

TriMet's Web Mapping Solution for Portland Transit

Nathan Banks – TriMet
Tim Schaub – The Open Planning Project



Introductions

The Open Planning Project

- Starting and sustaining open source projects

Introductions

The Open Planning Project

- Starting and sustaining open source projects
- High tech hybrid – competing in commercial marketplace, reinvesting in open source projects

Introductions

The Open Planning Project

- Starting and sustaining open source projects
- High tech hybrid – competing in commercial marketplace, reinvesting in open source projects
- GeoServer, OpenLayers, OpenPlans

Introductions

TriMet: Tri-County Metropolitan Transportation District of Oregon

- Public transit service for the Portland metro area
- Service Highlights

Project Description

Two separate, yet related, end products

- Interactive System Map
Replaces current system
- Barriers Application
To support paratransit

Project Description

Desired Features (a selection)

- Google Maps-like 'slippy' interface with an aerial option
- Integrated trip planner (draws trip on map)
- Geocoding to street and taxlot data

Project Description

Desired Features (a selection)

- Hover/Click tools return feature data
- Integrate TransitTracker data
- Measure distance/slope tool
- Print/Save/Send map

Project Team

Scott Davis, Consultant, Denver, CO

Bibiana McHugh, GIS Manager, TriMet

Frank Purcell, Programmer, TriMet

Guy Tinat, Programmer/Systems
Administrator, TriMet

Mike Gilligan, Programmer, TriMet

Nathan Banks, GIS Analyst, TriMet

Potential Strategies

Commercial Solution

Free API Solution (Google, Yahoo,
etc.)

Open Source Solution

Pros – Commercial Solution

- Least Business Risk
- Mature software
- Paid 24/7 support
- Paid training
- Professional Services
- Good Documentation

Cons – Commercial Solution

- Vendor Lock-in
- Proprietary interfaces
- Proprietary formats
- Required Expertise
- Requires data
- Cost: Tens or hundreds of thousands of dollars

Pros – Free API Solution

- Free Service
- Easy to use
- Low barrier to entry
- Targeted at Neogeographers (no formal training)
- Free Data
- Good Documentation

Cons – Free API Solution

- Service Limitations
- Interfaces can (and do) change
- Cannot host your mission-critical infrastructure in-house
- No 24/7 paid support
- Limited ability to incorporate your own data layers
- Potential threat of advertising in the future

Pros – Open Source Solution

- Standards-based Solutions
- Blended commercial and open source solutions possible
- Vendor lock-in is impossible
- Cost: The software is free
- Supports both Websites and Web Services
- Direct support from developers is available in some cases

Cons – Open Source Solution

- Emerging Support
- Not a Turn-key Solution
- Some GIS expertise is required
- Documentation is scarce
- The specs are well defined, but examples and tutorials can be hard to find

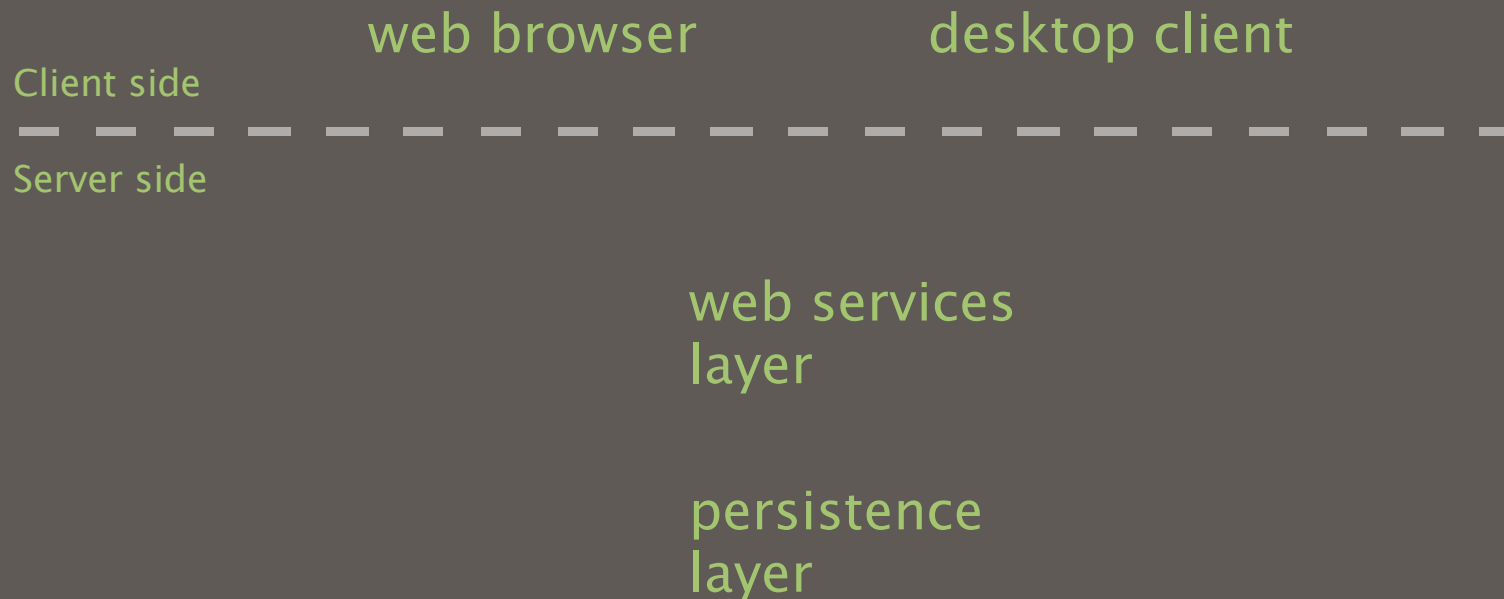
TriMet's Open Source Solution

- GeoServer: Java-based geographic data server
- OpenLayers: Javascript library/API
- TileCache: Python tile caching application
- PostgreSQL/PostGIS: RDBMS for geospatial data

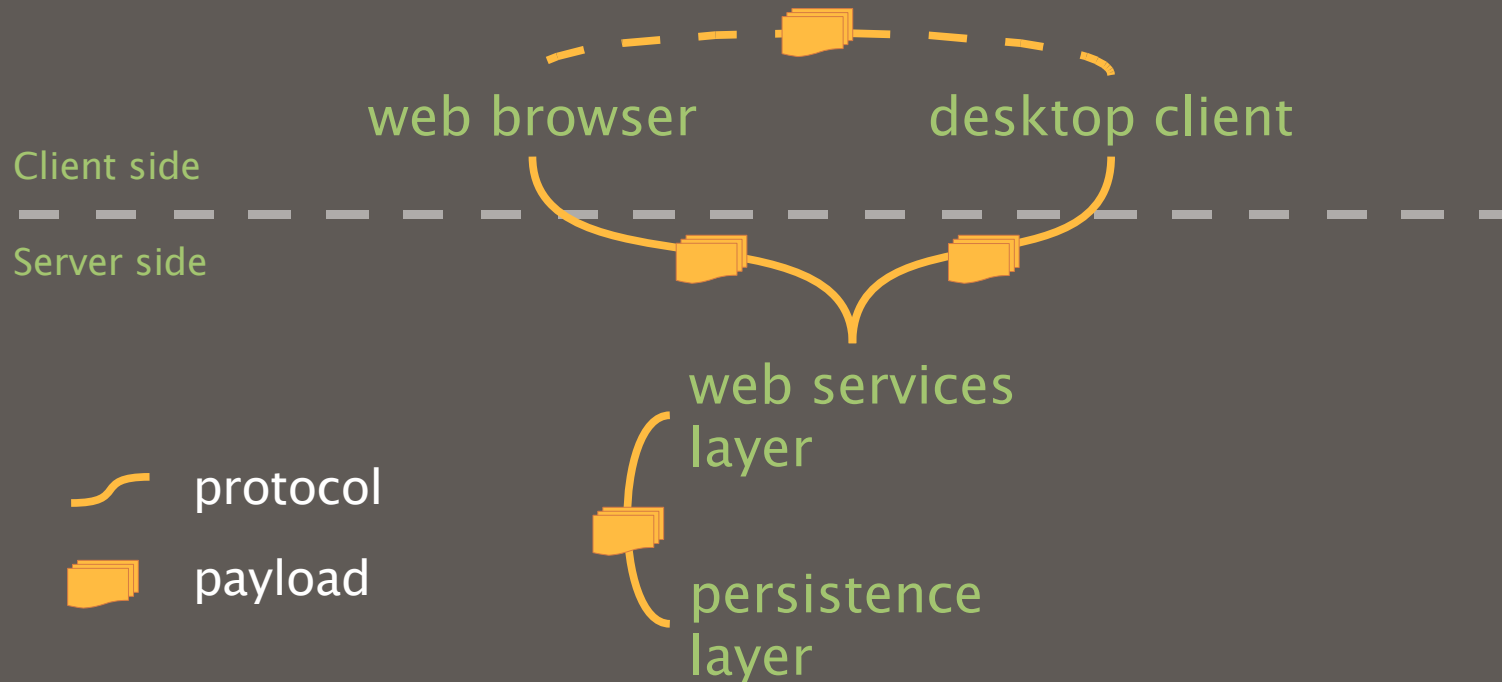
TriMet's Open Source Solution

- Grails: OS web application framework
- Solr: Search engine for geocoding
- Tomcat: Java servlet container
- Apache: The Internet's favorite web server
- CentOS (Community Enterprise Operating System): RedHat derived Linux distribution

Architecture Overview



Architecture Overview



Standards Overview

Open Geospatial Consortium (OGC)

- standards organization
- defines specifications for mapping related protocols and payloads
- supports investigations and implementations

Standards Overview

OGC Web Map Service (WMS)

- Map representation

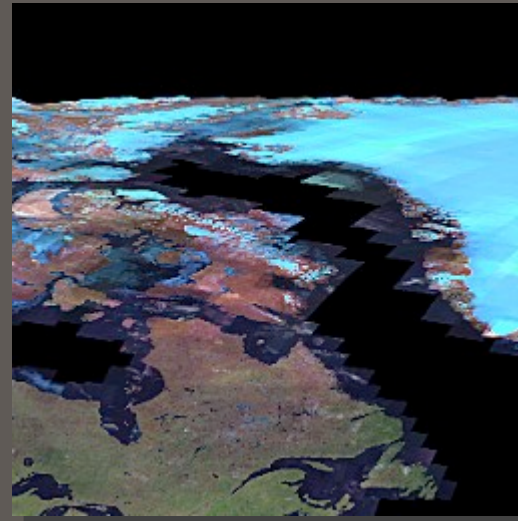
Request: **Get Map**
Service: **NASA JPL WMS**
Layers: **Global Mosaic**
Location: **Iqaluit, Nunavut**

Standards Overview

OGC Web Map Service (WMS)

- Map representation

Request: **Get Map**
Service: **NASA JPL WMS**
Layers: **Global Mosaic**
Location: **Iqaluit, Nunavut**



Standards Overview

OGC Web Feature Service (WFS)

- Map source

Request: **Get Feature**

Service: **GeoServer WFS**

Type: **Archaeological Sites**

Location: **South Dakota, USA**

Standards Overview

OGC Web Feature Service (WFS)

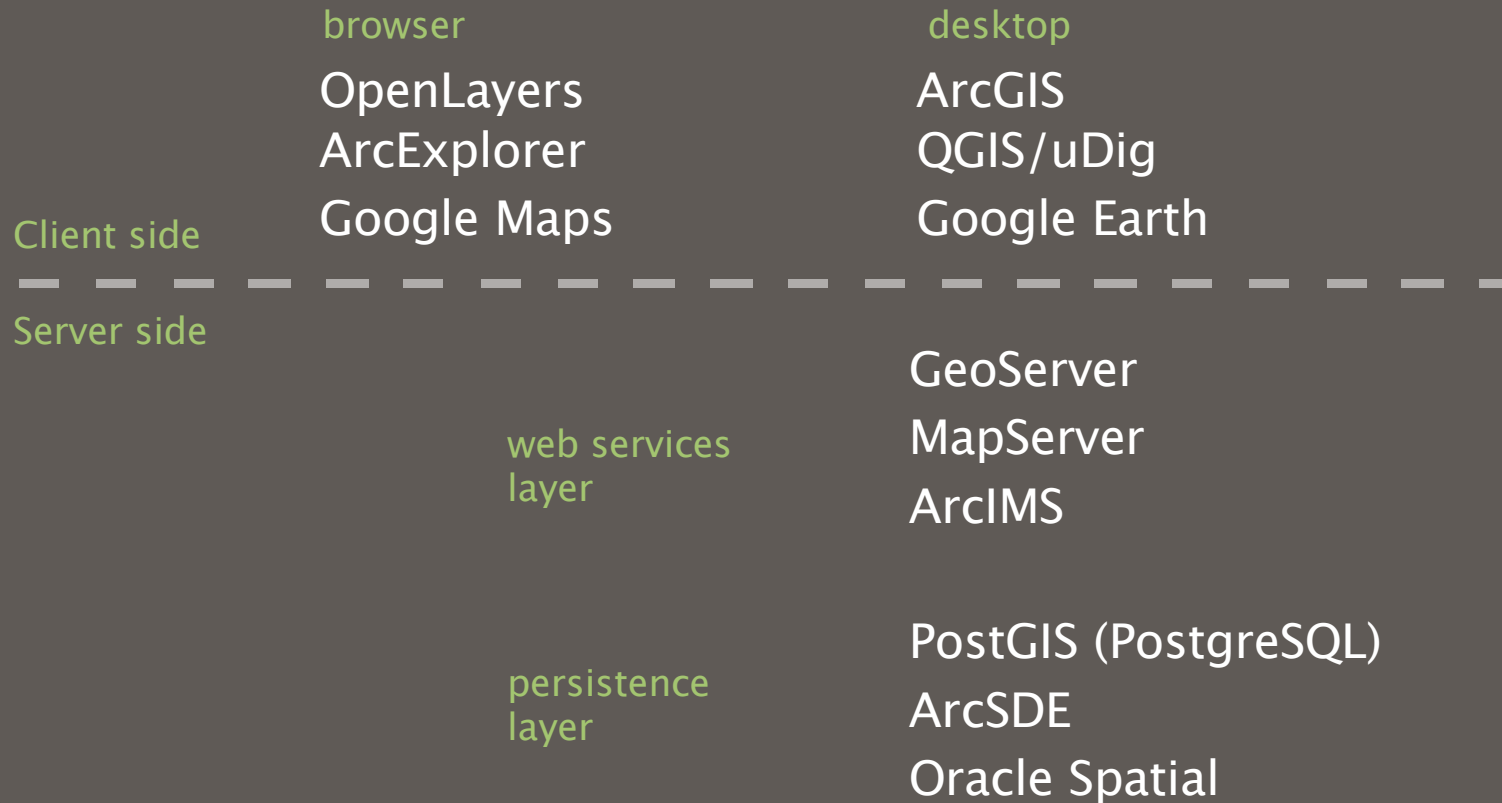
- Map source

Request: **Get Feature**
Service: **GeoServer WFS**

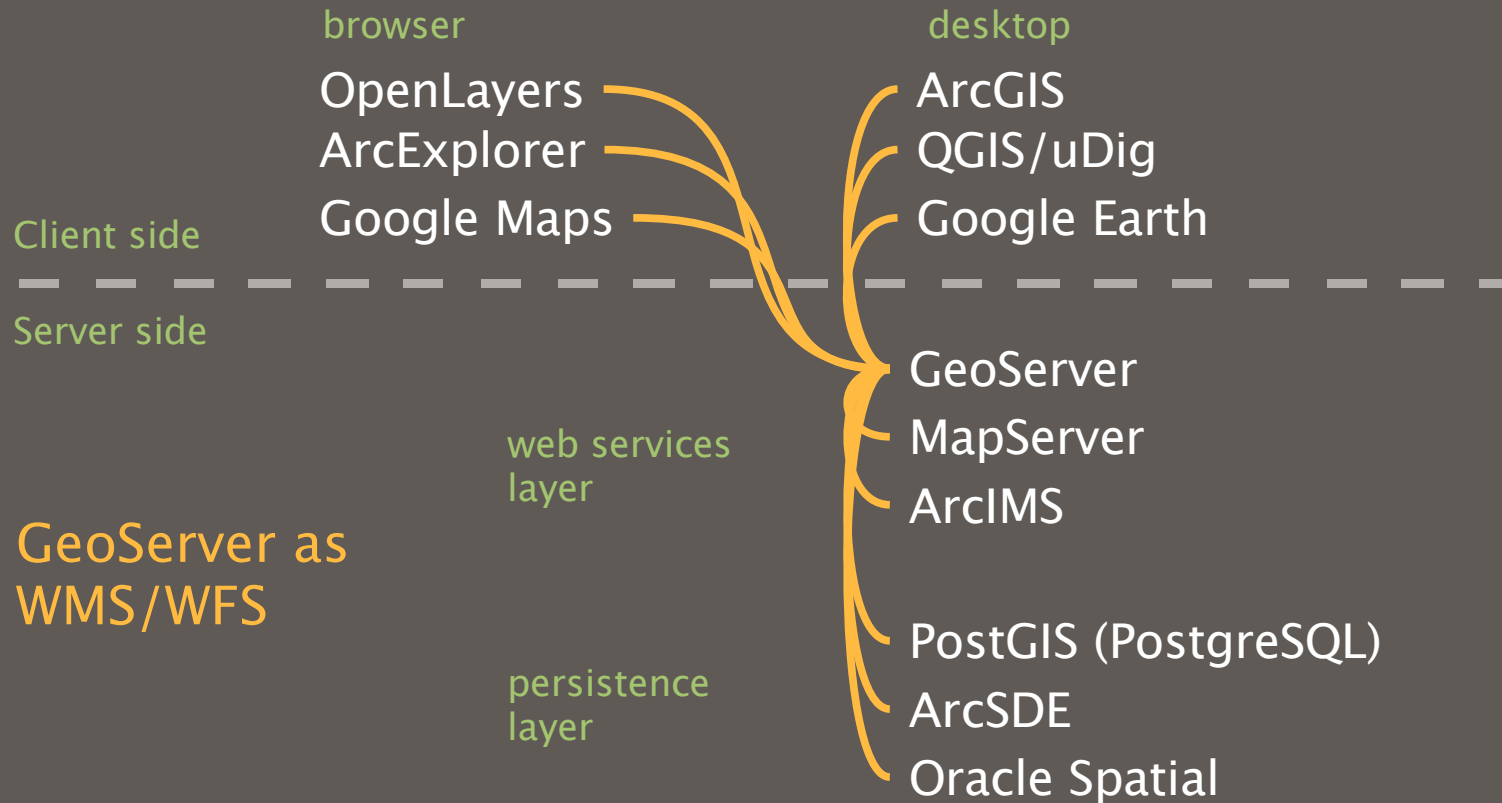
Type: **Archaeological Sites**
Location: **South Dakota, USA**

```
<gml:featureMember>
  <topp:archsites fid="archsites.2">
    <topp:cat>2</topp:cat>
    <topp:str1>No Name</topp:str1>
    <topp:the_geom>
      <gml:Point srsName="http://www.opengis.net/gml/srs/epsg.xml#26713">
        <gml:coordinates decimal="." cs="," ts=" " >
          591950,4923000
        </gml:coordinates>
      </gml:Point>
    </topp:the_geom>
  </topp:archsites>
</gml:featureMember>
```

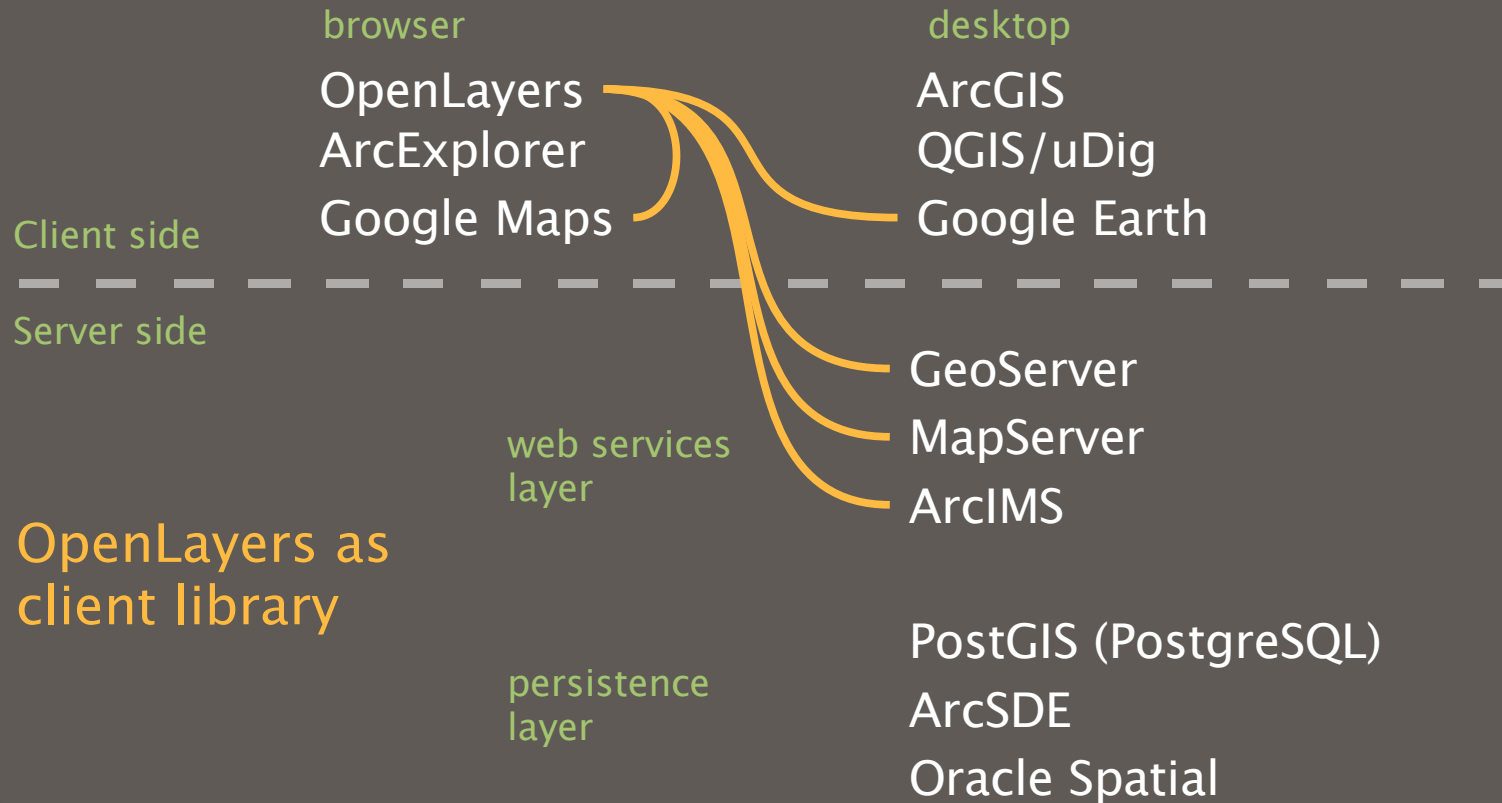
Architecture Overview



Architecture Overview



Architecture Overview



PostGIS (PostgreSQL)

The screenshot shows the pgAdmin III interface. The left pane displays the 'Object browser' with a tree view of the PostgreSQL Database Server 8.2 (localhost:5432). The 'public' schema is expanded, showing various objects including 'Tables (11)'. The 'archsites' table is selected. The right pane shows the 'Properties' tab for the 'archsites' table, displaying the following details:

Property	Value
Name	archsites
OID	17732
Owner	spearfish
ACL	
Primary key	revision, gid
Rows (estimated)	101
Rows (counted)	101
Inherits table(s)	No

The bottom pane shows the SQL definition for the 'archsites' table:

```
-- Table: archsites
-- DROP TABLE archsites;
CREATE TABLE archsites
(
  gid serial NOT NULL,
  cat bigint,
  str1 character varying(80),
  the_geom geometry,
  revision bigint NOT NULL,
  expired bigint NOT NULL DEFAULT 9223372036854775807::bigint,
  CONSTRAINT archsites_pkey PRIMARY KEY (revision, gid),
  CONSTRAINT archsites_revision_fkey FOREIGN KEY (revision)
    REFERENCES changesets (revision) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT enforce_dims_the_geom CHECK (ndims(the_geom) = 2),
  CONSTRAINT enforce_geotype_the_geom CHECK (geometrytype(the_geom) = 'POINT'::text OR
  CONSTRAINT enforce_srid_the_geom CHECK (srid(the_geom) = 26713)
)
```

Retrieving Table details... Done. 0.67 secs

GeoServer

The screenshot shows the GeoServer FeatureType Editor interface in Mozilla Firefox. The browser title is "GeoServer FeatureType Editor - Mozilla Firefox" and the address bar shows the URL "http://geo.openplans.org/geoserver/config/data/typeSelectSubmit.do". The page has a green header with "GeoServer" and "My GeoServer" branding. A navigation menu includes "Welcome", "Config", "Data", "FeatureTypes", and "Edit". A "Logout" link is visible in the top right. The main content area is titled "FeatureType Editor" and "Edit Feature Type definition and schema".

On the left side, there is a "Data:" section with a green progress bar and a table of recent actions:

GeoServer	Sep 18, 3:12 PM
Configuration	Sep 18, 3:12 PM
XML	Sep 18, 3:12 PM

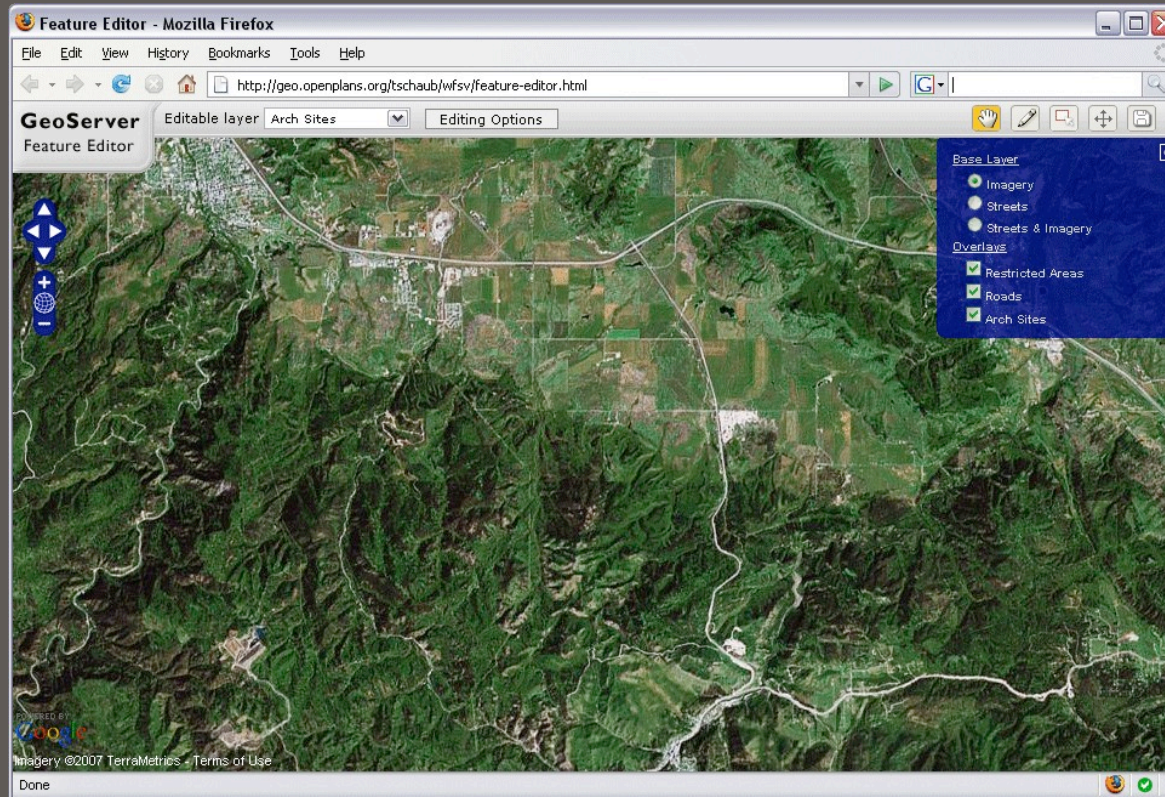
Below the table are "Apply", "Save", and "Load" buttons.

The main form contains the following fields and controls:

- Name: archsites
- Style: point (dropdown menu) with a "Create new SLD" button.
- Additional Styles: A list containing "line", "point", "polygon", and "raster" with left and right arrow buttons for selection.
- SRS: 26713 (text input) with a "Lookup SRS" button and a link to "SRS Help - SRS List".
- SRS WKT: PROJCS["NAD27 / UTM zone 13N", GEOGCS["NAD27", DATUM["North American Datum 1927", SPHEROID["Clarke 1866", 6378206.4, 294.9786982138982, AUTHORITY["EPSG","7008"]], TOWGS84[-4.2, 135.4, 181.9, 0.0, 0.0, 0.0, 0.0], AUTHORITY["EPSG","6267"]], PRIMEM["Greenwich", 0.0, AUTHORITY["EPSG","8901"]], UNIT["degree", 0.017453292519943295], AXIS["Geodetic longitude", EAST], AXIS["Geodetic latitude", NORTH], AUTHORITY["EPSG","4267"]], PROJECTION["Transverse Mercator", AUTHORITY["EPSG","9807"]], PARAMETER["central_meridian", -105.0], PARAMETER["latitude_of_origin", 0.0], PARAMETER["scale_factor", 0.9996], PARAMETER["false_easting", 500000.0], PARAMETER["false_northing", 0.0], UNIT["m", 1.0], AXIS["Easting", EAST], AXIS["Northing", NORTH], AUTHORITY["EPSG","26713"]]
- Native SRS PROJCS["NAD27 / UTM zone 13N", GEOGCS["NAD27", DATUM["North American Datum WKT: 1927" SPHEROID["Clarke 1866" 6378206.4 294.9786982138982

The status bar at the bottom left shows "Done".

OpenLayers

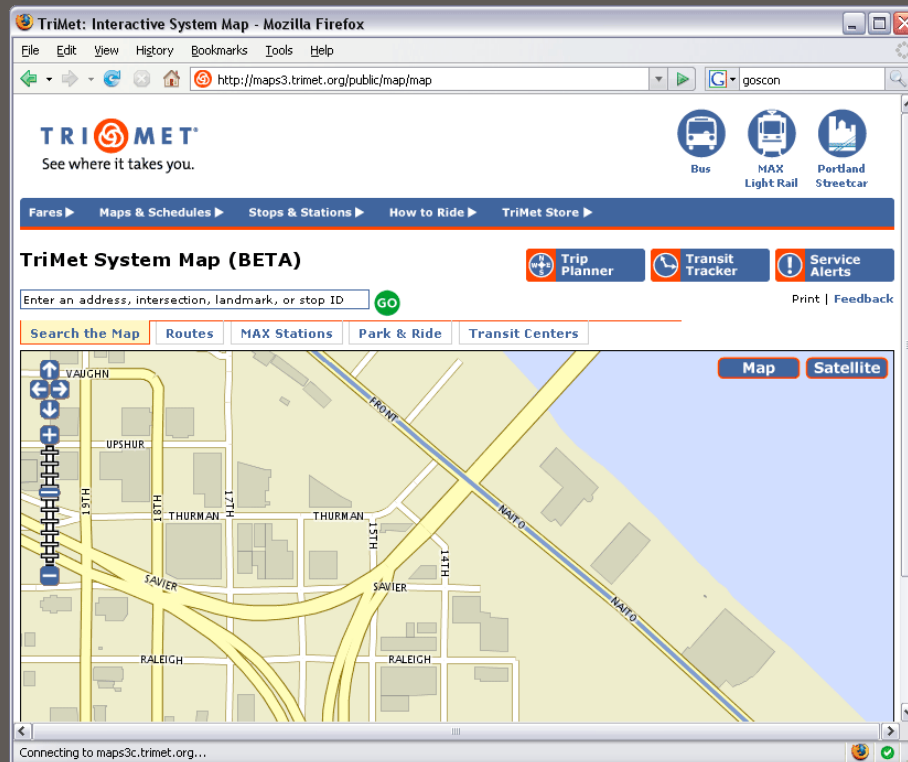


Demos

- OpenLayers
 - WMS layer
 - Vector editing
- GeoServer
 - KML output
 - Versioned editing with Postgres back end

TriMet's Open Source Solution

Walk through of
TriMet's online
mapping system.



Thanks

Nathan Banks
TriMet
banksn@trimet.org

Tim Schaub
The Open Planning Project
tschaub@openplans.org