Echtzeitkartographie
Echtzeitendaten aufbereiten

Stefan Maihack Dipl. Ing. (FH)

14.09.2015

Echtzeitdaten verwenden – 2 Verfahren

Thema:

- Kennenlernen der 2 verbreitesten Verfahren zur Beschaffung von Drittanbieterdaten.

Inhalt:

➔ Daten aus CSV-Textdateien auftrennen und für die spätere Nutzung sinnvoll speichern.

➔ SQL-anstelle der bisher verwendeten Textdateien im Dateisystem (XML, CSV usw.) – Zur serverseitigen Datenspeicherung verwenden.

➔ SQL-Abfragen optimieren, um die gewünschten Daten schnell und einfach zu extrahieren.

➔ Die sichtbaren HTML-Daten von Webseiten parsen und die interessanten Teile daraus übernehmen. „Screen Scraping“.

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Daten downloaden

- Beispiel - Datenbasis ASR-Datenbank (Antenna Structure Registration Datenbase)
- Warum ASR-Datenbank:
  → Die Daten dürfen kostenlos genutzt werden
  → Die Daten sind leicht erhältlich und gut dokumentiert
  → Der Umfang der Daten ist beträchtlich
  → Breiten- und Längenangaben befinden sich bereits in der ASR-Datenbank
- Grobe Vorgehensweise:
  1. Herunterladen der Daten als Textdateien
  2. Parsen der Textdateien (Was ist wichtig? Was nicht?) und in eine sinnvolle lesbare Form bringen.
  3. CSV-Daten in Datenbanktabellen ablegen
  4. SELECT-Abfragen erstellen, um die Daten aus der Datenbank zu lesen und graphisch darzustellen.

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Daten aus Textdatei

- Herunterladen der Daten

  → http://wireless.fcc.gov/uls/data/complete/r_tower.zip (ca. 30 MB)

  → Entpacken des ZIP-Archivs

  → Die Dateien RA.dat und CO.dat in ein Arbeitsverzeichnis kopieren (z.B. ../data). Die anderen Dateien werden vorerst nicht benötigt
- Offizielle Dokumentation http://www.wireless.fcc.gov/cgi-bin/wtb-datadump.pl
- Beschreibung der Dateien:

  → RA.dat enthält neben eindeutigen Kennzahlen der einzelnen Bauten deren physikalische Daten wie Größe und Straßenangeben.
  → CA.dat enthält die Koordinaten der Bauten mit Breiten- und Längenangaben.

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Datei RA.dat

- Tabelle RA.dat – Verwendete Spalten

Spalte	Datenelement	Datentyp
3	Registrierungsnummer	Char(7)
4	Eindeutige SystemID	Int(9)
12	Baudatum	Mm/dd/yyyy
13	Abrissdatum	Mm/dd/yyyy
23	Bauort Straße	Varchar(80)
24	Bauort Stadt	Varchar(20)
25	Bauort Staat	Char(2)
26	Bauhöhe	Float(5.1)
27	Höhe des Standorts	Float(6.1)
28	Gesamthöhe über Grund	Float(6.1)
29	Gesamthöhe über mittleren Meeresspiegel	Float(6.1)
30	Bautyp	Char(6)

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Datei CO.dat

- Tabelle CO.dat – Verwendete Spalten

Spalte	Datenelement	Datentyp
3	Registrierungsnummer	Char(7)
4	Eindeutige SystemID	Int(9)
6	Breite: Grad	Int
7	Breite: Minuten	Int
8	Breite: Sekunden	Int
9	Breite: Richtung	Char(1)
11	Länge: Grad	Int
12	Länge: Minuten	Int
13	Länge: Sekunden	Int
14	Länge: Richtung	Char(1)

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Textdatei parsen

- Parsen der CSV-Daten

  → Daten der FCC-Datenbank in eine für uns sinnvolle Form bringen. http://www.php.net/fgets

- Beispielcode (Listing 5.1): Parsen einer mit dem Pipe-Zeichen (|) unterteilte Datei

```
<?php
// RA-Datendatei öffnen
$handle = @fopen("../data/RA.dat","r");

//Die ersten 50 USI-Nummern parsen und ausgeben
$i = 0;
if($handle) {
          while (!feof($handle)) {
                    $buffer = fgets($handle, 1024);
                    $row = explode("|", $buffer);
                    echo "USI#: ",$row[4]."<br> />\n";
                    if ($i == 50) break;
                    else $i++;
                    }
                    fclose($handle);
          }
?>
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Gründe für SQL DB

- Überführen der Daten in eine SQL-Struktur
- Warum:

  1. Falls die Daten in einem riesigen Array wären, so wären über 100MB Speicher notwendig. PHP-Scripte sind standardmäßig auf 8MB beschränkt.

  2. Falls drei Dateien gleichzeitig geöffnet werden müssen, dann muss zuvor sortiert werden. Falls das wieder mit PHP erfolgen würde, so käme man wieder an die Speichergrenzen.

  3. Würde man die Daten aus der Datei RA.dat mit der SQL-Anweisung INSERT einfügen und dann mit der UPDATE-Anweisung später beim Einlesen der Datei CO.dat die Leerfelder füllen, müsste man dazu die UPDATE-Funktion stark nutzen, die um ein vielfaches (10fache) langsamer als INSERT ist. (Importdauer > 8 Stunden)

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – SQL DB erstellen

```
###### Datenbank erstellen
DROP DATABASE IF EXISTS fcc;
SHOW WARNINGS;
CREATE DATABASE IF NOT EXISTS fcc DEFAULT CHARACTER
SET utf8 COLLATE utf8_general_ci;
SHOW WARNINGS;
USE fcc;
SHOW WARNINGS;


####### Tabellen erstellen
CREATE TABLE fcc_location (
          loc_id int(10) unsigned NOT NULL
auto_increment,
          unique_si_loc bigint(20) NOT NULL default
'0',
          lat_deg int(11) default '0',
          lat_min int(11) default '0',
          lat_sec float default '0',
          lat_dir char(1) default NULL,
          latitude double default '0',
          long_deg int(11) default '0',
          long_min int(11) default '0',
          long_sec float default '0',
          long_dir char(1) default NULL,
          longitude double default '0',
          PRIMARY KEY (loc_id),
          KEY unique_si (unique_si_loc)
) ENGINE=MyISAM;
```

```
CREATE TABLE fcc_structure (
  struc_id int(10) unsigned NOT NULL auto_increment,
  unique_si bigint(20) NOT NULL default '0',
  date_constr date default '0000-00-00',
  date_removed date default '0000-00-00',
  struc_adress varchar(80) default NULL,
  struc_city varchar(80) default NULL,
  struc_state char(2) default NULL,
  struc_height double default '0',
  struc_elevation double NOT NULL default '0',
  struc_ohag double NOT NULL default '0',
  struc_ohamsl double default '0',
  struc_type varchar(6) default NULL,
  PRIMARY KEY (struc_id),
  KEY unique_si (unique_si),
  KEY struc_state (struc_state)
) ENGINE=MyISAM;
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Anmeldedaten der DB

- PHP-Datei mit den Anmeldedaten erstellen

```
<?php
      $host            = "localhost";
      $user            = "root";
      $db_pass = "";
      $db_name = "fcc";
?>
```

- Mit der folgenden Anweisung wird diese Datei im eigentlichen PHP-Script eingelagert:

```
<?php
  include $_SERVER[´DOCUMENT_ROOT´]
  .´/../includes/db_credentials.php´;
?>
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Einlesen Teil 1/2

```
set_time_limit(0); // dies kann eine Weile dauern


// Verbindung zur Datenbank herstellen
require($_SERVER['DOCUMENT_ROOT'].'/db_credentials.php');
$conn = mysql_connect($host, $user, $db_pass);
if (!$conn) {die('keine Verbindung möglich: '.mysql_error());}

mysql_select_db($db_name, $conn);

//Datei mit dem physischen Standort öffnen
$handle = @fopen("../data/RA.dat","r");
if($handle) {
            while (!feof($handle)) {
                        $buffer = fgets($handle, 4096);
                        $row = explode("|",$buffer);
                        if ($row[3] > 0) {
                                    // Daten vor dem Einfügen ändern
                                    $row[12] = date("Y-m-d".strtotime($row[12]));
                                    $row[13] = date("Y-m-d".strtotime($row[13]));
                                    $row[23] = addslashes($row[23]);
                                    $row[24] = addslashes($row[24]);
                                    $row[30] = addslashes($row[30]);

                                    // Unsere Abfragen formulieren
                                    $query = "INSERT INTO fcc_structure
                                      (unique_si, date_constr,
                                      date_removed, struct_adress,
                                      struct_city, struct_state,
                                      struct_height, struct_elevation,
                                      struct_ohag, struct_ohamsl, struct_type)
                                      VALUES ('{$row[4]}', '{$row[12]}', '{$row[13]}',
                                                '{$row[23]}', '{$row[24]}', '{$row[25]}',
                                                            '{$row[26]}', '{$row[27]}', '{$row[28]}',
                                                            '{$row[29]}', '{$row[30]}')";

                                    // Abfrage ausführen
                                    $result = @mysql_query($query);
                                    if (!$result)
                        echo("FEHLER: Bauteninfo doppelt " . "#[$row[4]} <br>\n");
            }
}
fclose($handle);
}
echo "Bauten abgeschlossen. <br>\n";
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Einlesen Teil 2/2

```
// Datei mit Standorten öffnen
$handle = @fopen("../data/CO.dat","r");
if ($handle) {
                while (!feof($handle)) {
                                $buffer = fgets($handle, 4096);
                                $row = explode("|",$buffer);
                                if ($row[3] > 0) {
                                                if ($row[9] == "S") $sign = -1;
                                                else $sign = 1;
                                                $dec_lat = $sign*($row[6]+$row[7] / 60 + $row[8] / 3600);
                                                if ($row[14] == "W") $sign = -1;
                                                else
                                                                $sign = 1;
                                                $dec_long = $sign * ($row[11] + $row[12] / 60 + $row[13] / 3600);
                                                $query = "INSERT INTO fcc_location
                                                (unique_si_loc, lat_dir, latitude, longitude)
                                                VALUES ('{$row[4]}', '{row[6]}',
                                                '{$row[7]}', '{$row[8]}', '{$row[9]}',
                                                '$dec_lat', '{$row[11]}', '{$row[12]}',
                                                '{$row[13]}', '{$row[14]}', '$dec_long')";
                                                $result = @mysql_query($query);
                                                if (!$result) {
                                                                // Neuere Daten später in Datei UPDATE verwenden
                                                                $query =
                                                                "UPDATE fcc_location SET lat_deg='{$row[6]}',
                                                                lat_min='{$row[7]}', lat_deg='{$row[8]}',
                                                                lat_dir='{$row[9]}', latitude='$dec_lat',
                                                                long_deg='{$row[11]}', long_min='{$row[12]}',
                                                                long_sec='{$row[13]}', long_dir='{$row[14]}',
                                                                longitude='$dec_long,
                                                                WHERE unique_si_loc = '{$row[4]}'";
                                                                $result = @mysql_query($querey);
                                                                if (!$result)
                                                                                echo "Standort konnte nicht importiert " . "werden. #{$row[4]} <br>\n";
                                                                else
                                                                                echo "Bautenstandort aktualisiert." . "#{$row[4]} <br>\n";
                                                }
                                }
                }
                fclose($handle);
}
echo "Standorte abgeschlossen.<br>\n";
?>
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Nutzen der DB

- Die Daten aus der Datenbank können auf dreierlei Weise verwendet werden

  ➔ Verwenden von PHP zur Abfrage der einzelnen Tabellen und verbinden der Ergebnisse basierend auf dem Feld Unique STRUCTUTE ID in einem Array.

  ➔ Mehrere Tabellen mit einer SELECT-Abfrage gleichzeitig abfragen und SQL die Reorganisation der Daten überlassen.

  ➔ Falls die SQL-Datenbank Views (Ansichten) unterstützt, dann erstellt man eine Ansicht (virtuelle Tabelle) und wählt mit PHP die Daten aus dieser Ansicht aus.

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Reorg mit PHP

```
<?php
// Verbindung zur Datenbank herstellen
require($_SERVER['DOCUMENT_ROOT'].'/db_credentials.php');
$conn = mysql_connect($host, $user, $db_pass);
if("$conn) { die('Keine Verbindung möglich: ' . mysql_error();}

mysql_select_db($db_name, $conn);

//Arrays für temporäre Daten erzeugen
$hawaiian_towsers = array();
$usi_list = array();

//Liste der Bauten in Hawaii abfragen
$structures = mysql_query("SELECT * FROM fcc_structure WHERE struc_state='HI'");
for($i=0; $i<mysql_num_rows($structures); $i++) {
          $row = mysql_fetch_array($structures, MYSQL_ASSOC);
          $hawaiian_towsers[$row['unique_si']] = $row;
    $usi_list[] = $row['unique_si'];
}
unset($structures);

//Standort der obigen Bauten ermitteln
$locations = mysql_query("SELECT* FROM fcc_location WHERE unique_si_loc IN (".implode(",",$usi_list).")");
for($i=0; $i<mysql_num_rows($locations); $i++) {
          $row = mysql_fetch_array($locations,MYSQL_ASSOC);
          $hawaiian_towers[$row['unique_si_loc']] = array_merge($hawaiian_towers[$row['unique_si_loc']],$row);
}
unset($locations);

echo 'Benutzer Speicher: ');
echo memory_get_usage();
?>
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Reorg mit SQL

```
<?php
//Verbindung zur Datenbank erstellen
require($_SERVER['DOCUMENT_ROOT'] . '/db_credentials.php');
$conn = mysql_connect_($host, $user, $db_pass);
mysql_select_db($db_name, $conn);

//Multitabellen-Abfrage
$query = "SELECT * FROM fcc_structure, fcc_location WHERE struc_state='HI' AND unique_si = unique_si_loc";

$result = mysql_querey($query, $conn);
$count = 0;
?>

var markers = [
<?php while($row = mysql_fetch_assoc($result)): ?>
{
          'latitude': <?= $row['latitude'] ?>,
          'longitude': <?= $row['langitude'] ?>,
          'name': <?= addslashes($row['struc_address']) ?>,
},
<?
$count++;
endwhile;
?>
];
```

Echtzeitdaten verwenden

1. Verfahren - Nutzung einer heruntergeladenen Textdatei – Reorg mit Views

- Bei einer Ansicht (View) handelt es sich um eine temporäre Tabelle
- Erzeugen einer Ansicht „Erzeugen einer View aus 2 Tabellen“:

  CREATE VIEW fcc_towers AS SELECT * FROM fcc_structure, fcc_location
  WHERE unique_si = unique_si_loc
  ORDER BY struc_state, struc_type;

- Danach können einfach Abfragen gebildet werden:

  Beispiel: $query = „SELECT * FROM fcc_towers WHERE struc_state = ‚HI‘;

- Vorteile von Ansichten entstehen erst dann, wenn sie öfter verwendet werden.

Echtzeitdaten verwenden

Die Datenbank aktuell halten

- Daten können schon kurzer Zeit veraltet sein.
- Die täglichen Änderungen des FCC-Datenbank liegen unter „http://wireless.fcc.gov/cgi-bin/wtb-transactions.pl#tow
- Zur Aktualisierung der Daten in der Datenbank muss das Konto beim Webhoster folgendes bieten:

  ➔ Möglichkeit der Einplanung der periodischen Ausführung der Aktualisierungsperiode (einrichten von z-cron)
  ➔ Eine Shell-Scripting-Sprache, um den Aktualisierungsprozess schreiben zu können
  ➔ Ein Programm zur Übernahme der Transaktionsdateien in das neue Hilfsprogramm (einrichten von wget).

- Idee: Script schreiben, dass periodisch ausgeführt wird und eine ZIP-Datei mit Daten vom Daten-Provider lädt. Diese automatisch entpacken und wie lim Listing 5.3 die Daten davon in die Datenbank bringen.

Echtzeitdaten verwenden

Die Datenbank aktuell halten – Beispiel Shellscript

```
<?php
//Temporäre Dateien (z.B. aus der letzten Nacht) entfernen
exec(„rm r_tow_$day.zip CO.dat RA.dat“);

//Datum feststellen
$day = strtolower(date(„D“,strtotime(„yesterday“)));

//URL für die Übernahme der Daten mit wget festlegen
$url = „http://wireless.fcc.gov/uls/data/daily/r_tow_$day.zip“;

//ZIP-Datei herunterladen
exec(„/usr/bin/wget –q $url“);

//Relevante Teile der ZIP-Datei entpacken
exec(„/usr/bin/unzip –qq r_tow_$day.zip CO.dat RA.dat“);

//Daten mit dem Listing 5.3 in die Datenbank importieren. Der Pfad muss möglicherweise geändert werden
require_once(„../03/index.php“);

//Temporäre Dateien (vorbereitend) löschen
exec(„rm r_tow_$day.zip CO.dat RA.dat“);

?>
```

Echtzeitdaten verwenden

2. Verfahren - Nutzung Daten auf einer HTML-Seite – Screen Scraping

- Sichtbare HTML-Daten auf Webseiten parsen und extrahieren (Screen Scraping)
- Bei jeder Quelle muss unterschiedlich vorgegangen werden
- Grobe Vorgehensweise:

  1. HTML-Seiten mit „wget“ oder „cURL“ herunterladen und lokal abspeichern.

  2. Mit Hilfe von Schleifen und regulären Ausdrücken oder Umstellung von Strings die interessanten Datenquellen zusammenstellen.

  3. Textdaten in einer Datenbank ablegen

- ACHTUNG: Vorher nachfragen, ob diese Daten offiziell verwendet werden können und kostenfrei sind.

Echtzeitdaten verwenden

2. Verfahren - Nutzung Daten auf einer HTML-Seite – Screen Scraping

- Ein Beispiel:

  1. „wget“ herunterladen und installieren

  2. Folgende Beispielseite herunterladen http://www.urlaub-reise-tourismus.de/berlin-sehenswürdigkeiten-a-z.htm

  3. Muster erkennen

  … → Name der Sehenswürdigkeit

  52.507240
  13.390397

  Tip: In PHP gibt es eine praktische Funktion mit der Tags entfernt werden können „strip_tags()“

  Tip: Zur Umwandlung der Strings müssen nur noch jeweils das Vorzeichen für Breiten und Längengradangaben anhand N/S- bzw. E/W Kennung bestimmt werden.

- Siehe Beispiel-Script listing_5.5.php

Echtzeitdaten verwenden

2. Verfahren - Nutzung Daten auf einer HTML-Seite – Screen Scraping

Überlegungen zum Screen Scraping – Folgendes ist zu beachten

1. Für eine umfassende Fehlerprüfung sorgen, wenn Daten von einer dynamischen Quelle übernommen werden sollen.
2. Falls die Daten noch unsauber vorliegen, dann PHP-Erweiterung für reg. Ausdrücke verwenden.
3. Daten aus dem Internet können auch fehlerhaft sein.
4. Für statische Daten rentiert sich normalerweise kein Programm