenfolge, sonst werden eingefügte Backslashes auch ersetzt!)
- Eine Seite wird fälschlicherweise aus dem Cache geholt, statt mit aktuellen Datenbankabfragen. Dies passiert wenn folgenden Zeile nach <head> vergessen wurde:
<meta http-equiv="expires" content="0">

10 Fehler und Verbesserungsvorschläge

10.1 im öffentlichen Bereich

- Unter Forschung sollten auch Links auf Bilder möglich sein. Bisher werden alle eingegebenen Bilder immer gezeigt. Man möchte aber manchmal ein Bild erst bei Bedarf anzeigen, oder durch ein Klick auf ein verkleinertes Bild das grosse Bild laden.
(Beispiel: in Forschung→Surface Science Labor wird beim Anklicken von “Fig.1” nichts gefunden)
- Sprachauswahl sollte noch eingehender getestet werden. Es sind noch nicht alle Texte zweisprachig gemacht, somit haben wir immer noch eine Mischung zwischen deutsch und englisch.
- In Vorlesungen→Seminarvorträge könnte man den nächsten aktuellen Vortrag hervorheben.
- Navigationsrahmen auf Wunsch ausschalten wäre hübsch. (Ist schon in tbl_opt vorbereitet)
- Stundenstatistik vom aktuellen Tag wäre auch noch hübsch.

10.2 im Administrationsbereich

- Wenn Änderungen in Vorlesungen gemacht wurden wird danach nicht mehr an die richtige Stelle zurückgesprungen. Dieser Fehler ist auch noch an anderen Stellen (im Admin-Bereich) vorhanden. Lässt sich vielleicht mit Javascript (BODY ONLOAD=”wechseln()”) beheben.
- Wenn Fehler in Cron vorkommen wird jeweils eine mail an root geschickt. Sollte periodisch abgefragt, oder besser umgeleitet werden.
- Verwendung von “Chemikalien DB” führt zur veralteten Variante.

11 Literatur und Internetadressen

Internetadressen

Mysql	www.mysql.com
Httpd	www.apache.org
EmbPerl	www.embperl.com
modperl	perl.apache.org/dist

Literaturhinweise

- **Perl**, Einführung Anwendung Referenz,
Farid Hajji,
Addison-Wesley 2000, 2.Auflage, ISBN 3-8273-1535-2
(hier wird neben Perl auch MySQL gut erklärt)
- **MySQL**, Einführung, Programmierung, Referenz,
Michael Kofler,
Addison-Wesley 2001, ISBN 3-8273-1762-2
(war beim Realisieren unserer Datenbank noch nicht vorhanden)