Community Service with Web-Based Geographic Information Science and Technology (GIST)

Blended Pedagogies for the Twenty-First Century

Andy Anderson Amherst College, Academic Technology

Jon Caris Smith College, Spatial Analysis Lab

Scott Gilman Smith College, Spatial Analysis Lab

Charlie Schweik UMass, Environmental Conservation

Web GIST: A New Course at UMass

Student learning goals are to understand:

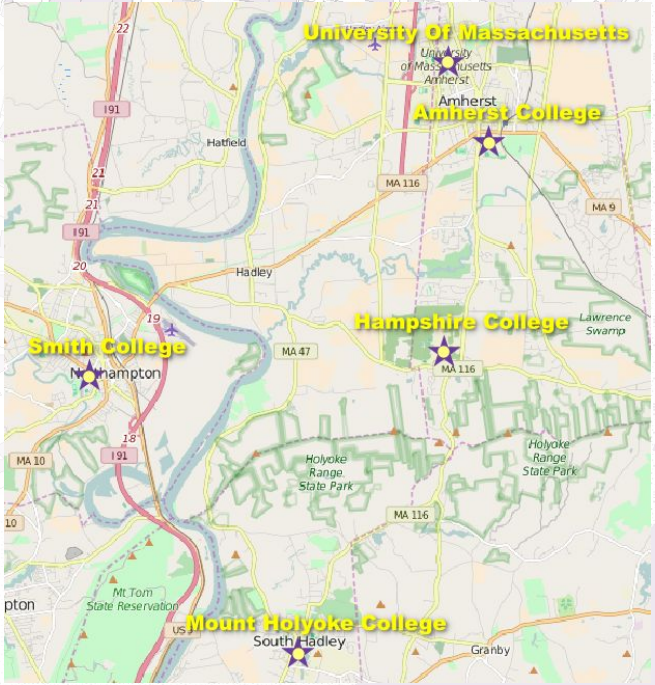
- how the World-Wide Web works;
- the nature of geographic information and how it is processed and visualized on the Web;
- the importance of open standards, software, and data;
- the value of teamwork;
- how to creatively apply their knowledge of GIST and Web programming to benefit a community organization.

William Greenlaw
Open Geospatial Consortium

A Five College Effort

Five College Blended-Learning Grant

- Charlie led first half:
  - OGC, QGIS, PostGIS, GeoServer
- Andy led second half:
  - HTML/CSS/SVG, JavaScript, REST, Bootstrap, Leaflet, Node
- Jon and Scott provided content, feedback, and student support.



Blended Pedagogies

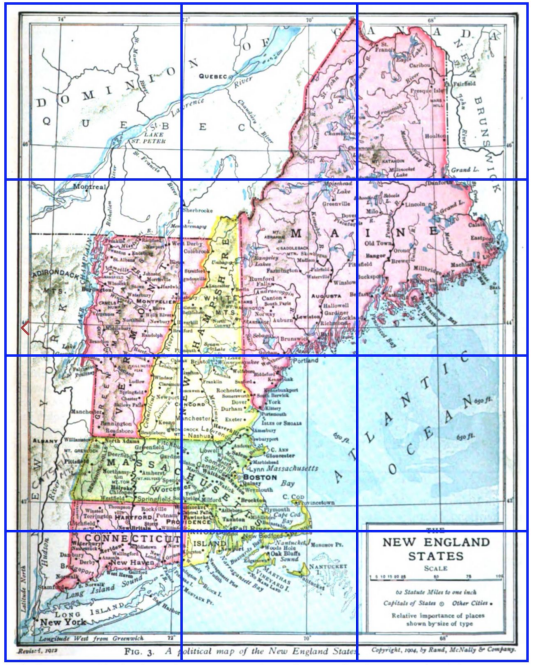
The course implemented several non-traditional teaching methods:

- blended learning
- flipped (workshop) learning
- open learning
- team-based learning
- service learning

Blended Learning

Course materials covered foundational knowledge and:

- were online and organized in Moodle;
- were often visual and interactive;
- used examples from local geography;
- included localized exercises;
- provided short quizzes requiring *recall*, *understanding*, and *application*.



Moodle Quizzes

In an HTML document, markup elements provide

Select one:

- Meaning
- Visual styling
- Document Structure
- All of the above ✓

For an object like

```
var someplace = { County: 'Hampshire', Organization: 1662, Area: 585,
        "Population 1910": 19431, "County Seat": 'Northampton' };
```

the following method returns population density and could be assigned to similar objects.

Select one:

- a. function popDensity(someplace) { return someplace["Population 1910"] / someplace.Area }
- b. function popDensity() { return someplace["Population 1910"] / someplace.Area }
- c. function() { return this["Population 1910"] / this.Area } ✓ Correct!
- d. function() { return someplace["Population 1910"] / someplace.Area }

Flipped (Workshop) Learning

Class time was student-centered, with:

- review of exercise and quiz difficulties;
- Q&A and discussions about important concepts in the course materials;
- active learning with hands-on problem-solving tasks that encourage *analysis*, *evaluation*, and *creativity*;
- project work, eventually increasing to fill class time;
- shared successes as students found solutions to problems.

Five Colleges Workshop

Open Learning

The course explicitly used:

- open educational resources to build content
- open-source software in its examples
- open data whenever possible
- open standards for GIS and Web technologies
- open components in student projects
- open to all Five College students with GIS skill

Open Educational Resources

The main course content was made available via GitBook, and built from many pieces:

- documentation available on-line
- openly-licensed material, some by others, some by us, repurposed with our examples
- brand-new material providing just enough background, sharable via a Creative Commons Attribution license

Chapter 7: Dynamic Data on the World-Wide Web

Lesson Goals

Students are knowledgeable composers of JavaScript, the programming language of the Web, and can create Web pages by manipulating the Document Object Model.

Lesson Outcomes

Students can:

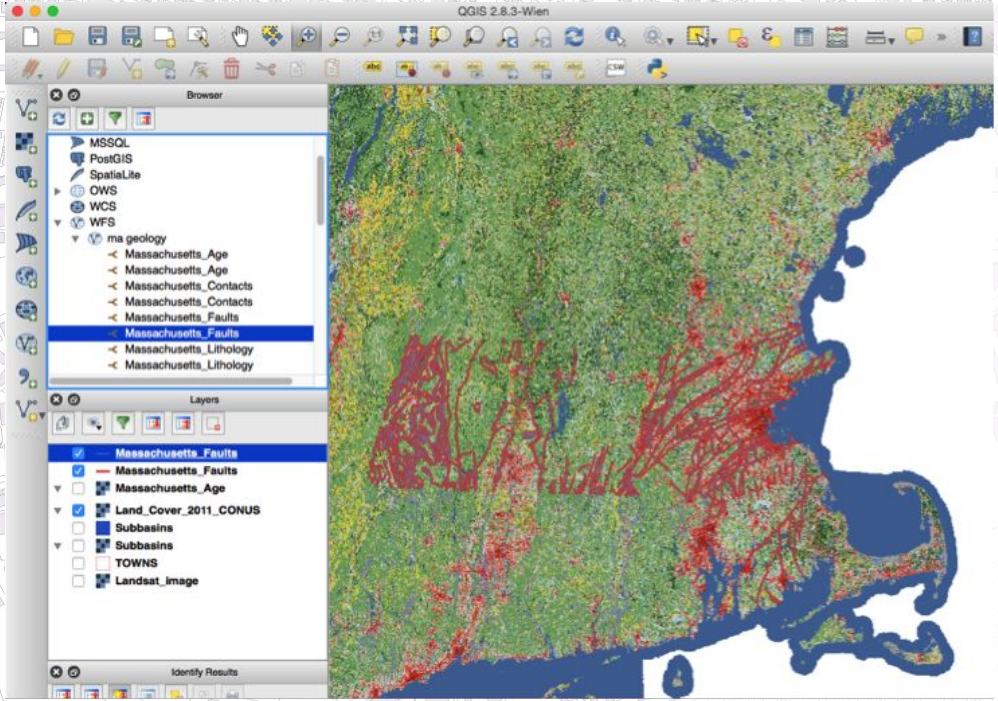
- Program basic JavaScripts to process content
- Develop Web pages using JavaScript
- Manipulate the Document Object Model using the programming library jQuery

---

Introduction

Web pages are built from textual, graphic, and other content using the HyperText Markup Language (HTML) and Scalable Vector Graphics (SVG), and are styled using Cascading Style Sheets (CSS).

Open-Source Software: QGIS



The course introduced students to QGIS:

- a desktop application to process data
- similar to ArcMap
- focus on connecting to Internet servers

Open-Source Servers: OpenVZ

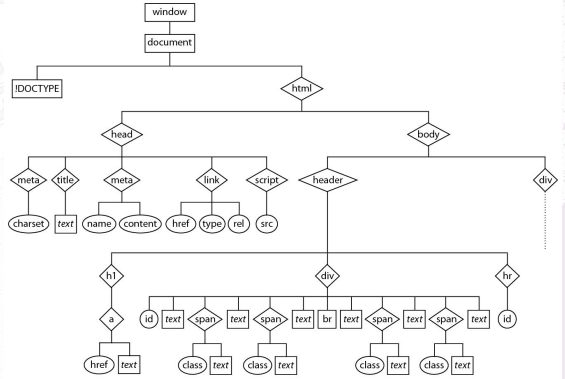
To provide servers for students, we set up an OpenVZ system:

- sits on top of an underlying Linux system
- provides multiple containers for lightweight subsystems:
  - one for the course and one for each student team
- each configured with enterprise software (an OIT first!):
  - Database: PostgreSQL
  - Web map server: GeoServer
  - Web page servers:
    - Apache, Node

World-Wide Web Development

Students learned the basics of developing Web content:

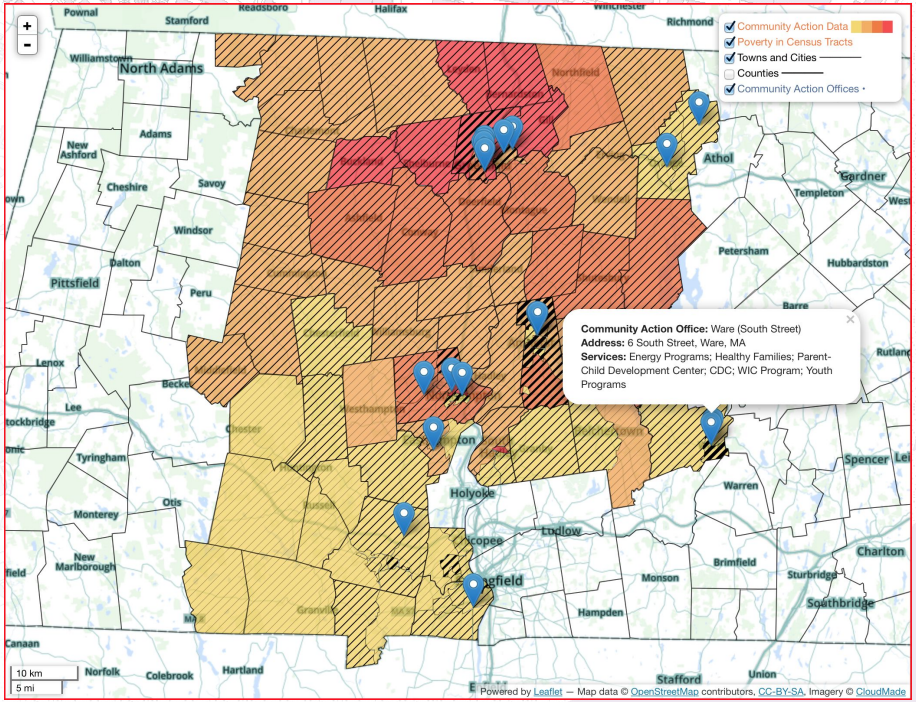
- HyperText Markup Language (HTML) and Cascading Style Sheets (CSS), the languages of documents *and rasters*;
- Scalable Vector Graphics (SVG), the language of *vector features*;
- JavaScript, the programming language that builds documents — *and maps*;



Leaflet.js

Leaflet.js greatly simplifies the creation of Web maps:

- a lightweight Google Maps-style interface
- easily load data from servers, directly or as WMS or WFS



Hack for Western Mass

Team-Based Learning

23 students (14 F, 9 M) formed 7 teams based on interests, and:

- were 2 – 5 in size
- distributed different skills:
  - on 1–5 scale: GIS skill 3±0.7; other tech skill 2.4±0.7
- learned about team roles, project management, and gender dynamics:
  - NDGI = (F – M)/(F + M) > 0 for all teams but one = –1

Service Learning

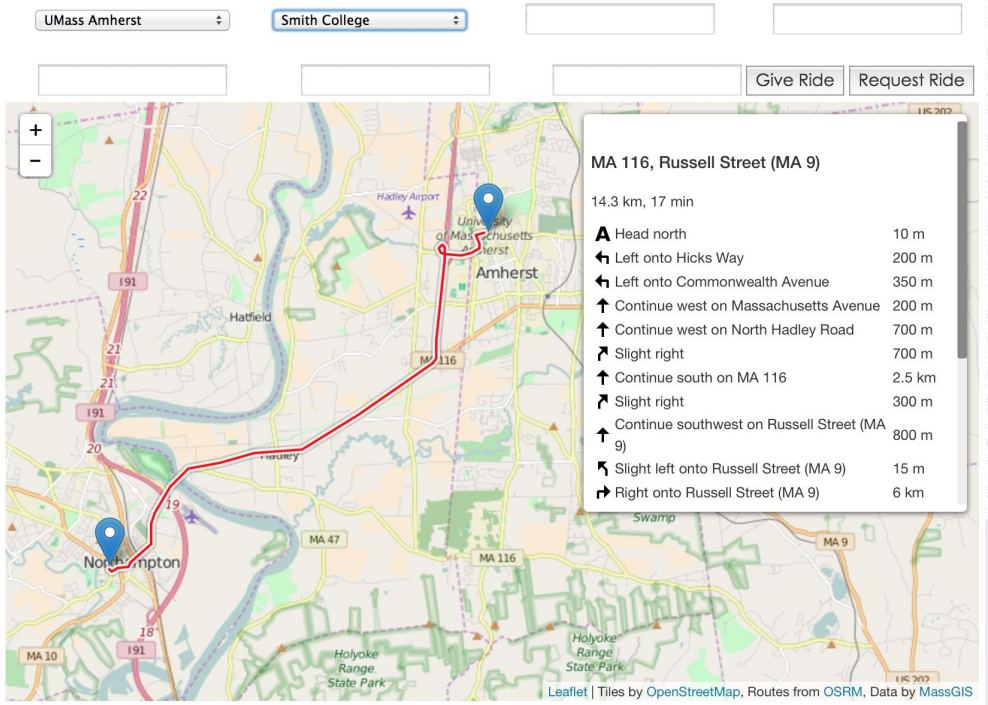
Students created projects that:

- provided a useful service for a community organization or one of the Five Colleges;
- were developed in consultation with their representatives.
- For this first run of the course, we focused on “safe” projects, i.e. our campuses or very familiar groups

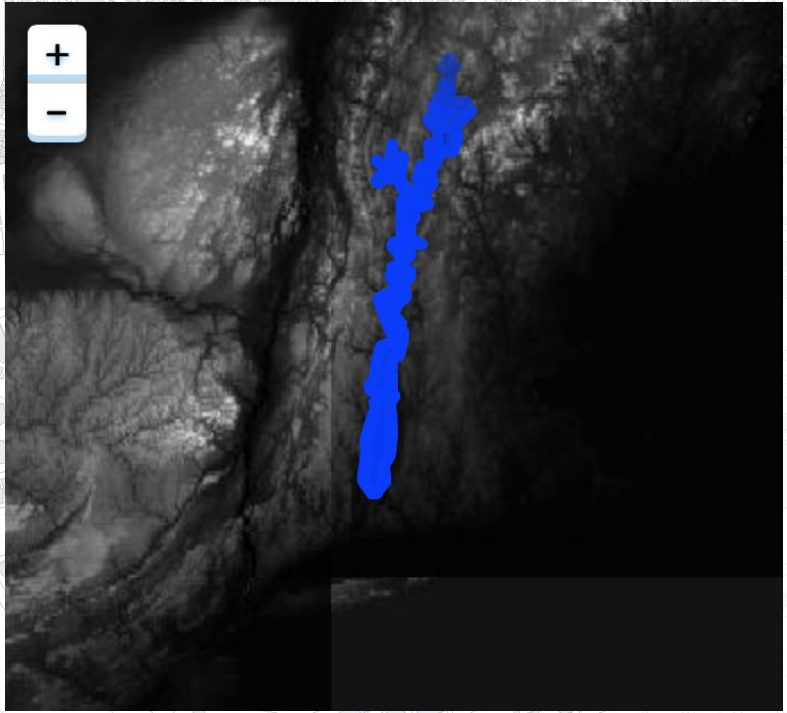
Project: 5CollegeRide

Goal: help fellow students looking for a ride or willing to offer someone else a ride.

- used Leaflet routing engine;
- eventually an account system.



Project: Glacial Lake Hitchcock



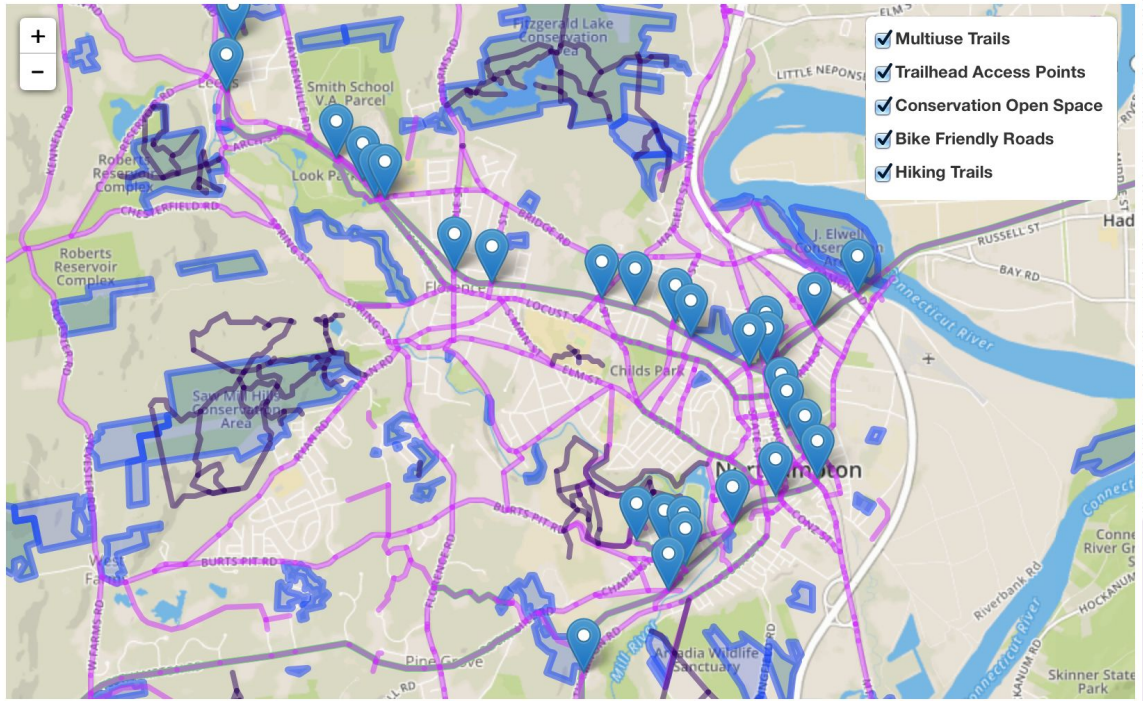
Goal: provide an animated presentation of this prehistoric lake's retreat, 16 Ky – 12 Ky BCE.

- extensive research into current knowledge of extent and locations of dams
- eventually a time slider, perhaps a 3D model

Project: Northampton Trails

Goal: provide an interactive map for the [Friends of Northampton Trails and Greenways].

- eventually routing, mileage, elevation



Project: Smith College Dashboard

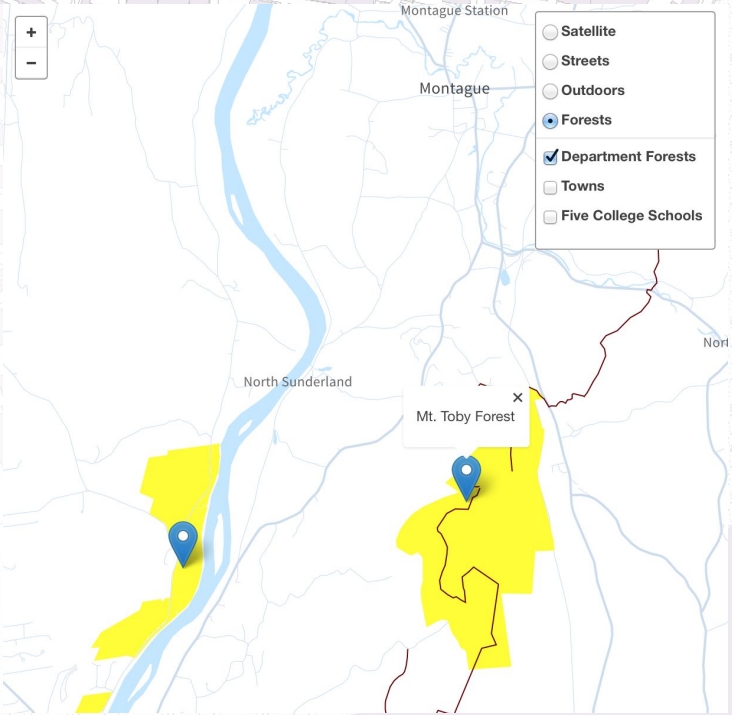
Goal: monitor campus usage of water, gas, and electricity.

- pulled data from existing monitoring system, *Building OS*
- colored buildings by usage
- eventually real-time data and display temporal patterns

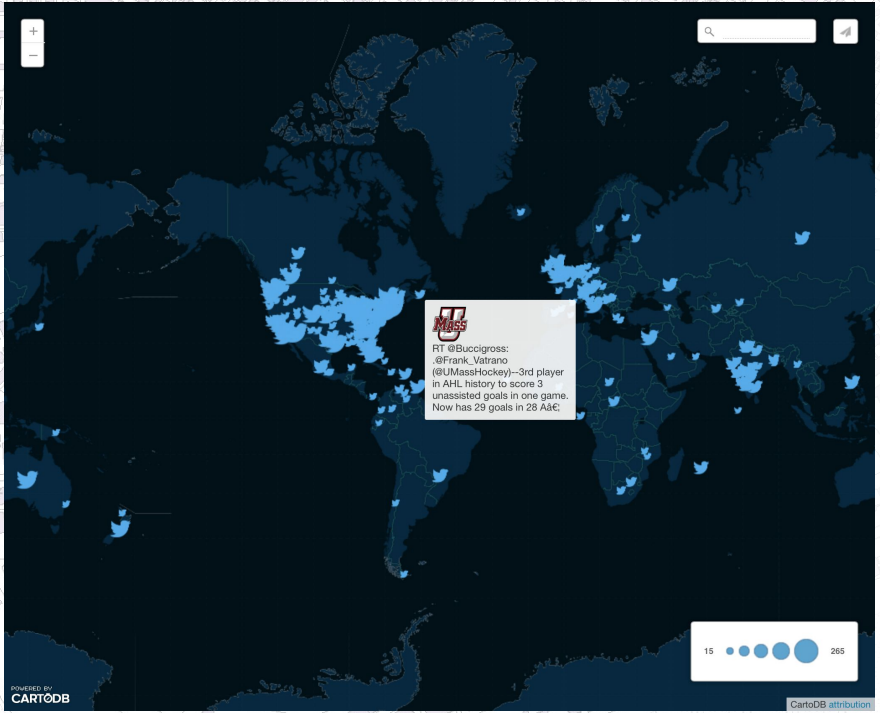
Project: UMass ECo Forests

Goal: provide an interactive map of the forest properties maintained by [UMass Department of Environmental Conservation](#).

- used Mapbox API (an extension of Leaflet)
- eventually add account system to access “sensitive” information



Project: UMass Twitter Chatter



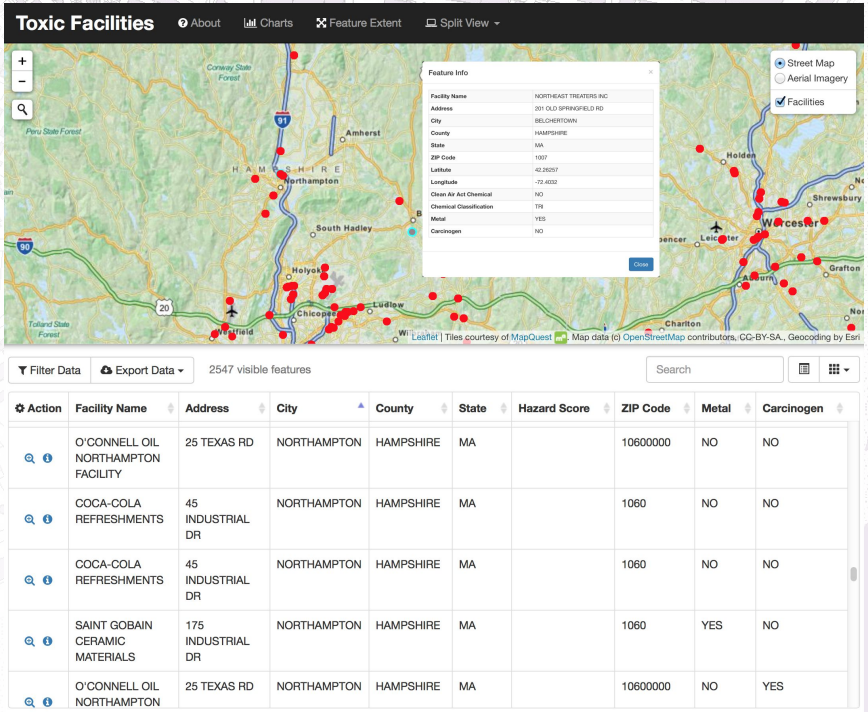
Goal: Track the on and off-campus Twitter pulse for UMass Facilities and Planning.

- Used the Cartodb platform and its Twitter interface
- Eventually expand the keyword search and control the time-frame

Project: Corporate Toxics Database

Goal: an interactive map displaying toxic air pollution facilities, for UMass' Political Economy Research Institute.

- *Nominatim* location search
- open-source database mapping framework
- eventually census info



Issues

- not all content was prepared in advance but JIT or delayed
- not all exercises and applications met our objectives
- some concepts didn't immediately connect to maps
- many students were overwhelmed with learning new languages and learning to program for the first time
- some students shouldered more of the work than others
- not all projects reached a completed state

Future Possibilities

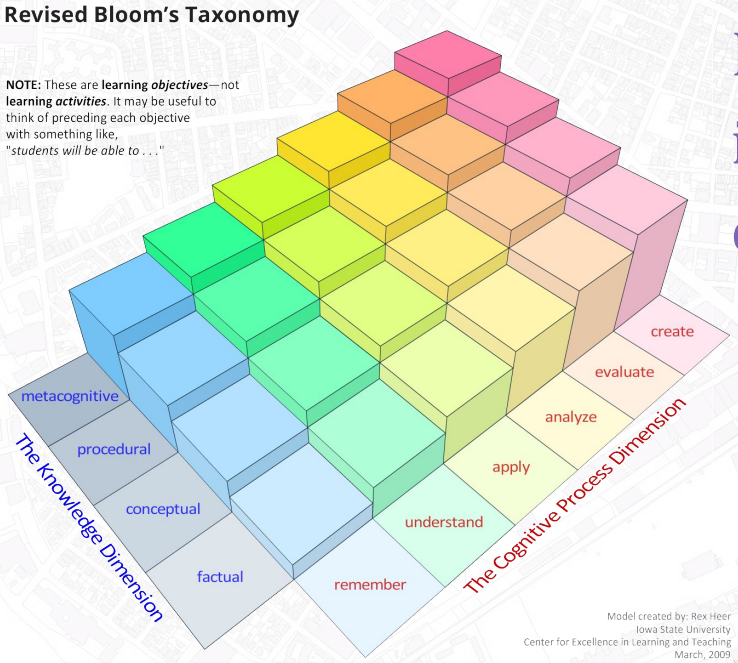
Course materials will hopefully be enhanced with:

- animations and additional interactive features;
- videos of technical procedures;
- videos of instructors and guest speakers;
- build projects using git or Github for sharing and version control.

Acknowledgements

- Teagle Foundation (blended learning grantor)
- Five Colleges, Inc.: Luke Phelan, Nate Therien, Criss Guy
- UMass IT Enterprise Services: Andrew DeSiervo, Lars Marshall
- Sponsors and supporters:
  - Michael Ash, UMass PERI
  - Sasha Stepanov, UMass Facilities and Campus Planning
  - Julie Brigham Grette, UMass Geosciences
  - Tekla Harms, Amherst College Geology

Bloom's Taxonomy of Learning



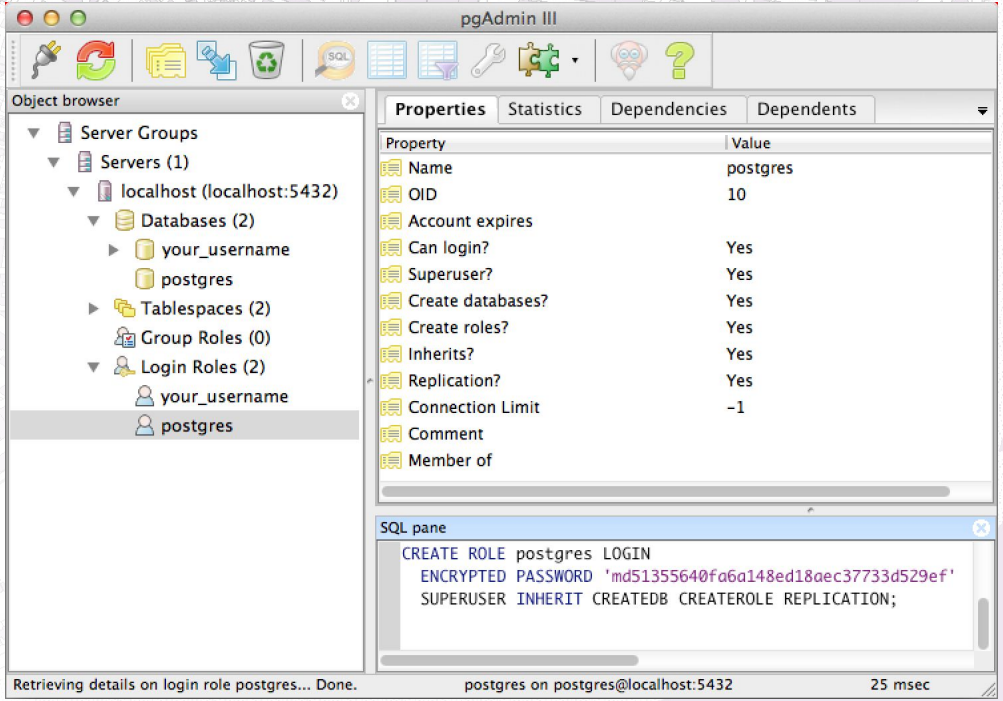
Many of these pedagogies aim to improve learning by reaching more of these cognitive objectives:

- remembering
- understanding
- applying
- analyzing
- evaluating
- creating

Open-Source Server: PostgreSQL

To make data available, students learned to use PostgreSQL + PostGIS:

- a spatial database
- remote administration with pgAdmin + QGIS
- learned to write SQL



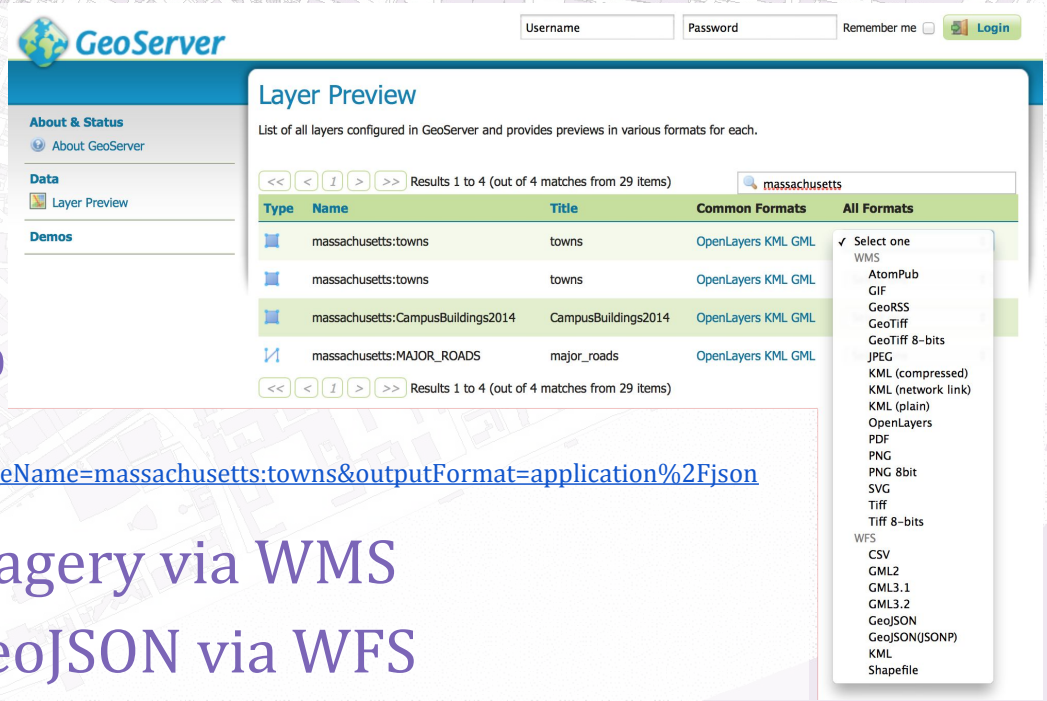
Open-Source Server: GeoServer

Students also worked with [GeoServer]():

- provides a REST interface for the Web

  e.g. http://geoserver/ows?service=WFS&typeName=massachusetts:towns&outputFormat=application%2Fjson

  - tiled basemap imagery via WMS
  - feature data as GeoJSON via WFS



Open-Source Server: Node.js

A lightweight and efficient web server: Node.js

- Express.js framework
- JavaScript can be used:
  - to process forms
  - interact with Postgres
  - apply templates

```
var express = require('express');
var path = require('path');
var favicon = require('serve-favicon');
var logger = require('morgan');
var cookieParser = require('cookie-parser');
var bodyParser = require('body-parser');

var routes = require('./routes/index');
var users = require('./routes/users');

var app = express();

app.set('views', path.join(__dirname, 'views'));
app.set('view engine', 'ejs');

app.use(logger('dev'));
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: false }));
app.use(cookieParser());
app.use(express.static(path.join(__dirname, 'public')));
```

World-Wide Web Libraries

Web development is greatly enhanced by the use of libraries that hide cross-browser complexities:

- jQuery.js to facilitate document object management and server interaction `$('map').click(function)`
- Bootstrap.js to build "responsive" Web pages that work well on mobile devices
- Leaflet.js to build Web maps