

Aspekte der Weiterentwicklung

Dr.-Ing. Peter Korduan
Universität Rostock

Agrar- und Umweltwissenschaftliche Fakultät
Professur für Geodäsie und Geoinformatik

- Wunsch und To-Do Liste
- Query-Editor
- Entwicklung für den Massenmarkt
 - Native, Web, Hybrid for mobile
 - Single Map Clients, Generische Erstellung
- Migration
- Client – Server Kommunikation mit REST-API und MVC
- Scaffolding
- Augmented Reality

- Wir greifen weiterhin Ideen für künftige Weiterentwicklungen auf
- Hohe Priorität kommt in ToDo-Liste

Wunsch-Liste

Ideen für zukünftige Entwicklungen, Diskussion über geplante Entwicklungen. Bei allgemeinem Konsens werden diese Punkte in die ToDo-Liste aufgenommen. zur [ToDo-Liste](#)

Inhaltsverzeichnis [\[Verbergen\]](#)

- 1 Event Call back-Funktionen
- 2 CRUD Call back-Funktionen
- 3 Time-Felder mit Kalender
- 4 PRJ-Datei im Shape-Import
- 5 Filterung der ALB-Daten nach Gemarkungen u. Fluren
- 6 SVG-Export
- 7 Terminierung für user
- 8 Login-Versuche beschränken
- 9 Flurübersicht in Druckausschnittwahl als Option
- 10 ZIP-Archiv für Rechercheergebnis
- 11 Rechercheergebnis im neuen Fenster
- 12 Routing
- 13 Suchergebnis ALK-Adresssuche
- 14 Termine mit Benachrichtigung
- 15 Daten mit Flurstücksbezug
- 16 Adressrecherche ohne Flurstückslayer
- 17 Abrechen-Knopf bei der Datenerfassung im GLE
- 18 Verbesserung Geometrieeditor
- 19 Weiterentwicklung Nachweisrecherche/anzeige
- 20 Mitrutschende Menüleiste
- 21 Aliasnamen für Layer

- Aufnahme von Vorschlägen aus Wunschliste
- Konsens auf Anwendertreffen
- Abarbeitung erfolgt kontinuierlich
- Weiterentwicklung im Rahmen von Pflegeverträgen
- Spezielle Funktionen werden im Rahmen von individuellen Projekten entwickelt

ToDo-Liste

In der ToDo-Liste werden alle Projekte aufgelistet, die aus der WebGIS-Initiative M-V heraus als allgemein gewünschte Entwicklung beschlossen wurden. In dieser Liste werden außerdem alle Projekte aufgelistet, die auf Grund des Wunsches einzelner kwvmap-Nutzer entwickelt werden sollen. Der Status informiert über den Projektstand.

Die Diskussion über die genaue Beschreibung der Funktionalität findet bei den einzelnen Schlagworten auf der Diskussionsseite statt.

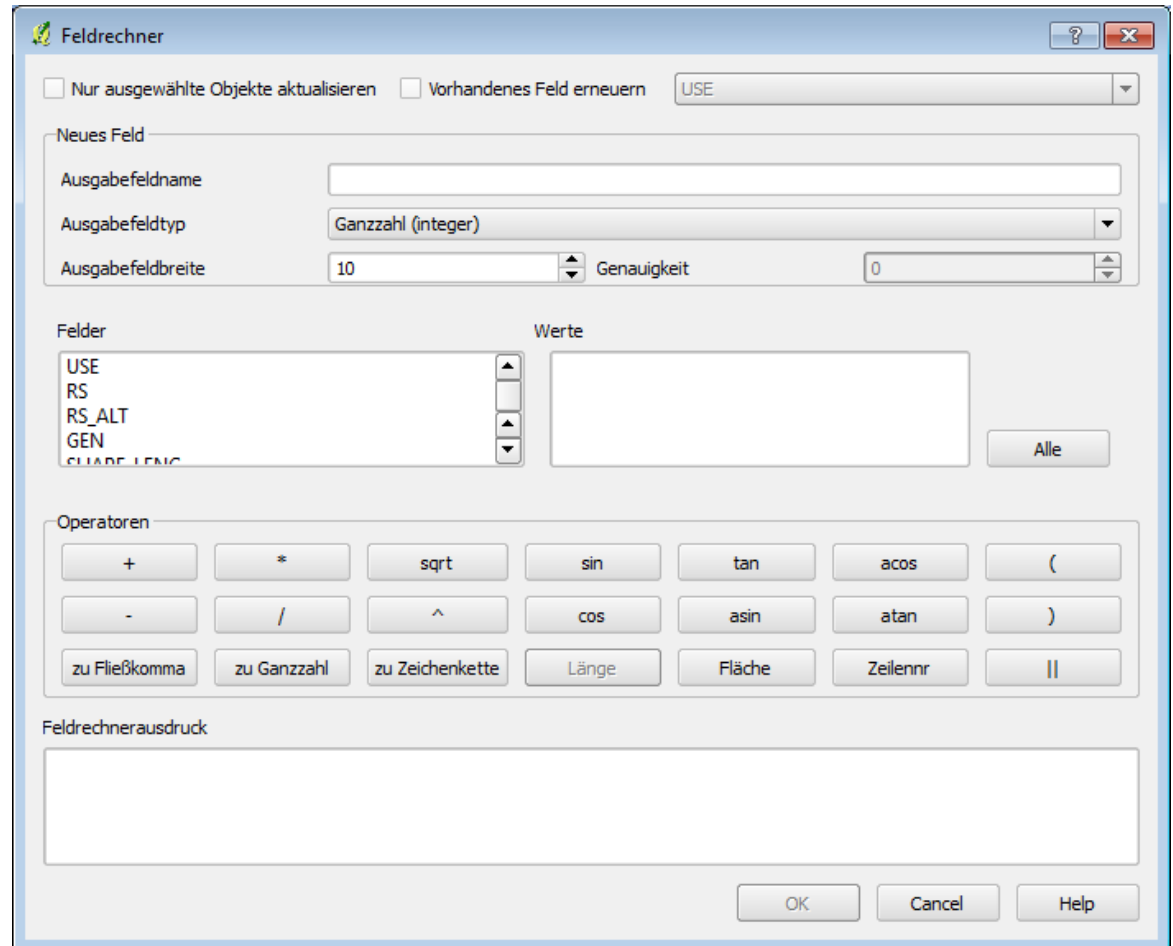
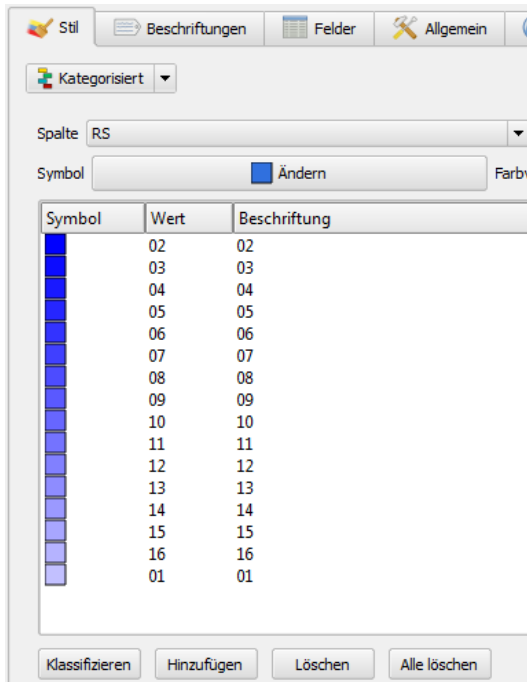
2012

Funktionalität	Status	Bemerkung
Erweiterung Nachweisverwaltung	abgeschlossen	
Tausender-Trennzeichen bei Zahlenanzeige	abgeschlossen	
Flächenberechnung über Infopolygon	abgeschlossen	
ALWAYS_DRAW rollenbezogen	abgeschlossen	
Polyline-Editor mit Geometrieübernahme	abgeschlossen	
Polyline-Editor mit Streckenmessung	abgeschlossen	
alb_fortfuehrung mit Dateinamen	in Entwicklung	
Weiterentwicklung Mobile Version	in Entwicklung	
Adresssuche erweitern	abgeschlossen	

2011

Funktionalität	Status	Bemerkung
ALKIS / APK-Funktionalitäten	in Entwicklung	
Mapserver 6.0	in Entwicklung	
UKO-Import - Dateiname	abgeschlossen	
Link als Vorschauattribut in Subform	abgeschlossen	
Jagdkataster	abgeschlossen	
BORIS	abgeschlossen	
Katastererneuerung durch Flurneuordnung	in Entwicklung	
WGS84 Koordinatenabfrage, -zoom und -anzeige	abgeschlossen	
Werkzeug "Wegstrecke messen"	abgeschlossen	
Speichern bei mehreren Datensätzen	abgeschlossen	

- Abfrage-Editor?
- Auto-Klassifizierung?
- Vorbild könnte QGIS sein



- Studie von Tobin Bradley aus Mecklenburg County, NC, USA
 - z.B. StreetView, Traffic und Google Earth nutzen nur ca. 2.2% - 3.5%
- Allan Glen, Denver
 - Vergleich Single-Topic Maps vs. allumfassendes Portal
 - Single-Topic maps 3 x mehr Zugriffe als Portal mit 55 Layern
 - 60% Anfragen kommen von Suchmaschinen
 - Mehr SEO Techniken anwendbar ⇒ mehr Treffer
 - Auto-complete fördert klarere Fragestellungen
 - Spitzenzeiten für Zugriffe sind lokale Veranstaltungen
 - Nutzer wollen da nur bestimmte Layer sehen (nicht alle)
 - Durchschnittliche Verweildauer 1:43 min
 - Einmal geladen gleich auf Marker, Info und Tschüss.
 - Nutzer interagieren mit Infofenstern
 - 1.7Mill clicks auf Marker 250T clicks auf Links im Infofenster
 - Nutzer ändern selten die Hintergrundkarten
 - 2% der Zeit für Wechsel
 - 0.5% für full-sceen Button

Weniger GIS,

mehr auto-complete
und
Suchoptimierung

durch Single-Topic Maps.

[Home](#) > [Map](#) > [Zoning](#)

Zoning

Find your zoning and learn about the zoning code.












- layouts/sql_dumps/mysql_update.sql und
- layouts/sql_dumps/postgis_update.sql
- Update per Hand
- Auswahl der Neuerungen an Hand der Kommentare mit Versionsnummern
- Keine Rückmigration möglich
- Keine Angabe der Version in der Datenbank

```
#-----  
# Änderung von 1.8.0 zu 1.9.0  
  
ALTER TABLE `rolle` ADD `coordtype` ENUM( 'dec', 'dms' ) NOT NULL DEFAULT 'dec' AFTER `epsg_code2`;  
  
CREATE TABLE `rolle_csv_attributes` (  
  `user_id` int(11) NOT NULL,  
  `stelle_id` int(11) NOT NULL,  
  `name` varchar(50) NOT NULL,  
  `attributes` text NOT NULL,  
  PRIMARY KEY (`user_id`,`stelle_id`,`name`)  
);  
  
ALTER TABLE `layer_attributes` ADD `decimal_length` INT( 11 ) NULL AFTER `length`;
```


- Änderungen in einzelnen Dateien
- Datumsstempel im Namen der Änderungsdateien
- Definition der Rückmigrationen

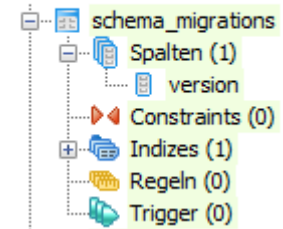
```
<?php
include('databaserelation.php');

class CreateFleets extends Databaserelation {
    function up() {
        $sql = "CREATE TABLE `rolle_csv_attributes` (
            `user_id` int(11) NOT NULL,
            `stelle_id` int(11) NOT NULL,
            `name` varchar(50) NOT NULL,
            `attributes` text NOT NULL,
            PRIMARY KEY (`user_id`,`stelle_id`,`name`);";
        $database->execSQL($sql);
    }
    function down() {
        $sql = "DROP TABLE `rolle_csv_attributes`;";
        $database->execSQL($sql);
    }
}
```

-  20091126123205_create_users.rb
-  20091126123206_create_roles.rb
-  20091126162614_create_fleets.rb
-  20091126200718_create_boxes.rb
-  20091127104750_add_description_to_boxes.rb
-  20091202092428_create_waypoints.rb
-  20091202092822_add_bwkt_to_waypoints.rb



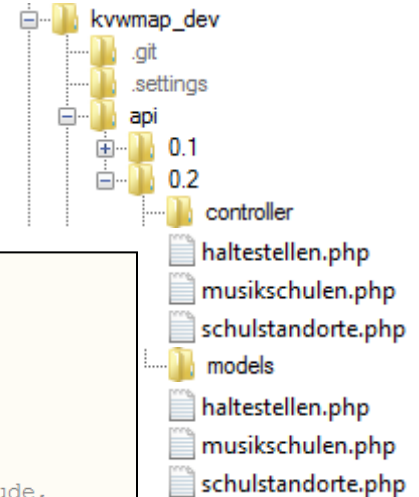
- Speicherung des aktuellen Standes der Migration in Datenbank
- Script zur Abarbeitung der Migration
 - db:migrate Update bis zur neuesten Version
 - db:migration VERSION="x" Update, bzw. Downgrade bis zu einer Version x



```
139.30.111.16 - PuTTY
fgs@eiger:~/fgs/apps/kvwmap_dev/layouts/sql_dumps> kvwmap db:migrate VERSION='20091202092428'
```

- Migration für auch für Quell-Code oder Daten?

- Trennung von
 - Modellen (Datenbanktabellen und Beziehungen)
 - Views (Darstellungsanweisungen)
 - Controller (Funktionen)



```

<?php
include('databaserelation.php');

class Haltestelle extends Databaserelation {

    function findByRadius($lat, $lng, $radius) {
        $radius_in_meter = $radius * 1000;
        $sql = "SELECT oid, standort, buslinien, Y(the_geom) AS latitude, X(the_geom) AS longitude,
        ST_distance(transform(GeometryFromText('POINT('.$lng.' ".$lat.'
        ')', 4326), 25833), transform(the_geom, 25833)) AS distance FROM ".$this->dbschema.".bushaltestellen
        WHERE ST_distance(transform(GeometryFromText('POINT('.$lng.' ".$lat.'
        ')', 4326), 25833), transform(the_geom, 25833)) < ".$radius_in_meter." ORDER BY distance LIMIT 10";
        $data = $this->findBySQL($sql);
        return $data;
    } # end of function findByRadius
    
```

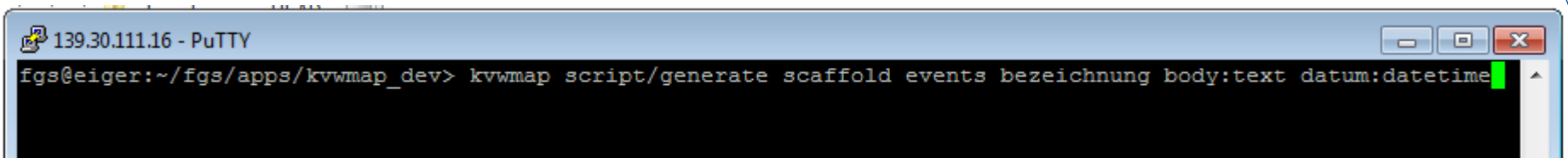
← Model

```

<?php
include('models/haltestelle.php');

switch($_REQUEST['request']) {
    case 'findByRadius' : {
        $haltestelle = new Haltestelle();
        $haltestellen = $haltestelle->findByRadius($_REQUEST['lat'], $_REQUEST['lng'], $_REQUEST['radius']);
        Haltestelle::output($haltestellen, $_REQUEST['radius'], $_REQUEST['format']);
    } break;
    }
    
```

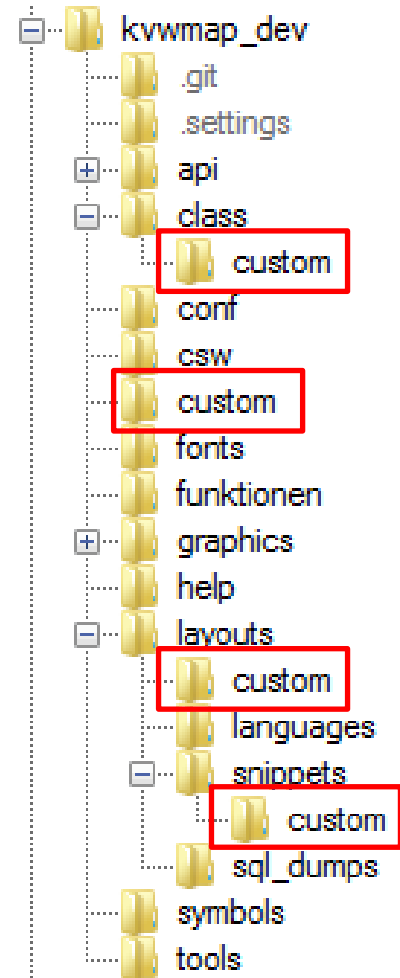
← Controller



```
139.30.111.16 - PuTTY
fgs@eiger:~/fgs/apps/kvwmap_dev> kvwmap script/generate scaffold events bezeichnung body:text datum:datetime
```

- Erzeugt:
 - Leere Tabelle in PostgreSQL
 - Layerdefinition
 - Single-Topic Map
- oder
 - Migrationsdatei
 - Model
 - Controller
 - Views
 - ...
- Je nach dem ob es Custom oder Core ist.

- Dinge für den eigenen Gebrauch (Custom)?
 - Cases in index.php
 - Classes
 - Graphics
 - Layouts
 - Snippets
- Eigene Fachschalen?
 - Gehören Dateien in Custom zu den Fachschalen?
- Klare Definition was zur Fachschale gehört
- Plug-In Style zum Austausch von Fachschalen
- Relativiert sich durch Single Topic Maps



- Müsste eigentlich eine Methode der Class Gewässer sein.
- Aufteilung der Anwendungsfälle in Methoden von Klassen
- Action in go Variable aufteilen in Controller und Action
- go=migrationGewaesser
- ⇒ c=gewaesser&a=migration
- eval("include(".\$c.".php");
- eval("\$g=new".\$c."()");
- eval("\$g->".\$a."()");

```

<?php
$goNotExecutedInIncludeCases = false;
switch($GUI->go) {
    case 'migrationGewaesser' : {
        if ($GUI->Stelle->isFunctionAllowed($go)) {
            include (CLASSPATH.'custom/wasserverband.php');
            $gewaesser = new gewaesser($GUI->pgdatabase);
            $gewaesser->truncateSpatial();
            $gewaesser->loadSpatial(
                '/home/fgs/wasserverband-kroepelin/gew-kroepelin_utm_ost');
            $gewaesser->loadSpatial(
                '/home/fgs/wasserverband-kroepelin/gew-kroepelin_utm_sued');
            $gewaesser->loadSpatial(
                '/home/fgs/wasserverband-kroepelin/gew-kroepelin_utm_west');
            $gewaesser->loadThematic();
            $GUI->gewaesser=$gewaesser;
            $GUI->main='custom/migrationGewaesser.php';
            $GUI->output();
        }
        else {
            # Benutzer ist nicht berechtigt zum Ausführen des Anwendungsfalles
            $GUI->Fehlermeldung=$GUI->TaskChangeWarning;
            $GUI->rollenwahl($Stelle_ID);
            $GUI->output();
        }
    } break;
    default : {
        $goNotExecutedInIncludeCases = true;
    }
}
?>

```

- Projektion von virtuellen Dingen in die Reale Welt
- Real ist das was man sieht
- Virtuell ist das was man eingeblendet bekommt
- Überlagerung in Bild der Kamera von Handys
- Verknüpfung mit Sachdaten



Quelle: <http://itunes.apple.com/de/app/wikitude-augmented-reality/id329731243?mt=8>



Vielen Dank für Ihre Aufmerksamkeit

Fragen?

www.kvwmap.de