Building Better Understanding of Sustainability and Freight Transport

CFIRE 06-04
February 2017

National Center for Freight & Infrastructure Research & Education
Department of Civil and Environmental Engineering
College of Engineering
University of Wisconsin–Madison

Author:
Tracey Holloway
University of Wisconsin–Madison

Principal Investigator:
Tracey Holloway
Nelson Institute for Environmental Studies
and the Department of Atmospheric and Oceanic Sciences
University of Wisconsin—Madison

This page intentionally left blank.

Technical Report Documentation

1. Report No. CFIRE 06-04	2. Government Accession No.	3. Recipient's Catalog No. **CFDA 20.701**
4. Title and Subtitle Building Better Understanding of Sustainability and Freight Transport		5. Report Date **February 2017** 6. Performing Organization Code
7. Author/s Tracey Holloway		8. Performing Organization Report No. **CFIRE 06-04**
9. Performing Organization Name and Address National Center for Freight and Infrastructure Research and Education (CFIRE) University of Wisconsin-Madison 1415 Engineering Drive, 2205 EH Madison, WI 53706		10. Work Unit No. (TRAIS) 11. Contract or Grant No. T002688
12. Sponsoring Organization Name and Address Wisconsin Department of Transportation Research and Library Services Section Division of Business Management 4802 Sheboygan Ave., Room 104 Madison, WI 53705		13. Type of Report and Period Covered **Final Report 09/01/2012–1/30/2017** 14. Sponsoring Agency Code
15. Supplementary Notes Project completed by CFIRE.		
16. Abstract Public information on sustainability and freight transportation is difficult to find outside of technical research journals. Even among college curricula, the freight sector is often overlooked as a major transportation and environmental issue. To fill the freight information void, the project proposes developing web-based freight and environmental sustainability resources geared toward the public, undergraduate students, and practitioners. During the past three and a half years, National Center for Freight and Infrastructure Research and Education (CFIRE) funded research investigating policy solutions to reduce environmental impacts of freight transport through truck-to rail modal shift, and using satellite data to inform spatial freight emissions patterns where ground-monitors do not exist have been conducted. Using data and results from two research projects the project will develop a suite of interactive graphics for the public and students, allowing users to learn about freight activity, emissions and air quality to answer questions specific to their interests and locations. For practitioners, the project will provide detailed guides for acquiring freight activity and emissions data, building emissions inventories, and moving between engineering and air quality data formats, as well as provide freight emissions factor data and inventories generated in the course of our research.		
17. Key Words Air quality management; Environmental impacts; Freight traffic; Modal shift; Pollutants; Sustainable development; Websites (Information retrieval)	18. Distribution Statement **No restrictions. This report is available through the Transportation Research Information Services of the National Transportation Library.**	
19. Security Classification (of this report) **Unclassified**	20. Security Classification (of this page) **Unclassified**	21. No. of Pages 22 / 22. Price **-0-**

Form DOT F 1700.7 (8-72) Reproduction of form and completed page is authorized.

DISCLAIMER

FINAL REPORT – CFIRE 06-04

P.I. Tracey Holloway, Nelson Institute for Environmental Studies and the Department of Atmospheric and Oceanic Sciences, University of Wisconsin—Madison, Madison, WI 53726

With support from CFIRE Grant 06-04, we launched a new program—the Center for Sustainability and the Global Environment (SAGE) visualization activity, or SAGEvis. SAGEvis launched as a user friendly, web-based visualization service. It was designed to help academic researchers and professional scientists efficiently and effectively present data on how atmospheric chemistry is affected by different transportation systems as well as data on U.S. air pollution to a wider, public audience.

SAGEvis provides a number of classic data-visualization tools (e.g. pie chart, lines), but also offers web-based geographic visualization to more efficiently integrate and present data. These visualization tools were uploaded to a web platform (e.g. web browsers) so that users could interact with and manipulate air-quality data, build the digital infrastructure needed to transfer research to a wider public audience, and tailor that data to help answer public questions and build familiarity with today's pressing air-quality issues.

At the conclusion of the project, SAGEvis released four sample works for the public to access.

How to access SAGEvis

URL: http://smog.sage.wisc.edu/SAGEvis/index.html
Web Browsers: Chrome, Firefox, Safari, Opera, etc. (**IE is not recommended**.)
More work here: http://smog.sage.wisc.edu/vis/

SAGEvis on the smog workstation:

The public folder of **SAGEvis** is at /Library/WebServer/Documents/SAGEvis

Data Preprocessing, Data Management

A good visual can convey information to an audience in an intuitive, immediate way that text cannot, but the big-picture goals for the visual must inform the visual's format and organization of its metadata. Meaningful data visualization requires efficient data management, especially when working with web visualization or when using large data sources. This part of the work might not be evident on the front end in the web visualization, but is critical to the performance of visualization.

Our project visualizes two types of data: emissions of pollutants (from a wide range of sources, including on-road transportation such as trucks and trains) and atmospheric concentration of pollutants (the ambient levels of pollutants, associated with emissions, chemical processing, atmospheric transport, and other modulating factors). Data inputs were in Excel (.xls), Comma-separated values (.csv), and netCDF (.nc) files. We developed original scripts (in Java) to preprocess the data, including reading, reformatting, aggregating, and outputting the results into a Javascript (.js) file or Json (.json) style so that the data may be easily accessed from HTML files.

There are two major components to SAGEvis. The first is the classic way of visualizing data with pie charts and timelines. The other is geographic visualization. Geographic visualization of environmental data usually involves GIS—a program for rendering data with a geographic map. Currently, GIS visuals for air quality data do not fit into widely used GIS platforms such as ArcGIS due to the large size of the files, the 3-dimensional structure of the input data, and the high degree of time variability. Rather, air quality data are typically written using specialized programming languages, such as the NCAR Command Language (NCL) or Python, and so are only accessible to a comparatively small community of researchers and programmers.

SAGEvis allows users without specialized programming skills to build their own geographic visualization with a simple-to-use web-based user interface. Most of the technology in SAGEvis already exists online, and includes publically accessible tools like Google Visualization Charts, Google Maps, Google Fusion Tables, and D3. We supplemented these already–existing tools with our own original visualization libraries, written in HMTL5 and Javascript.

All SAGEvis programming activities were performed by Computer Science graduate student Xiujun Li. Prior to launching SAGEvis, Mr. Li worked extensively on data visualization since 2009, and also brought to the project some familiarity with geographic visualization. With his rich background in classic data visualization, he chose to employ classic visualizations, such as pie charts, and lines, based on a few perceptual and cognitive principles – these were discussed at length with Prof. Holloway, and many versions of each visualization was developed and tested.

Initially, Mr. Li used the Google Visualization Charts, which provide some web visualization API. However, Google Visualization was not able to meet all of our needs, and for those cases that were beyond the abilities of Google Visualization, Mr. Li wrote original own visualization libraries in HTML5 (for example, Matrix Visualization, available at http://smog.sage.wisc.edu/vis/graphs/matrix/). Though it takes time to write the visualization API, doing so gives more freedom to manipulate and visualize data.

Another platform employed for SAGEvis was Google Fusion Table, an open tool for simple geographic visualization based on Google Maps and Google Tables. It has a number of limitations:

- Limited capacity: geographic visualizations are limited to a table size of 50 000 rows and 60 columns.
- The query function is quite simple, and can only support limited operations.
- The query function is also slow when the data is large.
- The Google Fusion Table can only support limited visualization.

Ultimately, Google's Fusion Table, was not able to satisfy our needs. We then decided to use Google Maps and build original visualizations over them. Most of our later web geographic visualization is based on the Google Maps style.

We provided a few web geographic visualizations using D3. One disadvantage of Google Maps is that one must load many graphical elements (e.g. circles, rectangles) over the map in order to build the visualization. When the data is large, it takes quite some time to load, and does not always interact well with the graph. D3 offers an advantage here because it can load data in a lightweight script (.json), and interacts well with the map. Here is an example: http://smog.sage.wisc.edu/vis/gis/omi/svg/counties.htm

To summarize, we tried these tools:

- Google Visualization Charts.
- Self-written visualization libraries.
- Google Fusion Table.
- Visualization over Google Maps
- D3 geographic visualization
- In a few cases we also wrote our own visualization libraries in HTML5. This part is not public yet.

In the rest of the document, we describe each visualization tool, including its unique characteristics or features, and the back-end technology needed to build it.

1. Pie Chart

[http://smog.sage.wisc.edu/vis/graphs/pies/]

[Location: /Library/WebServer/Documents/vis/graphs/pies/index.html]

The 3D pie chart shows the various emission sources for one pollutant in a certain year. Pollutants include CO, NOx, PM_{10}, $PM_{2.5}$, SO_2, VOCs, and NH_3, all from 1990 – 2010. The chart allows the user to select a pollutant and year.

The data is from U.S. Environmental Protection Agency NEI (http://www.epa.gov/ttnchie1/trends/).

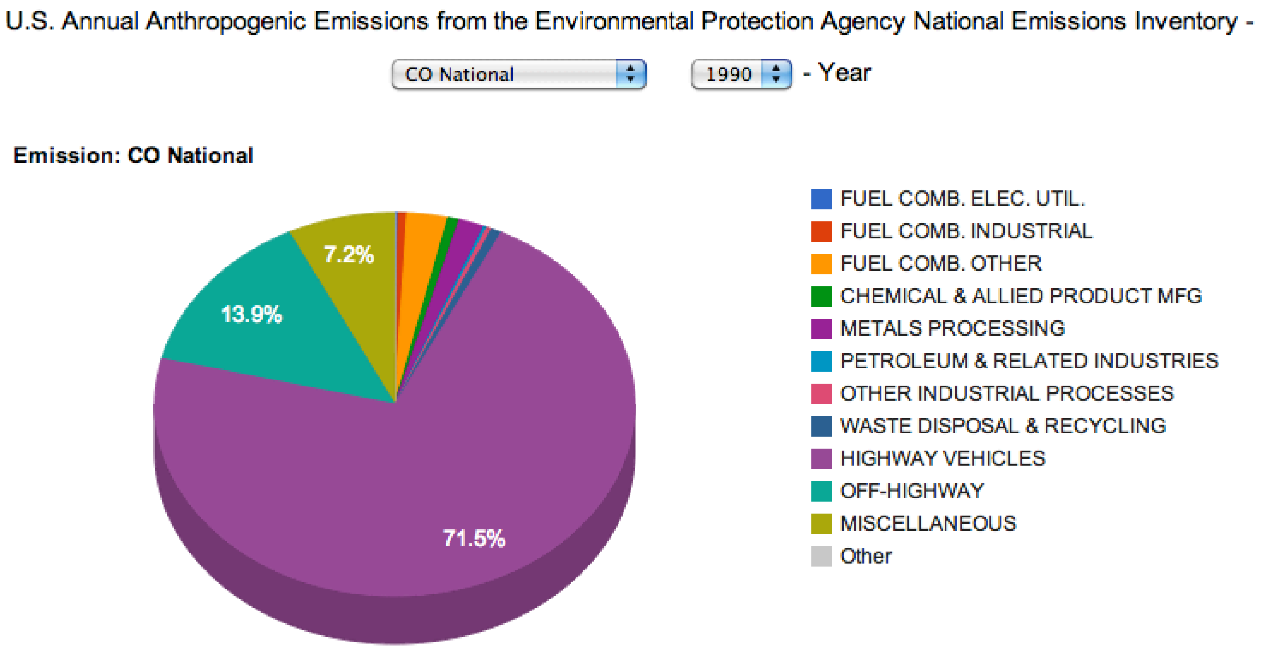


Figure 1. Pie chart

The back-end technology includes JavaScript, Google Visualization Charts, and HTML.

2. Small Multiples - Pies

[http://smog.sage.wisc.edu/vis/graphs/pies/multi.html]

[Location: /Library/WebServer/Documents/vis/graphs/pies/multi.html]

Small multiples is a series of small, similar pie charts, allowing easy comparison. Small multiples can be applied in any chart, e.g. bar, lines, pie etc. Here we built Small Multiples—Pies.

The back-end technology of Small Multiples—Pies includes JavaScript, Google Visualization Charts, and HTML. Here are the steps necessary for building small multiples.

1. Reorganize the data into a JavaScript file. This part of work is done in Java.
2. In the HTML file, load the JavaScript data file at the head section.
    a. Remove a pie chart from the web page by clicking the “X” in the right-hand corner.
    b. Change the color theme by choosing any item from the drop-down menu “Color”.

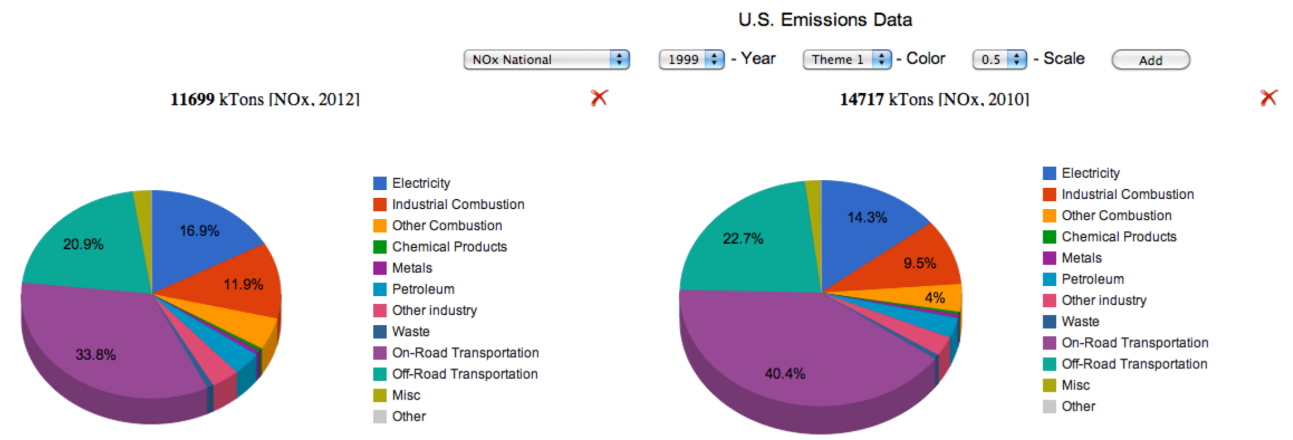


Figure 2. Small Multiples - Pies

3. Lines

[http://smog.sage.wisc.edu/vis/graphs/lines/]
[Location: /Library/WebServer/Documents/vis/graphs/lines/index.html]

The Lines chart provides both a table and lines to show the anthropogenic emissions of CO, NOx, PM10, PM2.5, SO2, VOCs and NH3 from 1990 – 2010. There are two line charts: one is non-normalized with the absolute values, the other is normalized by percent.

The back-end technology of Lines includes JavaScript, Google Visualization Charts, and HTML.

Select the observed data - CO National

Year	FUEL COMB. ELEC. UTIL.	FUEL COMB. INDUSTRIAL	FUEL COMB. OTHER	CHEMICAL & ALLIED PRODUCT MFG	METALS PROCESSING	PETROLEUM & RELATED INDUSTRIES	OTHER INDUSTRIAL PROCESSES	SOLVENT UTILIZATION	STORAGE & TRANSPORT	WASTE DISPOSAL & RECYCLING
1990	363	879	4269	1183	2640	333	537	5	76	1079
1991	349	920	4587	1127	2571	345	548	5	28	1116
1992	350	955	4849	1112	2496	371	544	5	17	1138
1993	363	1043	4181	1093	2536	371	594	5	51	1248
1994	370	1041	4108	1171	2475	338	600	5	24	1225
1995	372	1056	4506	1223	2380	348	624	6	25	1185
1996	408	1188	2741	1053	1599	354	561	1	70	2904
1997	423	1162	2742	1071	1710	367	582	2	71	2948
1998	451	1151	2727	1081	1702	366	590	2	72	3121
1999	496	1213	3829	350	1255	159	571	52	163	3019
2000	484	1219	3081	361	1295	161	592	51	169	1849
2001	485	1253	3088	372	1380	162	615	50	178	1851
2002	657	1267	3550	284	987	357	490	2	118	1594
2003	652	1228	3477	259	934	354	504	2	114	1580
2004	647	1189	3404	233	882	352	519	2	111	1567
2005	642	1149	3330	208	829	350	532	2	107	1553
2006	669	1064	3121	201	833	321	496	3	78	1494
2007	695	979	2913	193	837	291	461	4	48	1436
2008	722	894	2704	186	840	261	425	5	18	1377
2009	722	894	2704	186	840	261	425	5	18	1377
2010	722	894	2704	186	840	261	425	5	18	1377

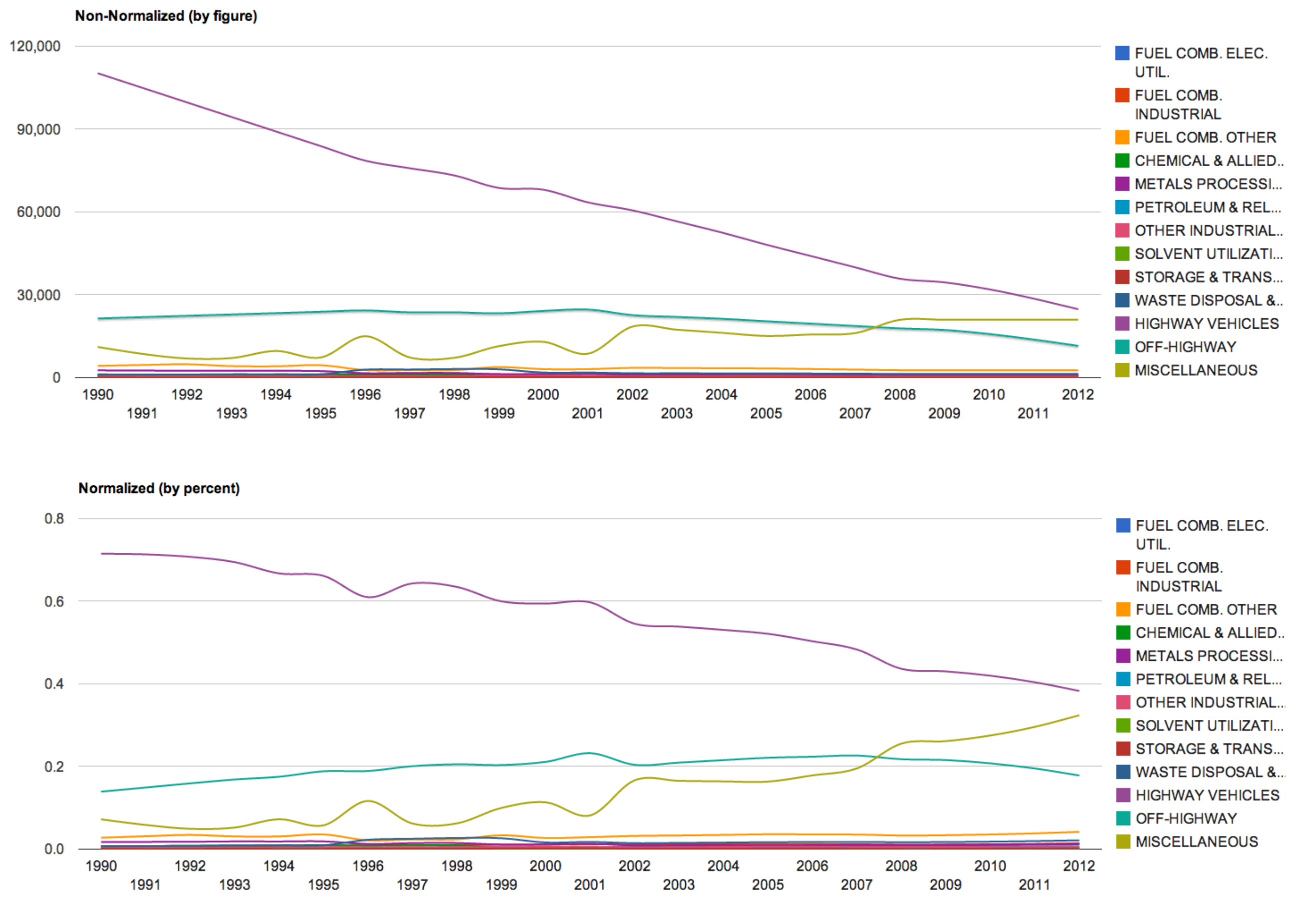


Figure 3. Table and Lines

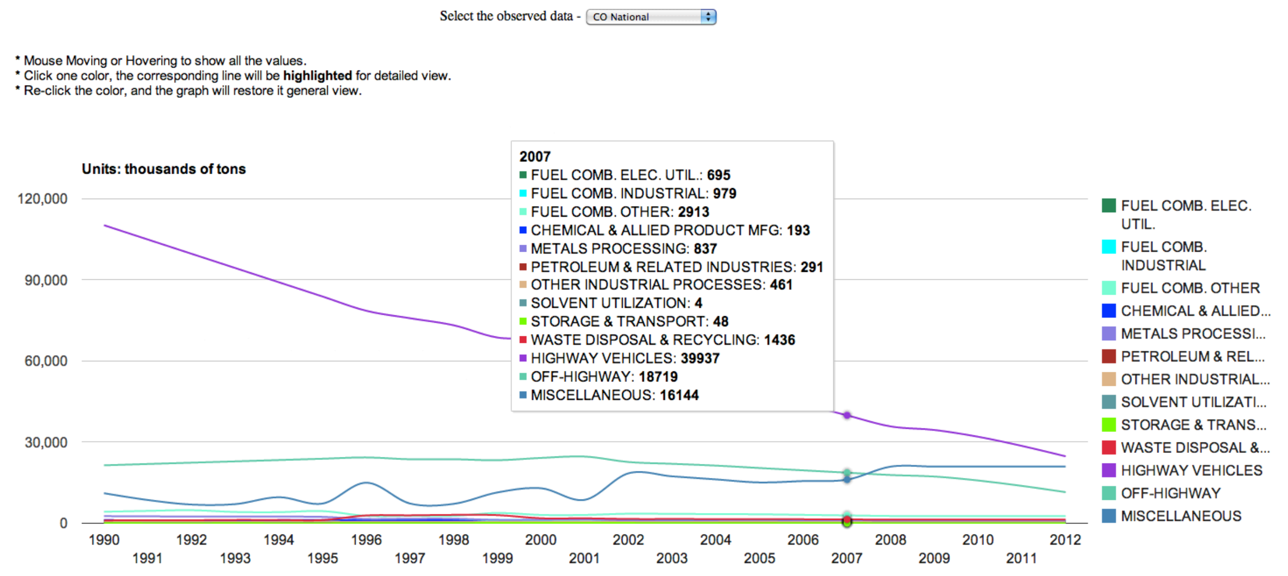
4. Interactive Lines

[http://smog.sage.wisc.edu/vis/graphs/lines/mline.html]
[Location: /Library/WebServer/Documents/vis/graphs/lines/mline.html]

The Interactive Lines chart provides line trends to show the anthropogenic emissions of CO, NOx, PM10, PM2.5, SO2, VOCs and NH3 from 1990 – 2010. It also supports the following interactions:

- Moving the mouse or hovering over the lines shows all the values of the sources for a certain year.
- Clicking one color in the legend highlights the corresponding line for a more detailed view.
- Clicking the color again restores the graph to a general view.

The back-end technology of Lines includes JavaScript, Google Visualization Charts, and HTML.



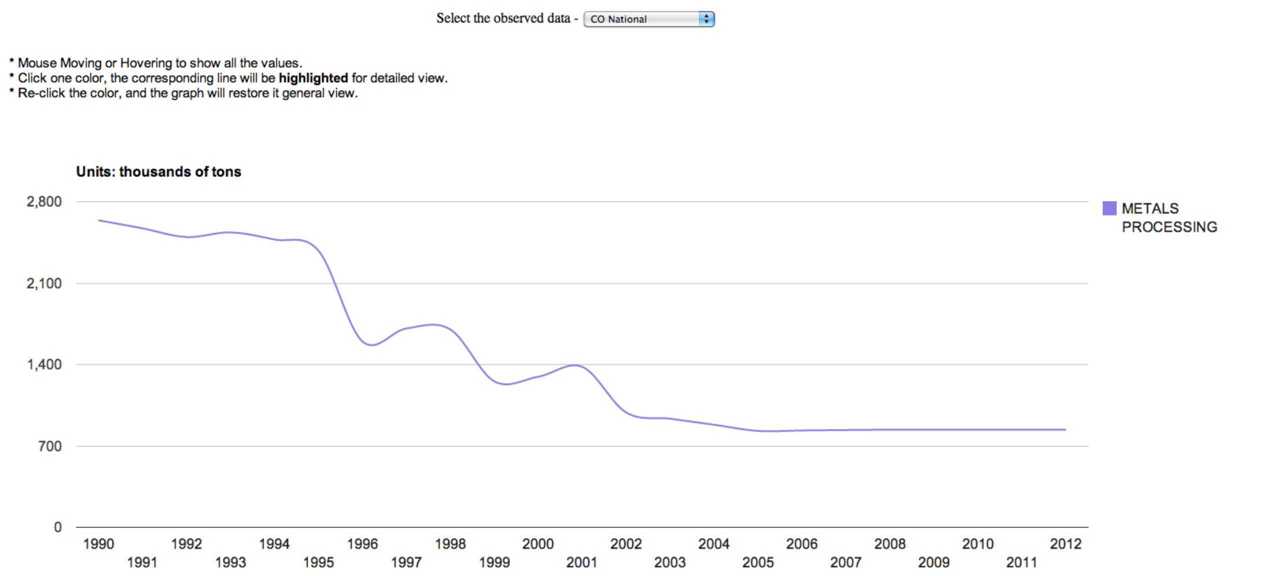


Figure 4. Interactive Lines
the top graph is overview, the bottom graph is detail view

5. Matrix Visualization

[http://smog.sage.wisc.edu/vis/graphs/matrix/]
[Location: /Library/WebServer/Documents/vis/graphs/matrix/index.html]

The Matrix Visualization is a compact representation that uses the gradient of one color to show the trend of one pollutant from 1990 – 2010. The X-axis represents the year, and the Y-axis represents the emission source..

The back-end technology of Matrix Visualization uses JavaScript, HTML5, and HTML. Mr. Li wrote the Matrix Visualization in HTML5, which supports basic graphical drawing.

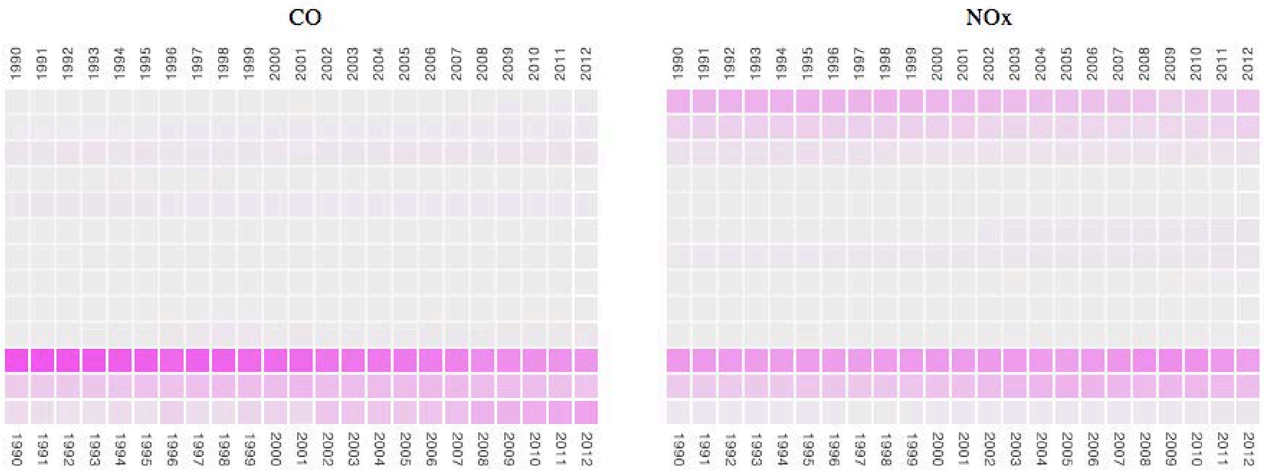


Figure 5. Matrix Visualization

6. Child Pies

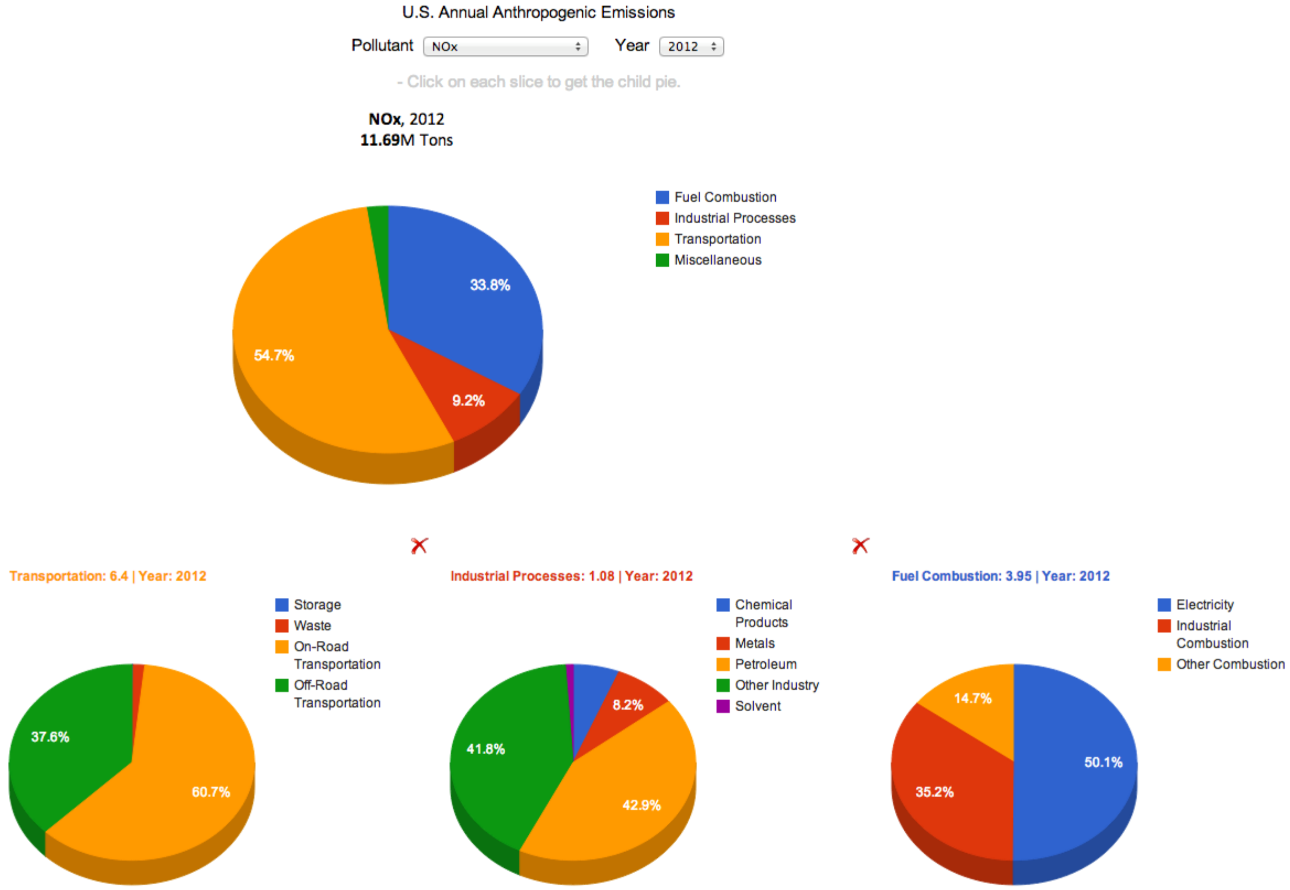
[http://smog.sage.wisc.edu/SAGEvis/childpie.html]

[Location: /Library/WebServer/Documents/SAGEvis/childpie.html]

The Child Pies demonstrate which unique emissions source contributes to the broader emission sector categories overall. Here, fuel combustion, industrial processes, and transportation sectors are broken down, detailing anthropogenic emissions of CO, NOx, PM10, PM2.5, SO2, VOCs and NH3 from 1990 – 2010. It allows the users to click on each pie wedge to view a breakdown of each sector category.

The back-end technology of Child Pies is with Javascript, Google Visualization Charts, HTML.

3. The first step is to re-organize the data into a Javascript file. This part of work is done in Java.
4. In the HTML file, you need to load the Javascript data file at the head section.
5. It also supports to remove a pie chart from the web page by clicking the "X" at the right corner.



6.

Figure 6. Child Pies

7. All Graphs

[http://smog.sage.wisc.edu/vis/graphs/index.html]

[Location: /Library/WebServer/Documents/vis/graphs/index.html]

The All Graphs allows access to many of the previously described interactive graphs based on the U.S. national emissions data from the EPA, including:

A. Matrix charts.
B. Pie charts.
C. Line plots.
D. Gathering all the charts onto one page. All Charts.
E. Small multiples—pies. Add, remove and compare pies.
F. Child Pies.

The back-end technology of All Graph includes JavaScript, Google Visualization Charts, HTML5, and HTML.

Figure 7. All Graphs

8. TreeMap

[http://smog.sage.wisc.edu/vis/graphs/treemap/]

[Location: /Library/WebServer/Documents/vis/graphs/treemap/index.html]

The TreeMap graph provides insights about hierarchical relations via a set of nested rectangles. For example, the large rectangle transportation contains smaller rectangles for on-road and off-road transportation. TreeMap features include:

- Click on one source category (e.g. transportation) to view the sub-sources in this category.
- Right-click to jump back to the parent level.

The back-end technology of TreeMap includes JavaScript, Google Visualization Charts, and HTML.

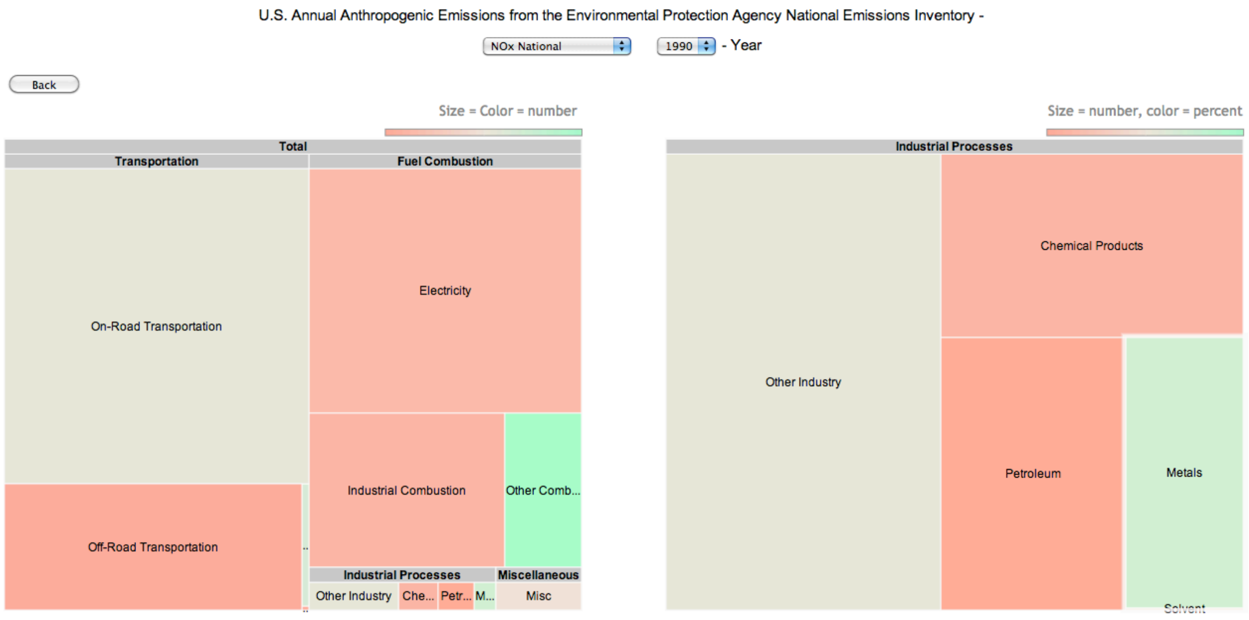


Figure 6. TreeMap

9. Ground-Based and Satellite Measurements

[http://smog.sage.wisc.edu/vis/gis/omi/omi]

[Location: /Library/WebServer/Documents/vis/gis/omi/omi.htm]

The ground-based and satellite measurements graph provides four views from different perspectives to visualize ground-based and satellite NO_2 in 2007. User can:

- Choose one view and one month from the drop-down menus to show the data.
- Click the point on the graph, to display a timeline graph showing the NO_2 trend. The color for the timeline is the same as the color of the point.
- Filter points on the map by checking or unchecking the value intervals.

The back-end technology of Ground-Based and Satellite Measurements includes JavaScript, Google Fusion Table, Google Maps, and HTML. Here are the steps for building ground-based satellite graphs:

1. Firstly, set up some CSS style to customize the Google Maps from its traditional style to an aesthetic web map.
2. In the example below, we tried the Google Fusion Table.. The Google Fusion Table requires a Google Account, and we have set one up specifically for this project: username: sage.wisc, password: lixiujun.wisc.
3. The Google Fusion Table sorts geographic coordinates for the data points.
4. When you filter or exchange data, Google Fusion Table will be queried.
5. For the JavaScript data file, store the 12-month values for each point on the map. This data file should be loaded at the head of HTML file.

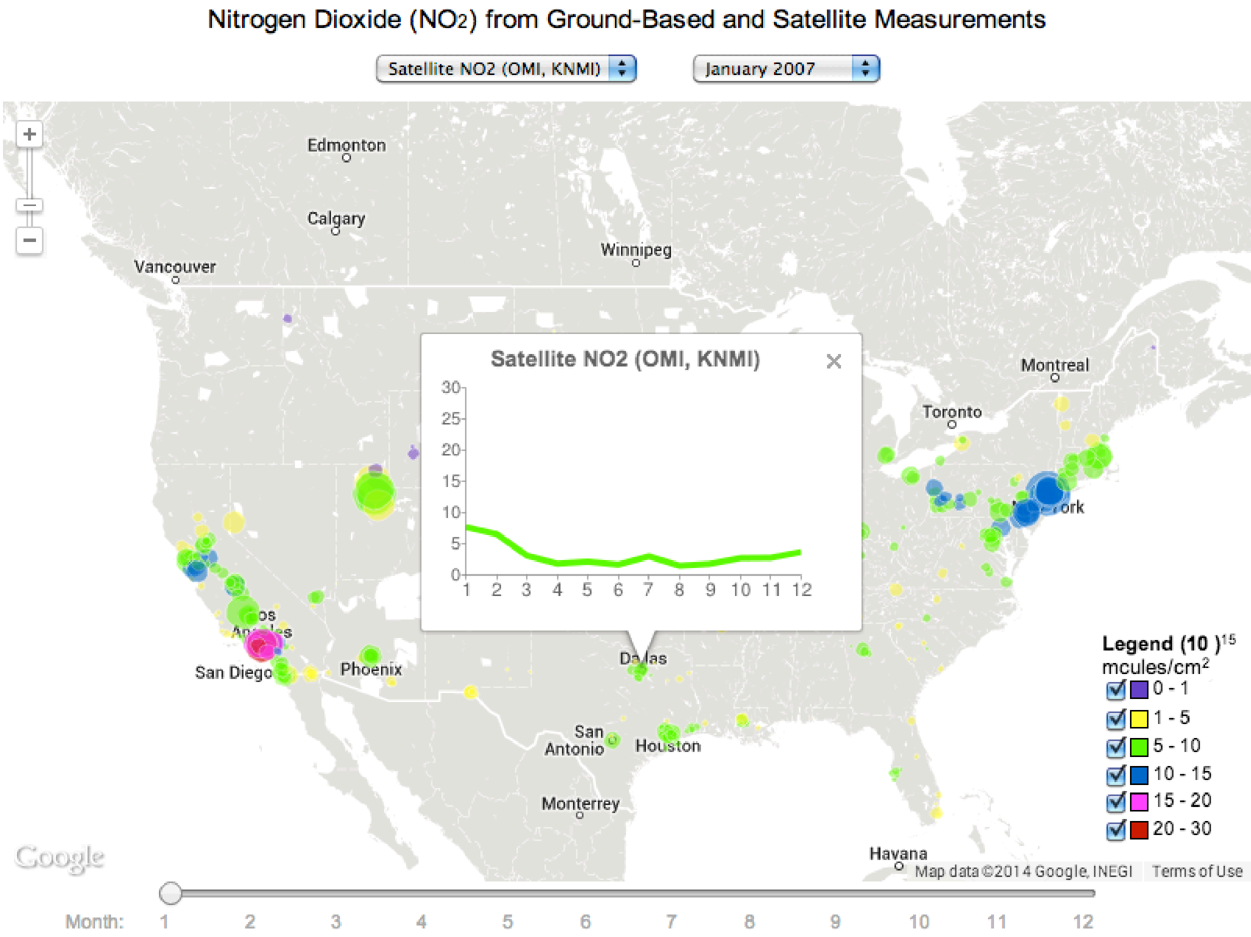


Figure 8. Ground-Based and Satellite Measurements for NO_2

10. Ground-Based and Satellite NO_2

[http://smog.sage.wisc.edu/vis/gis/omi/maps]

[Location: /Library/WebServer/Documents/vis/gis/omi/maps.htm]

The ground-based and satellite measurements graph gives four views from different perspectives : in situ observation, satellite NO_2, #days, R-value. When the user moves or zooms within one map, the other three maps will do the same.

The back-end technology includes JavaScript, Google Fusion Table, Google Maps and HTML.

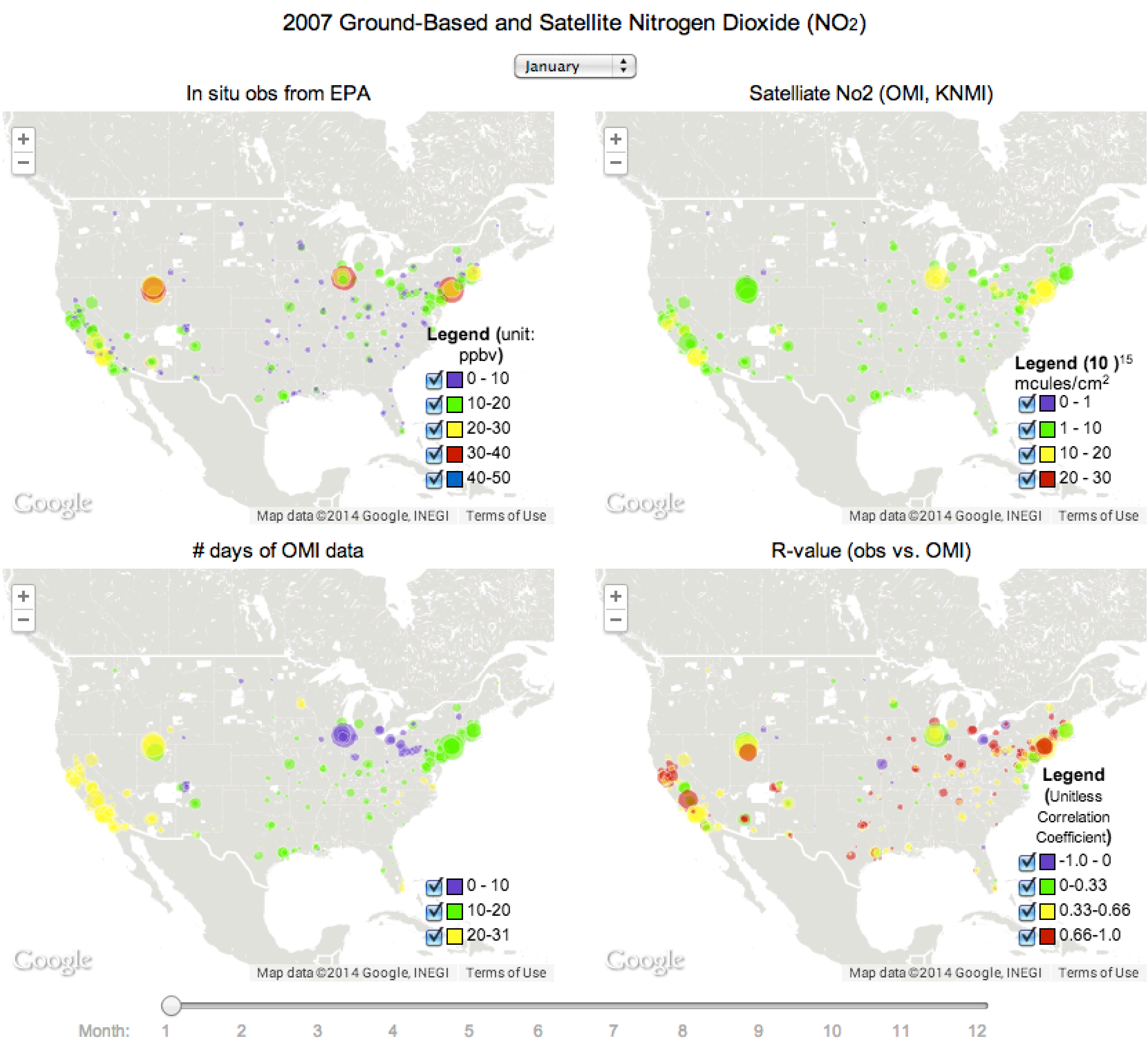


Figure 9. Small Multiples - Ground-Based and Satellite NO2

11. Ground-Based and Satellite NO2

[http://smog.sage.wisc.edu/vis/gis/omi/tranmap]

[Location: /Library/WebServer/Documents/vis/gis/omi/tranmap.htm]

The ground-based and satellite measurements graph gives also allows users to click the Legend to filter the points on the map. Finally, users can input two values to search for points within an interval.

The back-end technology of Ground-Based and Satellite NO2 includes JavaScript, Google Fusion Table and HTML.

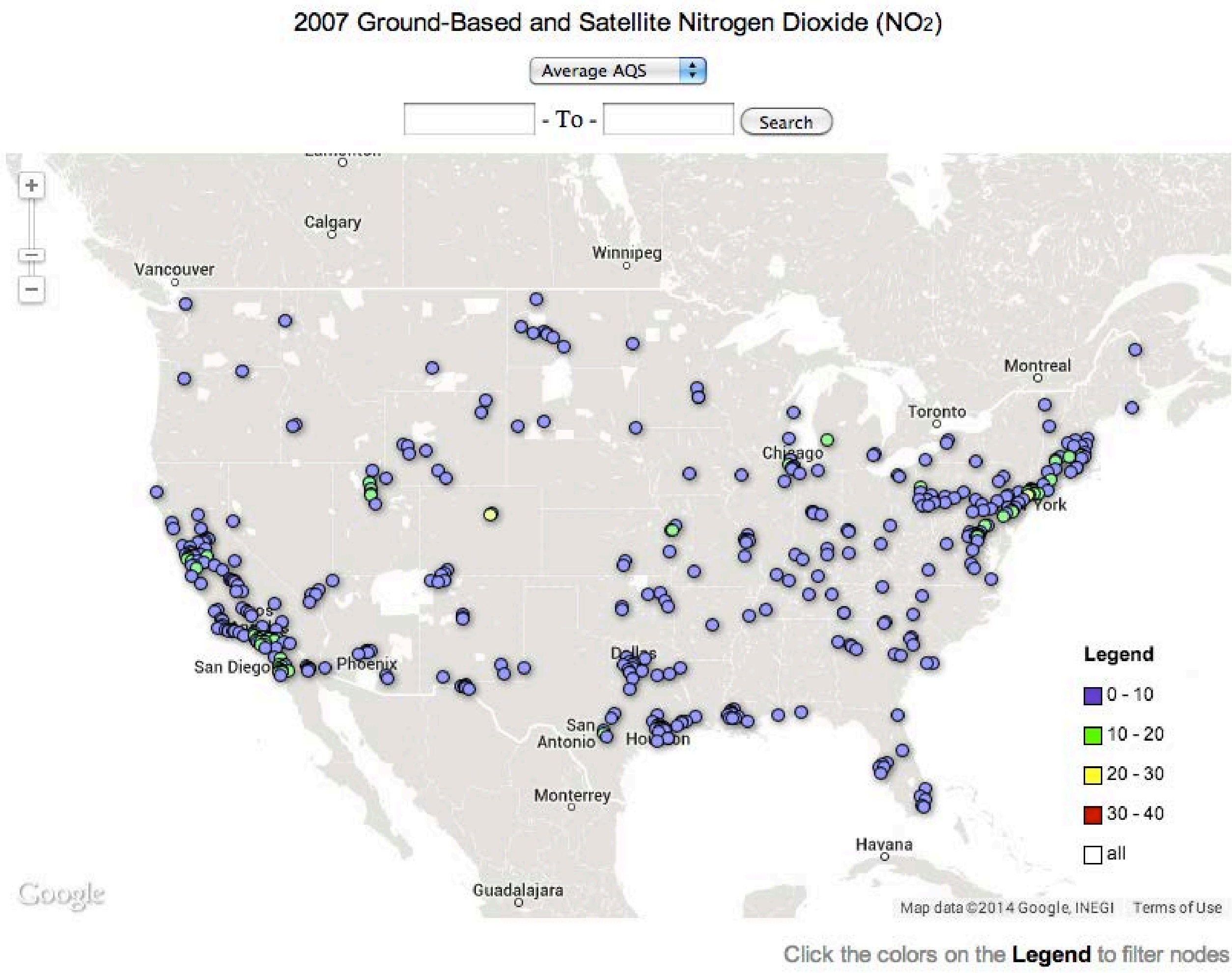


Figure 10. Ground-Based and Satellite NO2

12. Grid-based and Satellite Measurements

[http://smog.sage.wisc.edu/vis/gis/omi/No2map]
[Location: /Library/WebServer/Documents/vis/gis/omi/No2map.htm]

The grid-based graph displays data in a grid (e.g. 12km*12km, 36km*36km) based on Google Maps. It includes a legend in the form of a color bar across the bottom with eight colors. There is also a slider at the top, which indicates the middle value. Users can change the middle value by moving the slider.

The back-end technology of Grid-Based and Satellite Measurements includes JavaScript, Google Maps, CSS and HTML. To set up grid-based maps, make sure to:

- Convert the coordinates into grids with the Lambert Conformal Conic Projection on Google Maps. This step was done in Java.
- Every grid is a rectangle on the Google Maps.
- Store the data, including a longitude and latitude coordinates, in a JavaScript file. The HTML file will load this data file at the beginning.

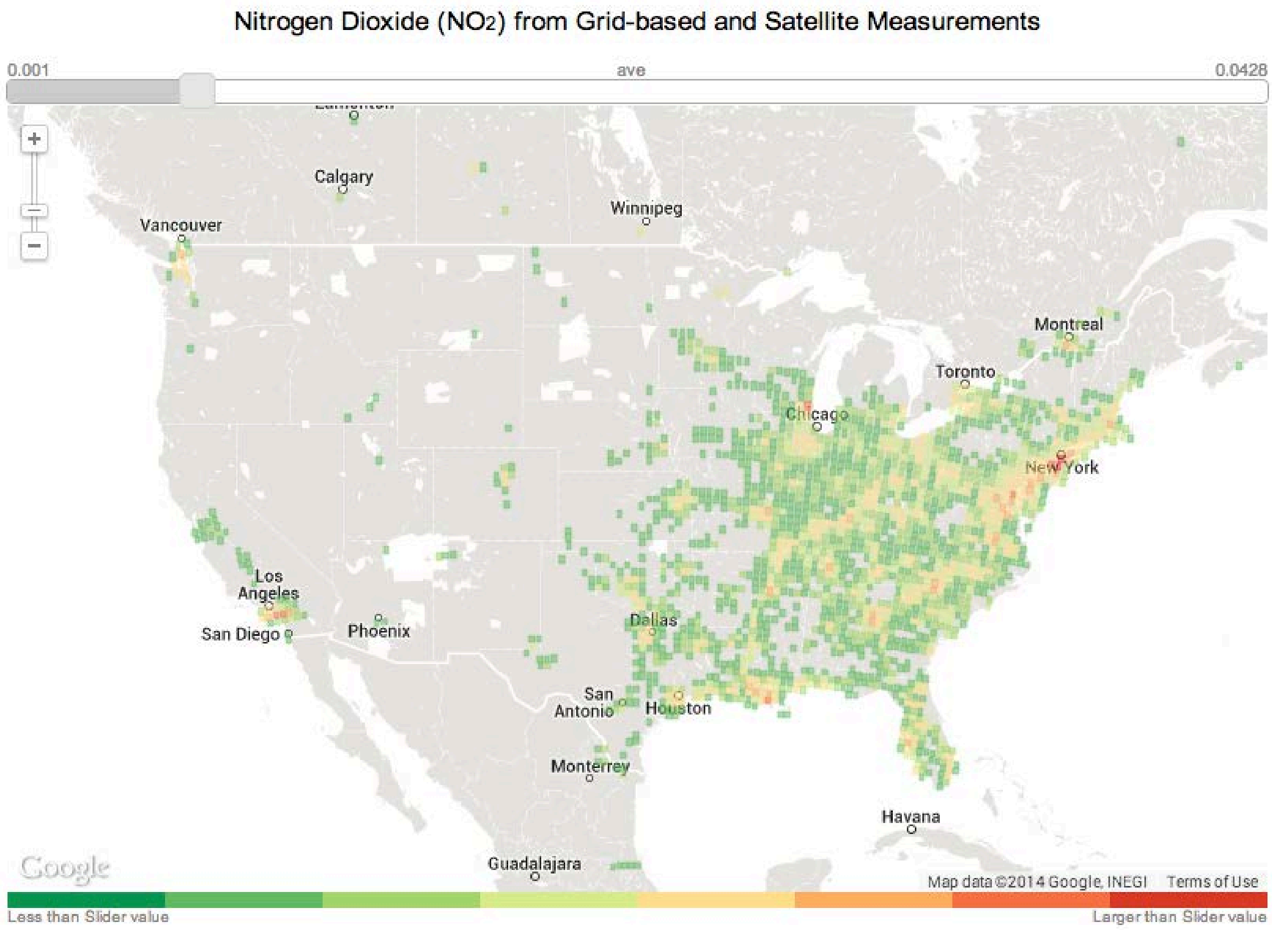


Figure 11. Grid-Based Satellite Measurement NO2

13. Aggregated State-level NO2

[http://smog.sage.wisc.edu/vis/gis/omi/annual/states.html]
[Location: /Library/WebServer/Documents/vis/gis/omi/annual/states.html]

The aggregated state-level NO2 graph uses Google Maps to visualize state-level NO2 emission. It supports a umber of features:

- Click any state on the map, and a pie chart will pop up to display the state's source distribution
- Move the slider to see more detail for each state.

The back-end technology of aggregated state-level NO2 includes JavaScript, Google Maps, Google Visualization Charts and HTML. To set up aggregated state-level maps:

- Create a mask file, which corresponds to the grid file (12*12km, 36*36km). The mask file tells how much each grid contributes to one county, and since counties are affiliated with states, you can aggregate the grid-level data into state-level data. This process is done in Java.
- Store the contour coordinates of the states on the map in a JavaScript file. Each state can be viewed as a polygon, and its contour coordinates data can be downloaded from the web (e.g. Google).
- When you click one state, it will generate a pie chart graph. This is done by Google Visualization Charts.

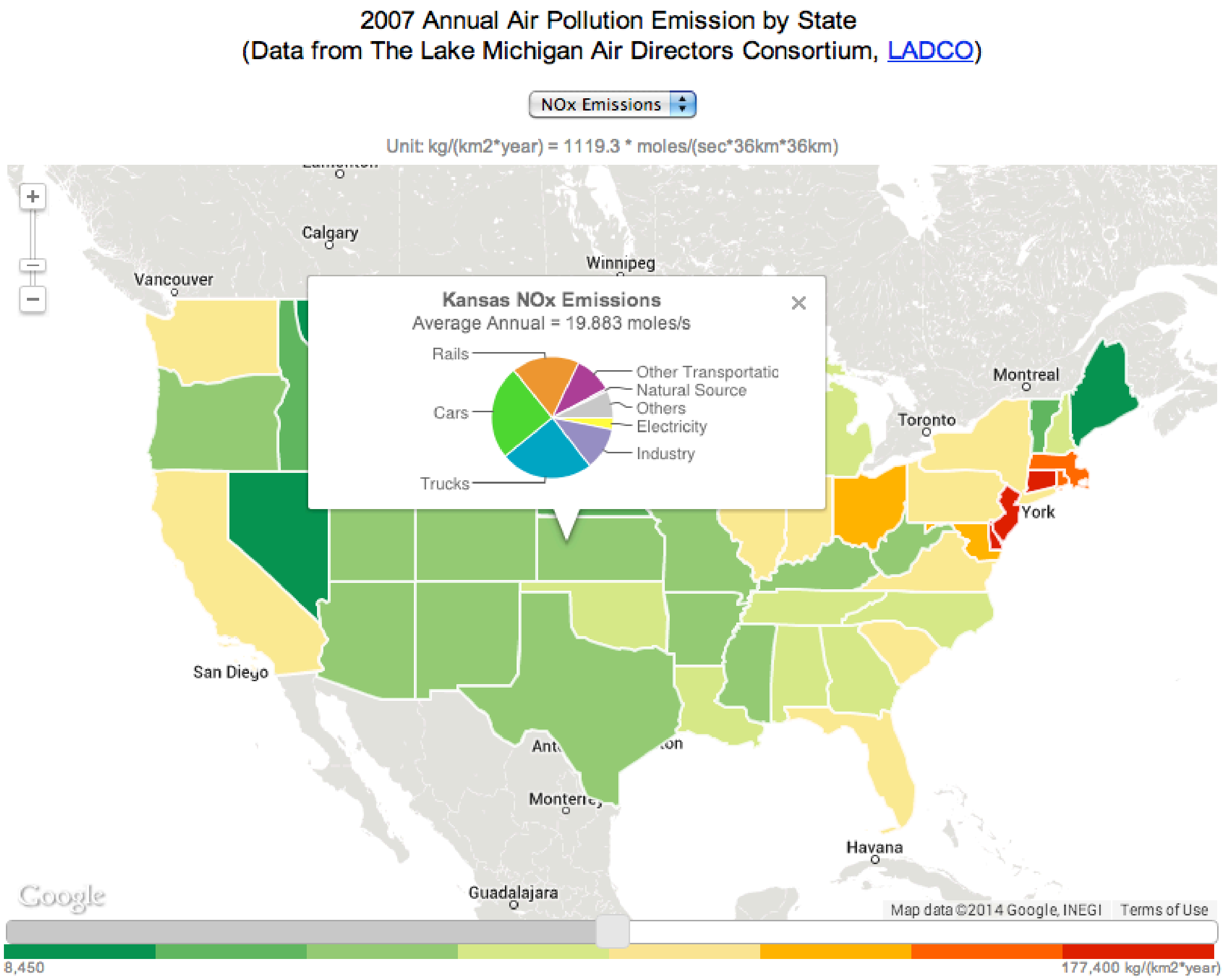


Figure 12. Aggregated State-level NO2

14. Aggregated County-level NO2

[http://smog.sage.wisc.edu/vis/gis/omi/annual/counties.htm]
[Location: /Library/WebServer/Documents/vis/gis/omi/annual/counties.html]

The aggregated county-level NO2 uses Google Maps to visualize county-level NO2 emission. It also supports such interactions, and it works much like the state-level map:

- Click any county on to pop up a pie chart displaying the source distribution for that county.
- Move the slider to can see the dynamic change of each county.

The back-end technology of aggregated county-level NO2 includes JavaScript, Google Maps, Google Visualization Charts and HTML.

- Create a mask file that corresponds to the grid file (12*12km, 36*36km). The mask file tells how much each grid contributes to each county, so you can aggregate the grid-level data into county-level data. This process is done in Java.
- Store the contour coordinates of each county in a JavaScript file; for the county, it can be viewed as a Pie-chart popups are done by Google Visualization Charts.

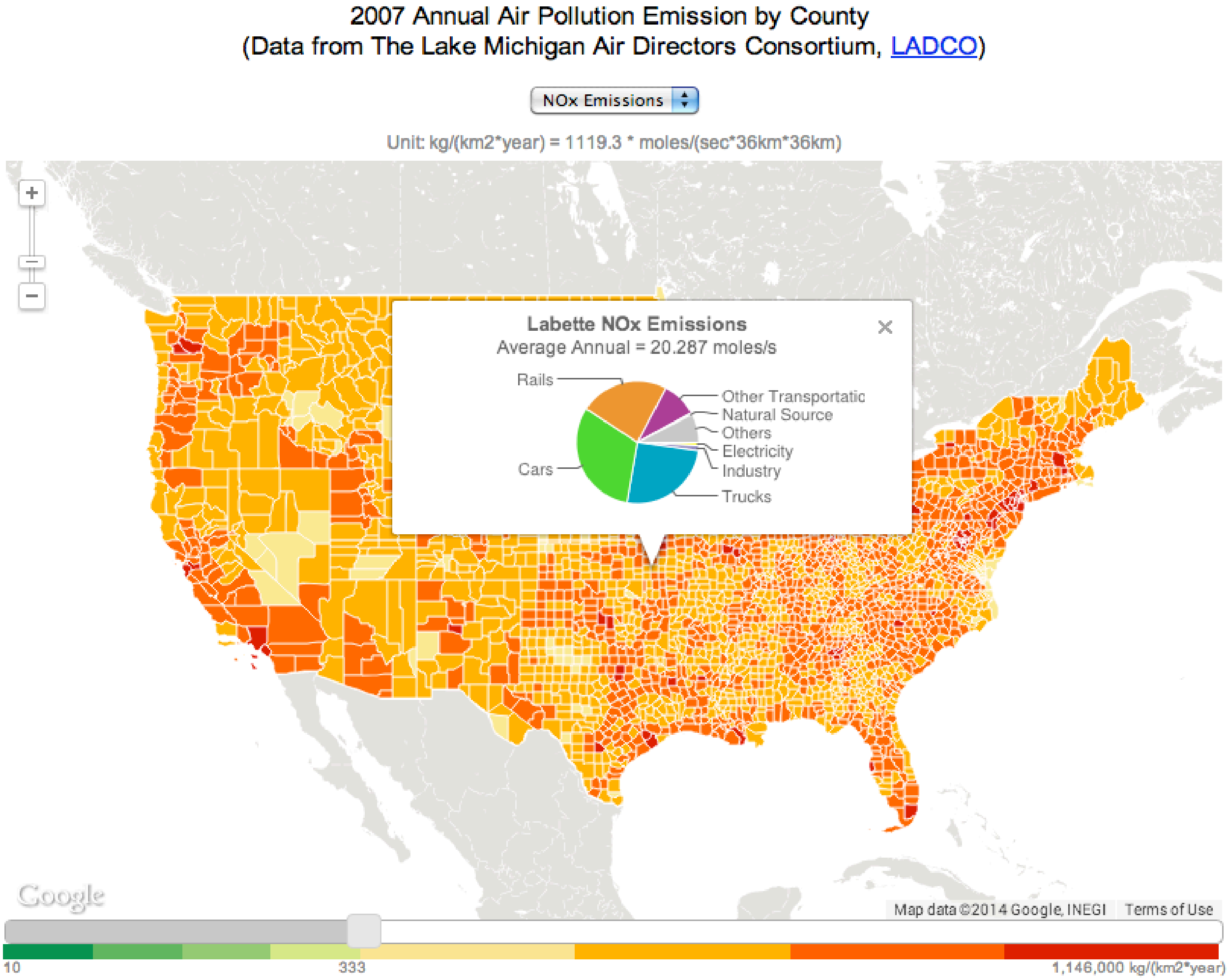


Figure 13. Aggregated County-level NO2

15. New-style Aggregated County-level NO2

[http://smog.sage.wisc.edu/vis/gis/omi/svg/counties.htm]
[Location: /Library/WebServer/Documents/vis/gis/omi/svg/counties.htm]

The new-style aggregated county-level NO2 graph uses D3 to visualize county-level NO2 emission. It is different from previous Google Maps-styled in these ways:

- It is easier to use and the programmer is given more freedom to organize the map.
- D3 has its own GIS-supported functions, and is lightweight compared to Google Maps.
- It can load more data, and is very fast compared to Google Maps.

The back-end technology of new-styled aggregated county-level NO2 includes JavaScript, Google Visualization Charts, D3, and HTML. To set D3 maps up you need to:

- Load libraries from D3.
- Organize the input data into a Json structure, which is a lightweight data exchange format for web application. This is done in Java.
- When the mouse is moving or hovering over one county, the corresponding county will be highlighted.

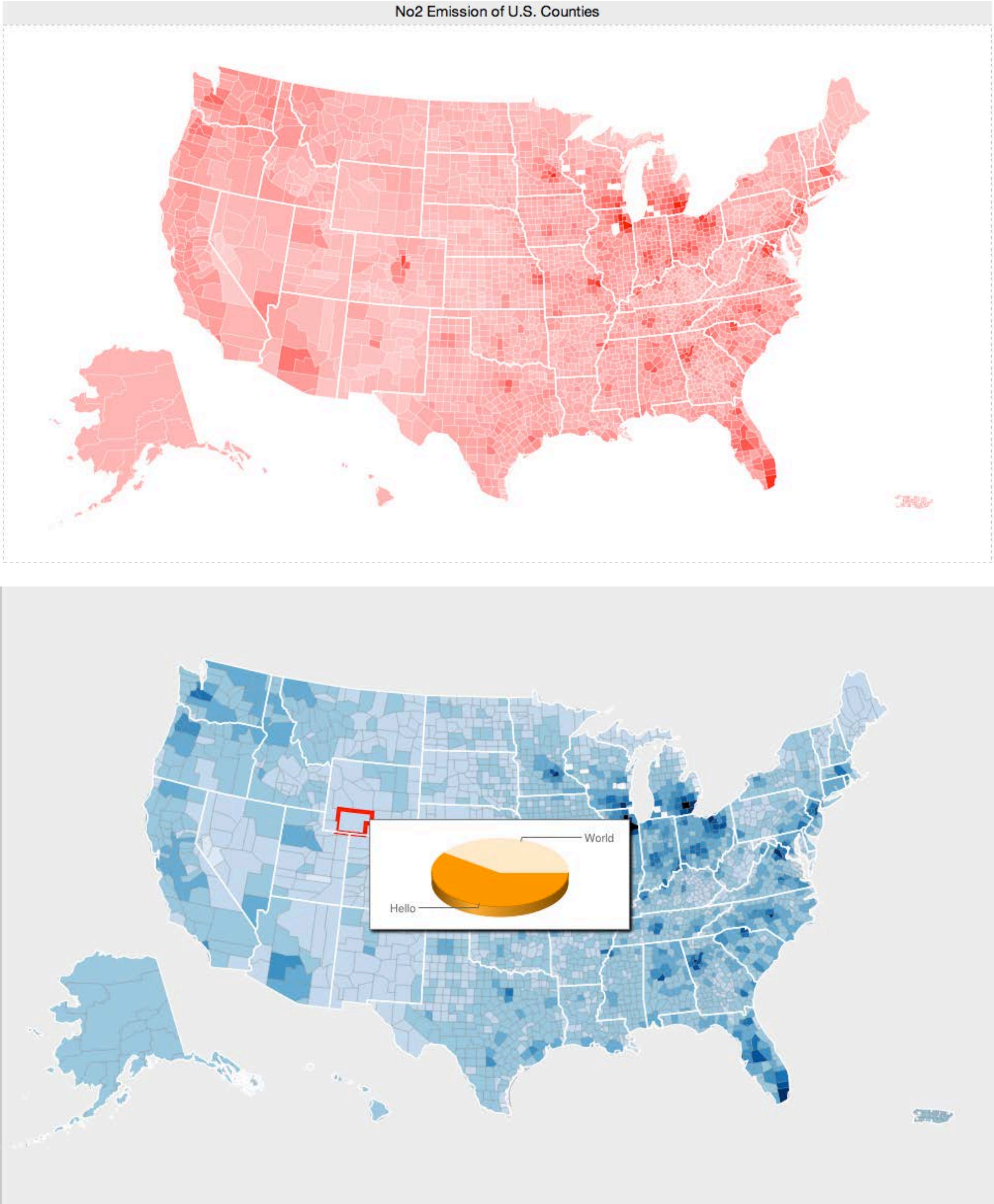


Figure 14. New-styled Aggregated County-level NO_2

16. More Styles

[Location: /Library/WebServer/Documents/vis/gis/omi/svg/multimaps.htm]

Mr. Li tried a few additional of web geographic visualization:

National NO_2 Emission in 2007

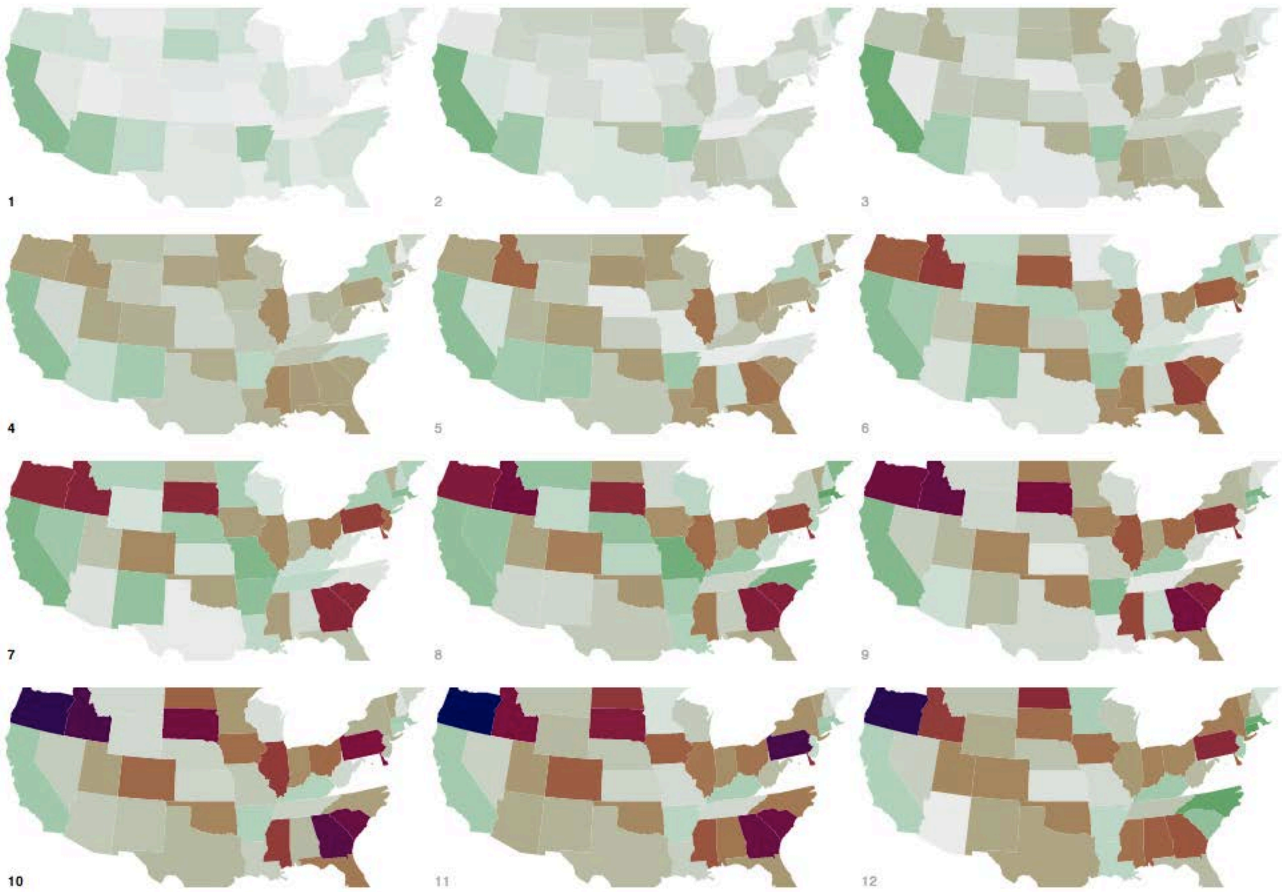


Figure 15. State-level NO2 Emission in 2007 over 12 months
[http://smog.sage.wisc.edu/vis/gis/omi/svg/multimaps.htm]

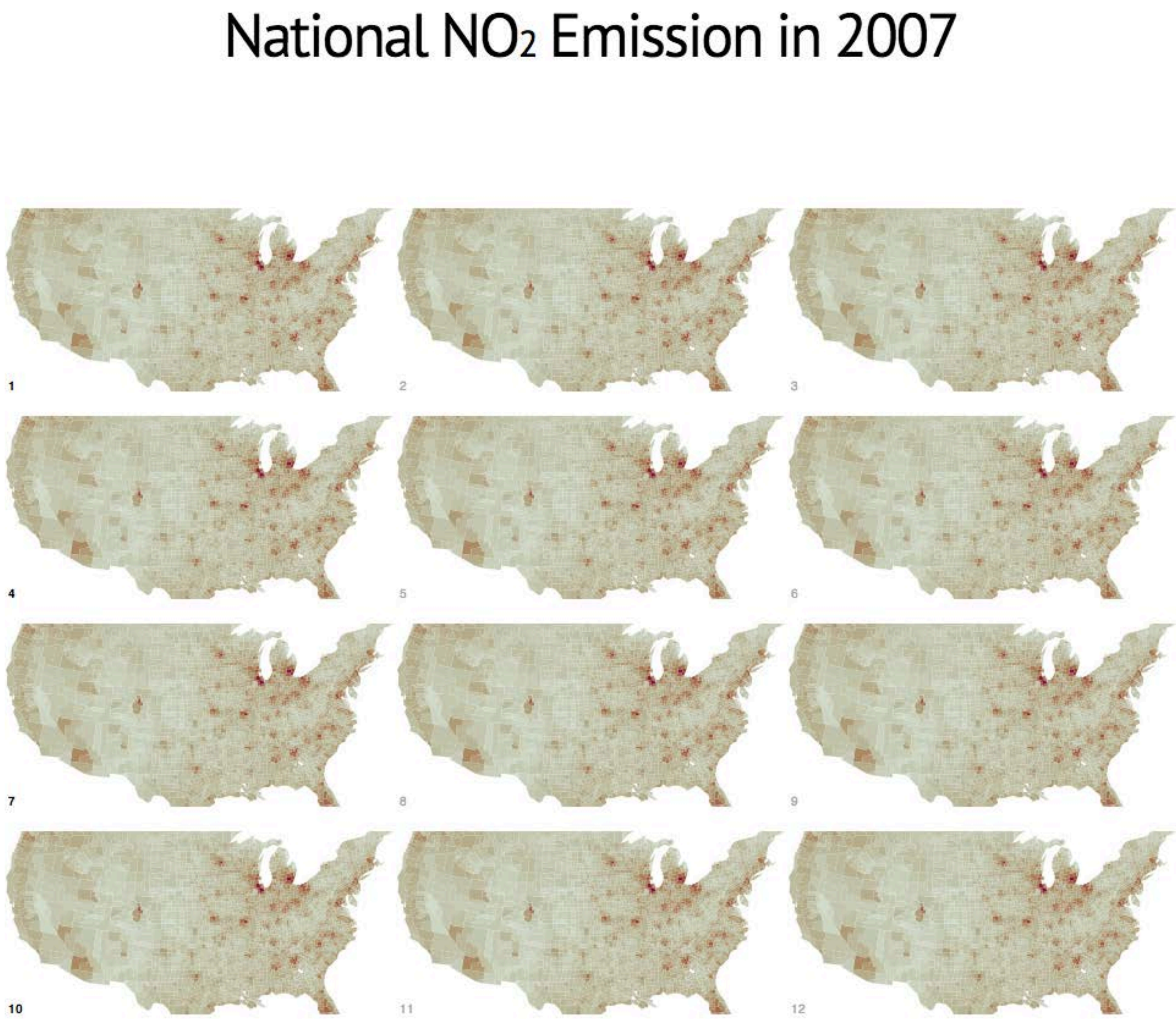


Figure 16. County-level NO2 Emission in 2007 over 12 months
[http://smog.sage.wisc.edu/vis/gis/omi/svg/multicounty.htm]

Emission Timeline

[http://smog.sage.wisc.edu/SAGEvis/timeline.html]
[Location: /Library/WebServer/Documents/SAGEvis/timeline.html]

The emission timeline is a poster visualization, showing the trend of multiple emissions (CO, NOx, PM10, PM2.5, SO2, VOCs, NH3) every five years from 1990 – 2010. It allows the user to select a pollutant and year to add a pie chart to the web page. By adding additional pie charts, the timeline allows the user to make a horizontal comparison over time for one pollutant, as well as vertical comparisons among different pollutants. Additional features include customizable color representation for emission sources, and fun facts from different categories such as Basics, Policy, Transportation, Electricity and Cool Science.
The data is from U.S. Environmental Protection Agency NEI (http://www.epa.gov/ttnchie1/trends/).

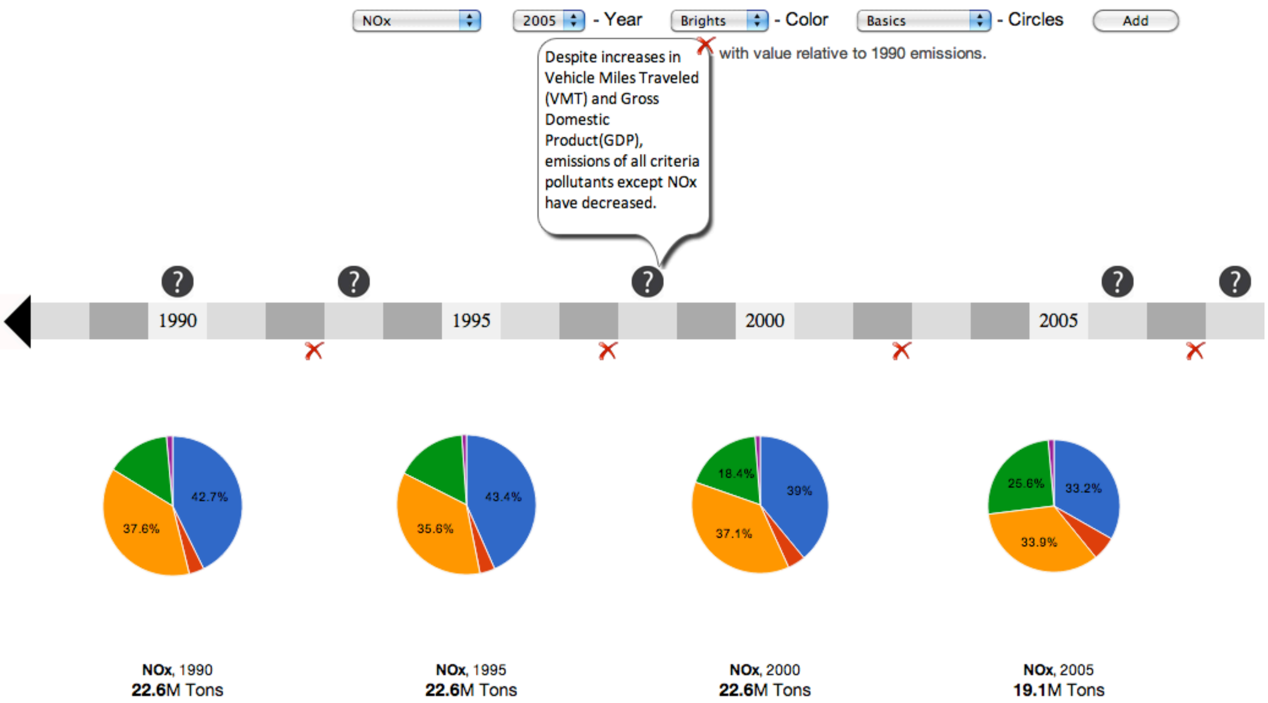


Figure 1. Emission timeline

The back-end technology of Emission Timeline includes Javascript, Google Visualization Charts, CSS and HTML. To set the timeline up:

- Aggregate the metadata (for example, which emission source is for combustion, which contributes to the industrial category etc.), then re-organize these data into a Javascript file. This part of the work is done in Java.
- In the HTML file, load the Javascript data file at the head section.
- The size of pie chart is proportional to its total emission at a specific year for one pollutant.
- The user can select the different color theme for representation; or the user can switch the colors in one theme.

NASA Slider Images

[http://smog.sage.wisc.edu/SAGEvis/nasaslider.html]
[Location: /Library/WebServer/Documents/SAGEvis/nasaslider.html]

The NASA Slider Images is an interactive image slider that allows the user to compare two similar images by moving the slider left or right.

The back-end technology of NASA Slider Images includes Javascript, CSS and HTML. If you want to load your own images for the slider, simply revise the image path to point to your images in the scripts.

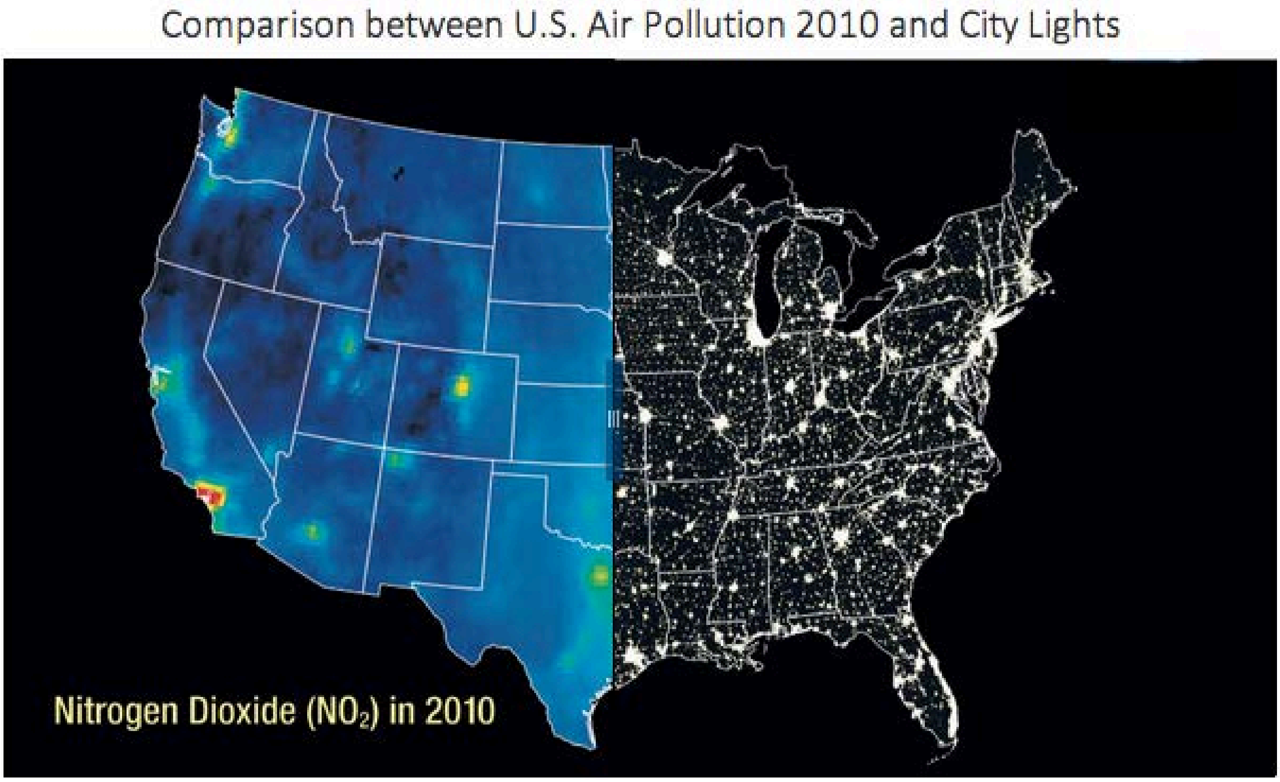


Figure 2. NASA Slider Images

MidWest Data Vis

[http://smog.sage.wisc.edu/SAGEvis/midwest.html]

[Location: /Library/WebServer/Documents/SAGEvis/midwest.html]

The MidWest Data Vis is a geographic visualization focusing on emissions (left) and concentrations (right) for the counties in the Midwest for January and July of 2005. Emissions data comes from the Lake Michigan Air Director's Consortium (LADCO). The concentration data was generated from the Community Multiscale Air Quality Model, v4.6, using LADCO emissions and meteorology from the Weather Research and Forecasting (WRF) model. A unique transportation emissions dataset, the Wisconsin Inventory of Freight Emissions (WIFE), was developed to study emissions from transportation in detail. Two modal shift scenarios were analyzed, Intraregional Freight transport and Through-Freight transport. The Through-Freight scenario shifts emissions from on-road diesel vehicles (trucks) to rail when freight is transported through the Midwestern U.S. Here, Through-Freight data for the concentration of NO2 under the base case in January, 2005 is presented.

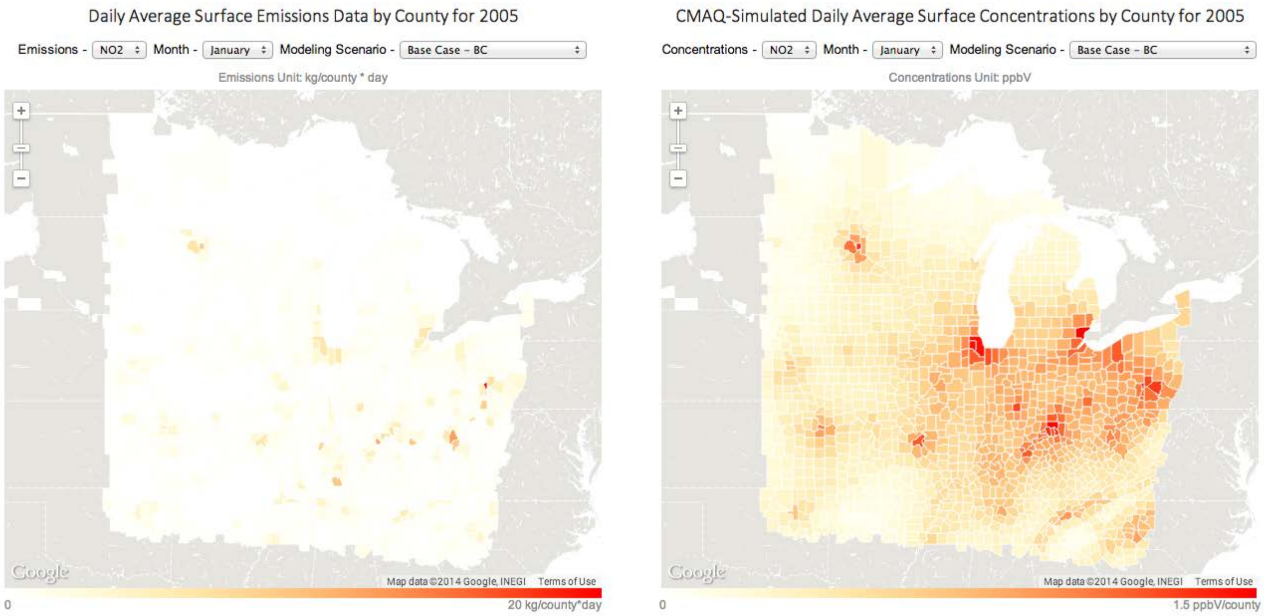


Figure 4. MidWest Data Vis

The back-end technology of MidWest Data Vis includes Javascript, Google Maps, HTML, CSS. To set MidWest Data Vis up:

- Use CSS to customize the Google Map from its traditional style to an aesthetic web map. Take advantage of the GIS functions provided by Google Maps, though there are many parameters to consider customizing.
- Since MidWest Data Vis is county-level, store the coordinates of the contours of all the counties. Then load these data as polygons into a layer of Google Maps. The contour coordinates of the U.S. counties can be downloaded from web (e.g. Google).
- Here, the raw data is gridded-based according to the netCDF files. But you'll need a process to extract the raw data from the netCDF files and aggregated it into a county-level and the re-organize the data into a Javascript file, so that the HML file can load the data (including contours of the counties).
    - How to aggregate the gridded-based data into county-level? It depends on the grid file. For example, if it is a 12km*12km file, you also need a corresponding mask file (based on your grid file) to map each grid.

This page intentionally left blank.

CFIRE

University of Wisconsin-Madison
Department of Civil and Environmental Engineering
2205 Engineering Hall
1415 Engineering Drive
Madison, WI 53706
Phone: 608-263-9490
cfire.wistrans.org