

Kapitel 4 (4a)

PostGIS Administration, Import/Export von Daten, Betrachten von Daten

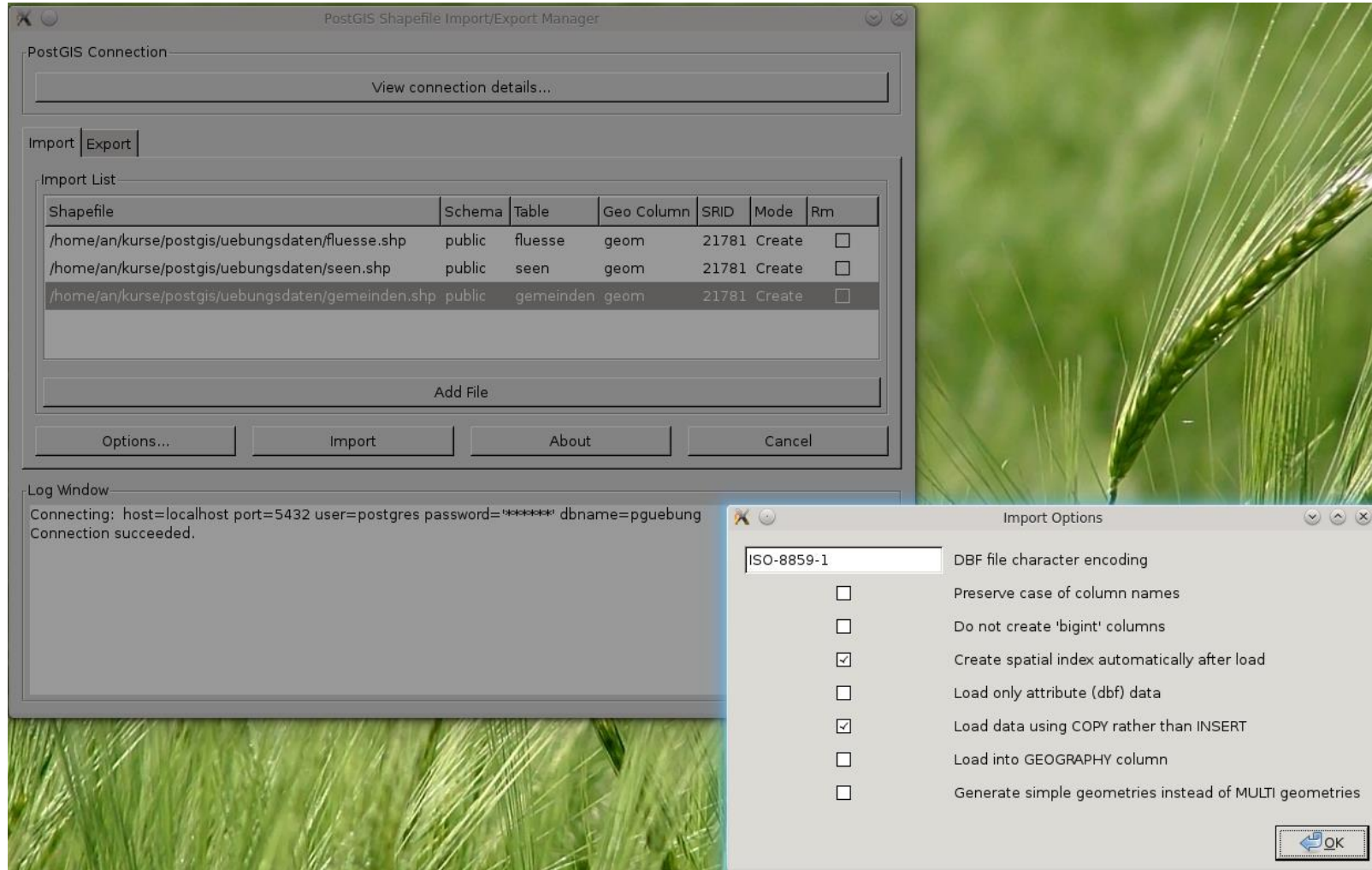
Stefan Keller

Dank an Dr. Andreas Neumann

Werkzeuge zum Laden von Geometrien

- ◆ **shp2pgsql / shp2pgsql-gui**
- ◆ **Laden von WKT/WKB Geometrien per CSV/SQL mit Type Cast**
- ◆ **Laden mit OGR**
- ◆ **QGIS DB-Manager / ArcCatalog**
- ◆ **GIS ETL-Tools (Extract / Transform / Load):**
 - GDAL / OGR
 - FME
 - speziell Interlis: InfoGrips-Tools; ili2db
- ◆ **ETL-Tools**
 - Talend (dual license)
 - SQL Server (kommerziell)
 - CSV Tools, dbt

shp2pgsql-gui



Laden per SQL/CSV-Datei

SQL-Dateien laden

psql:

```
% psql -d uster -f lwflaechen.sql
```

pgAdmin4:

Laden des SQL-Files im SQL-Editor und ausführen

Text-Dateien laden (CSV)

psql (COPY-Befehl):

```
% psql -d uster
```

```
% COPY av.gvz_nutzungscode  
FROM '/home/an/GVZ_Code.csv'  
DELIMITER AS ';' CSV HEADER;
```

Laden per OGR

Beispiel: Laden einer GeoPackage-Datei nach PostGIS:

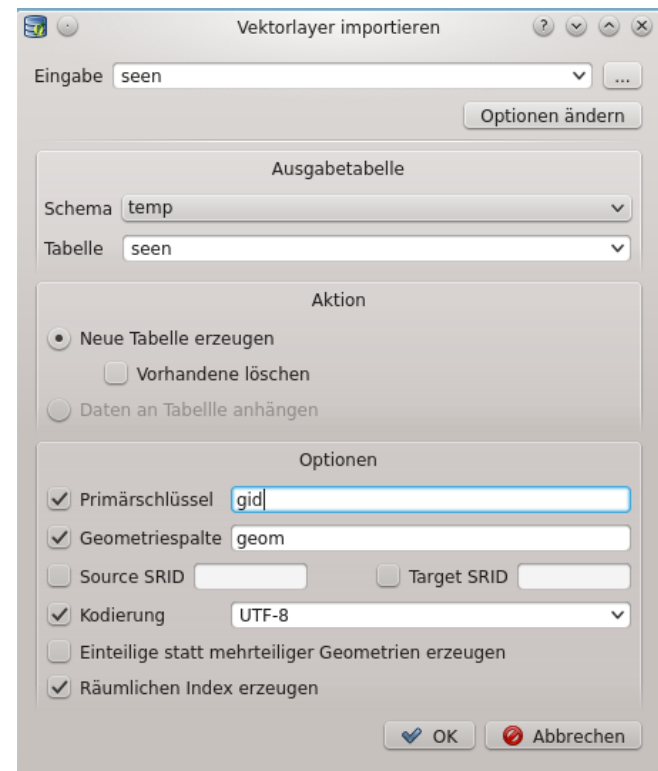
```
% ogr2ogr
  -f "PostgreSQL"

  PG:"host=DEIN_HOST
  user=DEIN_BENUTZER
  dbname=DEINE_DATENBANK
  password=DEIN_PASSWORT"
  -append
  -nln ZIELTABELLE
  -s_srs "EPSG:21781" -t_srs "EPSG:4326"
  QuellGeoPackage.gpkg
```

% zeigt an, dass das in einer Shell ist (DOS-Konsole)

Laden per QGIS DB-Manager

- ◆ Zuerst Datei in QGIS laden
- ◆ Danach DB-Manager starten (Menü Datenbank > DB-Manager)
- ◆ Im DB-Manager: Menü „Tabelle“ > „Layer/Datei importieren“
- ◆ Import-Optionen vornehmen



Beispiel ETL/FME

PostGIS Input Settings [?] [X]

Database Connection:

Host name:

Port:

Database:

User Name:

Password:

Constraints:

Table List: ...

Envelope:

☐ Use Search Envelope

Minimum X:

Minimum Y:

Width:

Height:

☒ Clip to Search Envelope

OK Cancel

Add Source Dataset [?] [X]

Source

Format: ...

Dataset: ...

Settings...

Coordinate System: ...

OK Cancel

Werkzeuge zum Anzeigen von Daten

Frage: Was sind das für Daten? Layer, Tabellen, Attribute, Datentyp, KRS, etc.

◆ QGIS

◆ pgAdmin4

◆ OGR → % ogrinfo fluesse.shp -al -so

◆ Geometry: Line String

–GESAMT1M_I: Integer (9.0)

–NAME: String (20.0)

–Feature Count: 49

```
Eingabeaufforderung
C:\Daten\daten_eigene_kurse_HSR-Kurs_PostGIS_2023-09\Uebungsdaten>ogrinfo fluesse.shp -al -so
INFO: Open of `fluesse.shp'
      using driver `ESRI Shapefile' successful.

Layer name: fluesse
Geometry: Line String
Feature Count: 49
Extent: (488131.394865, 85709.257813) - (831066.562500, 283450.062500)
Layer SRS WKT:
PROJCRS["CH1903 / LV03",
  BASEGEOGCRS["CH1903",
    DATUM["1903",
      EPOCH["1903-01-01T00:00:00",
        ID["EPSG",21781]]],
    PRIMARIES["Longitude", "Latitude"],
    UNIT["Meter", 1],
    AXIS["Longitude Easting", "Latitude Northing"],
    AUTHORITY["EPSG",21781]]],
  PROJECTION["Oblique_Spheroidal"],
  PARAMETER["Semi_Major_Axis", 6370185.523],
  PARAMETER["Semi_Minor_Axis", 6370185.523],
  PARAMETER["Longitude_of_Center_of_Mass", 0],
  PARAMETER["Latitude_of_Center_of_Mass", 0],
  PARAMETER["False_Easting", 0],
  PARAMETER["False_Northing", 0],
  PARAMETER["Scale_Factor", 1],
  AUTHORITY["EPSG",21781]]
Data axis to CRS axis mapping: 1,2
GESAMT1M_I: Integer (9.0)
NAME: String (20.0)
```


Werkzeuge zum Anzeigen von PostGIS-Daten

◆ OpenSource Desktop-GIS

- QGIS sowie GRASS

◆ Kommerzielle Desktop-GIS

- ArcGIS Pro, Geomedia, Autodesk Map3D, etc.

◆ Map-Server

- UMN Mapserver, Geoserver, QGIS Server

◆ Web-GIS-Applikationen

- Web-Framework GeoDjango (Python, selbst geschrieben)

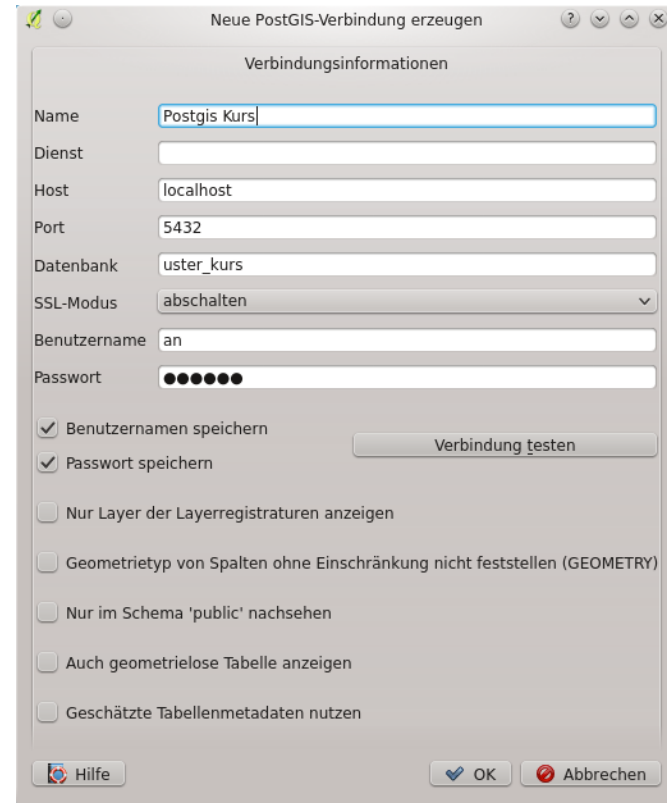
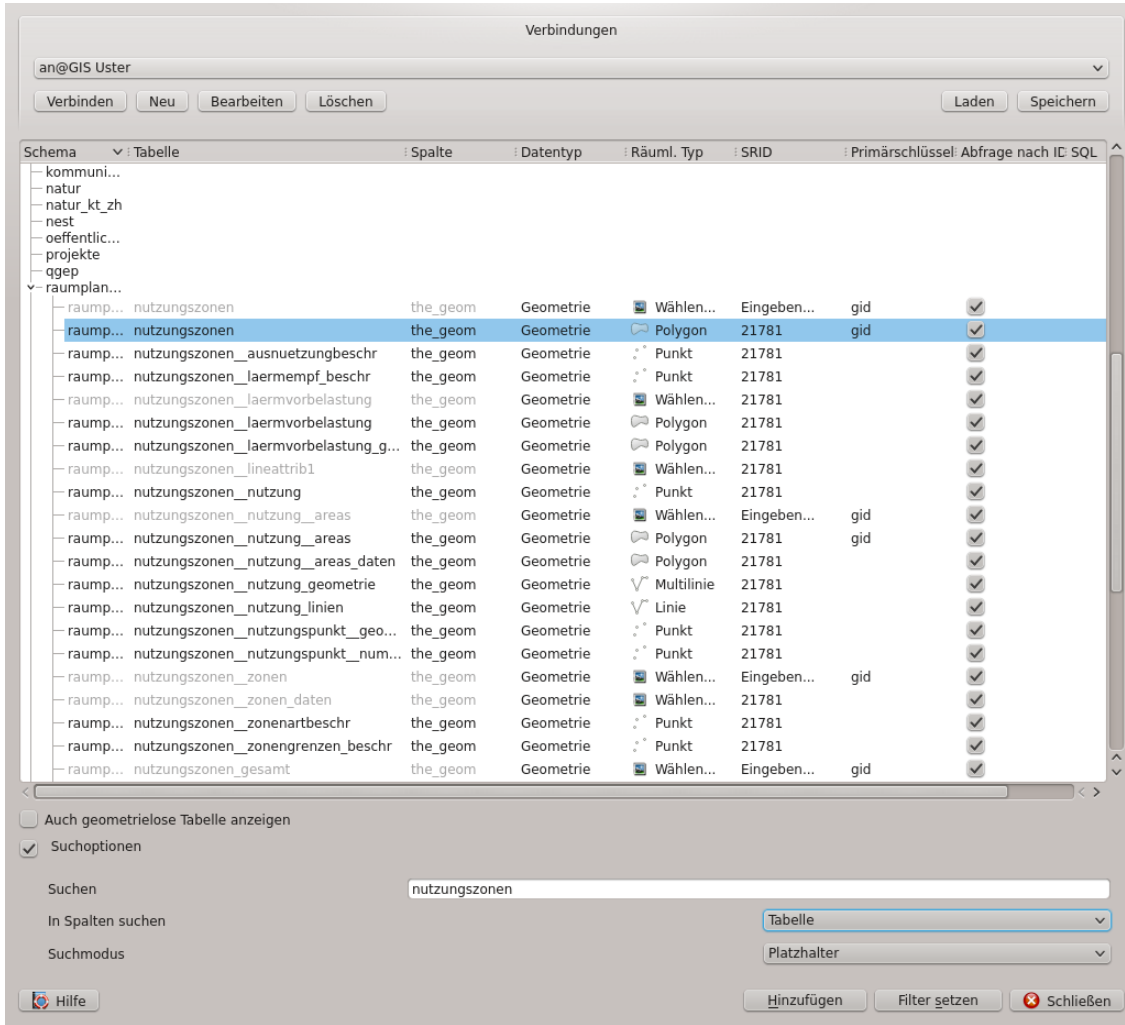
◆ Cloud-basierte GIS

- QGIS Cloud (QGIS Server), ArcGIS Online

◆ Geo-Datenbank mit Webserver (selbst geschrieben)

◆ Virtuelle Globen

PostGIS Lade-Dialog in QGIS



DB-Manager in QGIS

The screenshot shows the QGIS DB Manager interface. On the left, a tree view lists various database tables, with 'flurnamen' selected. The main panel on the right displays the 'flurnamen' table details, including general information, PostGIS metadata, and a list of fields.

flurnamen

Allgemeine Informationen

Relationsart: Ansicht
Besitzer: an
Kommentar: Flurnamen mit Polygonen und Zentroiden (ohne definitive Beschriftungspositionen)
Seiten: 0
Zeilen (geschätzt): 0
Zeilen (gezählt): 487
Rechte: select, insert, update, delete

PostGIS

Spalte: the_geom
Geometrie: POLYGON
Dimension: 2
Räuml. Bez.: CH1903 / LV03 (21781)
Grenzen: Unbekannt ([Feststellen](#))

Felder

#	Name	Typ	Länge	Null	Voreinstellung
1	ogc_fid	int4	4	Y	
2	_tid	int4	4	Y	
3	gemeinde	int4	4	Y	
4	the_geom	geometry (Polygon,21781)		Y	
5	name	text		Y	
6	zentroid	geometry (Point,21781)		Y	

Definition anzeigen

```
SELECT fn.ogc_fid, fn._tid, 198 AS gemeinde, (fn.the_geom)::geometry(Polygon,21781) AS the_geom,
replace((fn.name)::text, ' '::text, ' '::text) AS name, (st_pointfromtext((((('POINT('::text || fn.geometrie_0) || ' '::text) ||
fn.geometrie_1) || ' '::text), 21781)::geometry(Point,21781) AS zentroid FROM av.nomenklatur_flurname_areas fn
UNION SELECT fn.ogc_fid, fn._tid, 194 AS gemeinde, (fn.the_geom)::geometry(Polygon,21781) AS the_geom,
replace((fn.name)::text, ' '::text, ' '::text) AS name, (st_pointfromtext((((('POINT('::text || fn.geometrie_0) || ' '::text) ||
fn.geometrie_1) || ' '::text), 21781)::geometry(Point,21781) AS zentroid FROM
av_greifensee.nomenklatur_flurname_areas fn;
```

SQL-Queries in DB-Manager in QGIS

SQL-Abfrage: Speichern Löschen

```
SELECT * FROM av_user.flurnamen WHERE ST_Area(the_geom) > 100000;
```

Ausführen (F5) 82 Zeilen, 0.0 Sekunden Löschen

Ergebnis:

	ogc_fid	_tid	gemeinde	the_geom	name	zentroid
1	18	87000070	198	0103000020...	Langmorgen	0101000020...
2	27	87000079	198	0103000020...	Brunnacher	0101000020...
3	32	87000085	198	0103000020...	Hopperen	0101000020...
4	40	87000095	198	0103000020...	Hard	0101000020...

☒ Als neuen Layer laden

Spalte mit eindeutigen ganzzahligen Werten Geometriespalte Spalten laden

Layername (Präfix) Jetzt laden!


☐ Objektabfrage nach ID vermeiden

Schließen

PostGIS mit UMN Mapserver

```
LAYER
  NAME 'Bedeckung_Grundwasserleiter'
  TYPE POLYGON
  EXTENT 688000 240000 704000 254000
  CONNECTIONTYPE postgis
  CONNECTION "dbname='uster' host=localhost port=5432 user='www' password='xx' sslmode=disable"
  DATA 'the_geom FROM "hydrologie"."bed_gwleiter" USING UNIQUE gid USING srid=21781'
  METADATA
    'ows_title' 'Bedeckung Grundwasserleiter'
  END
  CLASS
    NAME 'Bedeckung Grundwasserleiter'
    STYLE
      SYMBOL "schraffur"
      ANGLE 90
      SIZE 10
      COLOR 220 100 100
    END
    STYLE
      SYMBOL 0
      WIDTH 1
      OUTLINECOLOR 220 100 100
    END
  END
END
```

PostGIS mit Geoserver

 **GeoServer**

(DE) Logged in as admin. [\(DE\) Logout](#)

(DE) Server

- (DE) Status
- (DE) Contact Information
- (DE) Global Settings
- (DE) JAI Settings
- (DE) About GeoServer

(DE) Services

- WCS
- WMS
- WFS

(DE) Data

- Stil Manager
- (DE) Workspaces
- (DE) Stores
- (DE) Layers
- (DE) Layer Groups

Layer Preview

(DE) Demos

(DE) Edit Vector Data Source

PostGIS NG
PostGIS Database

(DE) Basic Store Info

Workspace

- gis_uster

Data Source Name

- av_user

Description

- AV-Daten

☒ (DE) Enabled

(DE) Connection Parameters

dbtype

- postgis

host

- 10.63.206.138

port

- 5432

database

- uster

schema

- av_user

user

- www

passwd

- *****

namespace

- http://gis.uster.ch/

max connections

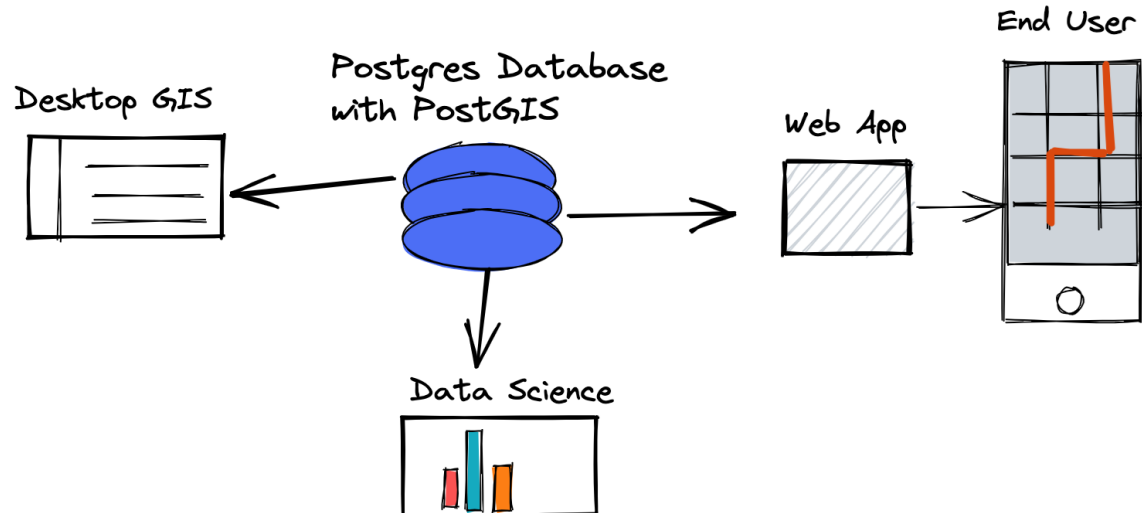
- 10

Title: host

Geo-Datenbank mit Webserver

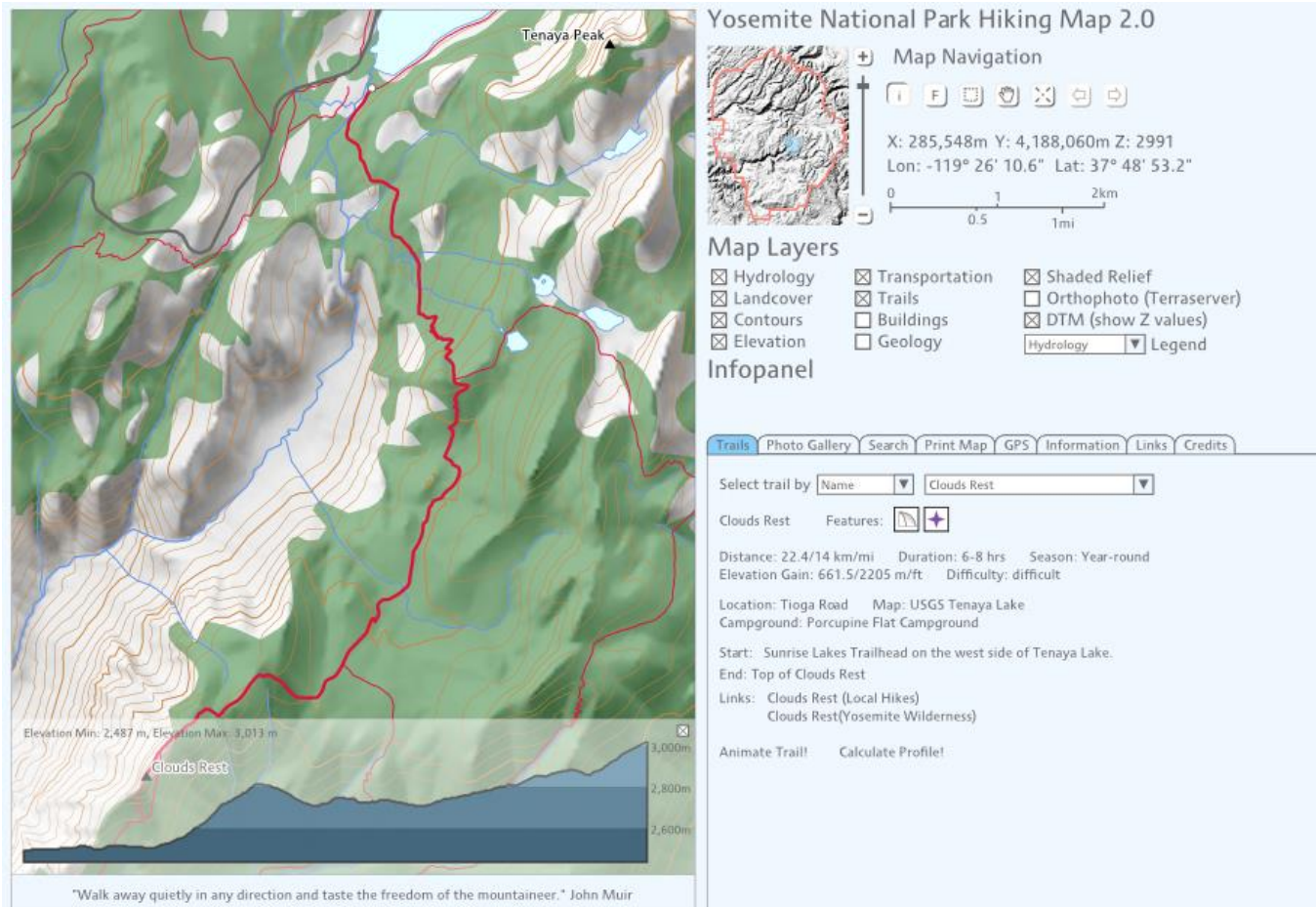
pg_featureserv

- ◆ Webserver verbunden mit PostGIS
- ◆ Selbst geschrieben, bzw. „gehostet“
- ◆ Leichtgewichtiger RESTful Feature Server, Go
- ◆ https://github.com/CrunchyData/pg_featureserv/blob/main
- ◆ siehe auch pg_tileserv



AsSVG() mit Scripten

ST_AsSVG(geometry, [rel], [precision])



<http://www.carto.net/williams/yosemite/>

Darstellen in Virtuellen Globen und Webkartenapps

Google Earth (über PostGIS AsKML, OGR, FME)

NASA Worldwind (über KML)

Microsoft VirtualEarth (über KML)

OssimPlanet (über KML)

Google Maps (über KML)

Export von KML

◆ **PostGIS SQL: AsKML(version,geom,precision)**

◆ **über OGR**

◆ **über FME**