Dynamic and interactive vector maps for the new web

Sergio Álvarez Leiva // @saleiva // vizzuality.com

Dynamic and interactive ~~vector~~ maps for the new web

I’m Founder and Lead Designer at **vizzuality**

Lead Designer means “avoid really hard questions please” :P

We work on **stories that matter**

The world through 3054 lenses.

An online collaborative effort to protect global linguistic diversity.

Start exploring

endangeredlanguages.com

Koro

threatened

Koro is a language previously unknown to science that was documented in the mountains of northeast India. It is spoken by no more than 4000 people.

Photograph of Abamu Degio singing in Koro. Provided by Living Tongues Institute for Endangered Languages, filmed by Jeremy Fahringer.

US

USRC Bear

US Revenue Cutter BEAR - steam sailing ship, built in 1874 and afloat for 89 years, served in two World Wars, and spent 40 years sailing in Bering Sea and Alaskan waters on a variety of duties.

Old Weather: Our Weather's Past, the Climate's Future

Introduction

Help scientists recover Arctic and worldwide weather observations made by United States' ships since the mid-19th century. These transcriptions will contribute to climate model projections and will improve our knowledge of past environmental conditions. Historians will use your work to track past ship movements and tell the

Project Statistics

Old Weather transcriptions so far

Zürich-Obersee: Guntliweid bis Bätzimatt (SZ) Reserves for Waterbirds and Migrants of International and National Importance

Zürich-Obersee: Guntliweid bis Bätzimatt (SZ), Switzerland



All Photos are provided by Panoramio. Photos are under the copyright of their owners.

1 2 3

Description

Official Record

81% COMPLETE

WDPA ID
178706

NAME
Zürich-Obersee: Guntliweid bis Bätzimatt (SZ)

ORIGINAL NAME
Zürich-Obersee: Guntliweid bis Bätzimatt (SZ)

COUNTRY / TERRITORY
CHE

EB Browsers & Technologies The Growth of the Internet +1

2005 2006 2007 2008 2009 2010 2011 2012

XMLHTTPRequest2

Offline Web Apps:AppCache

Drag & Drop

CSS3 2D Transforms

CSS3 3D Transforms

Web Workers

CSS3 Animation

Date/time input types

Touch Events

Content Security Policy

Full Screen API

8 8.1 9

8 8.5 9 9.1 9.2 9.5 9.6 10 10.1 10.5 10.6 11 11.1 11.5 11.6 12

7 8 9

2 3 3.1 4 5 5.1

1 1.5 2 3 3.5 3.6 4 5 6 7 8 9 10 11 12 13

Chrome

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Audio & Video elements

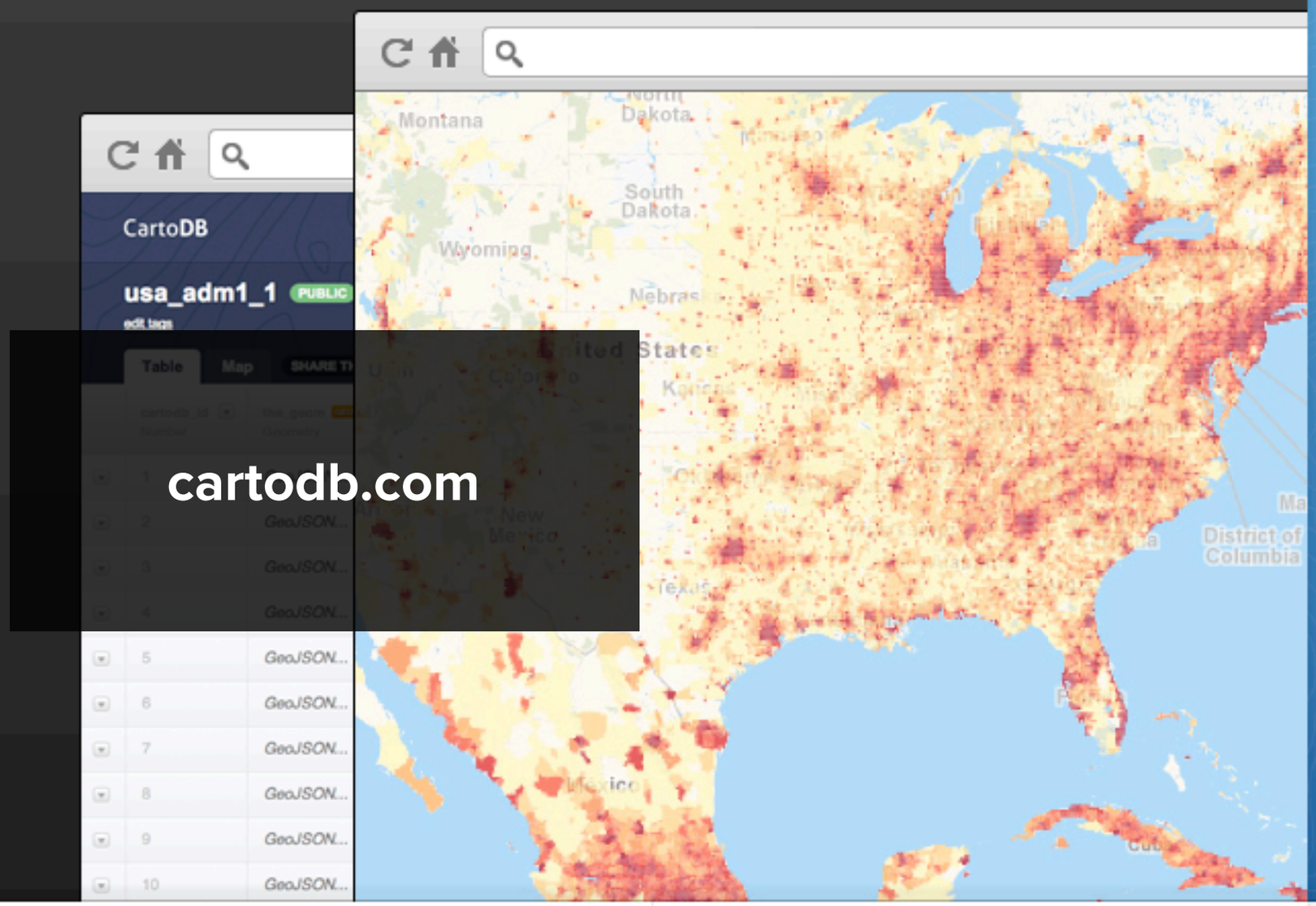
CSS3 Gradients

CARTODB

TOUR MAPS DEVELOPERS PRICING SIGN IN

Map, analyze and build applications with your data.

Sign up now
- it's free! -



Create dynamic maps with ease.

Upload and visualize your data within minutes, and share your stories on the internet.

Analyze your data with the power of PostGIS.

Use SQL based spatial operations to tell stories and generate new insights and powerful analysis.

Build location aware applications.

Our rich APIs reduce development time for building web and mobile applications.

Advanced mapping techniques we've used **real projects**

Some hacks we've done in **real projects**

Visualizing deforestation alerts **over time**

PLEASE NOTE THIS IS A PROTOTYPE. Read more

Global Forest Watch 2.0
A project by WRI

South America

Seach for a country

Sto

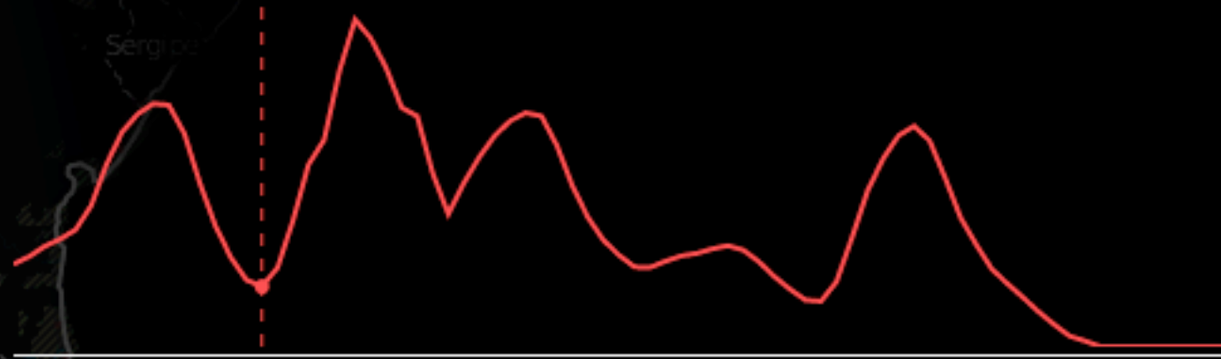
Maranhão
35704 hotspots this month

+ - Fullscreen map

Amapá
Pará
Piauí
Tocantins
Bahia
Goiás
Espírito Santo
São Paulo
Rio de Janeiro
Paraná
Pernambuco
Alagoas
Sergipe

Brazil

0% increase in May 2007
4% increase in the last six months



BRAZIL'S SITUATION

Due to the vastness of the Amazon rainforest, Brazil's average loss of
square kilometers of primary forest per year between 2000 and 2005
represents only about 0.8 percent of its forest cover. Nevertheless,
deforestation in Brazil is one of the most important global environme
issues today. Brazil holds about one-third of the world's remaining
rainforests, including a majority of the Amazon rainforest. It is also
overwhelmingly the most biodiverse country on Earth, with more thar
56,000 described species of plants, 1,700 species of birds, 695 amphi

MAY 2007

2006 2007 2008 2009 2010 2011

The challenges

Process and visualize millions or rows.

Show changes in data over time.

Visualize up to 5 layers of information.

How we did it

We host and process the data with CartoDB.

We generate a JSON file per tile containing the deforestation data codified in cells (each cell has a value for each time period - 15days).


```
{
  time: 0.00576543211,
  rows: [
    {
      upper_left_x: 16742472,
      upper_left_y: -2589569,
      time_series: [
          0,
          8,
          8,
          8,
          8,
          12,
          12,
          12,
          16,
          ...
```

How we did it

Load all the information of the JSON files in memory using TypedArrays.

Render the tiles in the client using the codified information -now in memory-. Each time the user moves the timeline, the tiles are rendered again.

```
{
  time: 0.00576543211,
  rows: [
    {
      upper_left_x: 16742472,
      upper_left_y: -2589569,
      time_series: [
          0,
          8,
          8,
          8,
          8,
          12,
          12,
          12,
          16,
          ...
```

How we did it

Load all the information of the JSON files in memory using TypedArrays for better performance.

Render the tiles in the client using the codified information. Each time the timeline is moved, the tiles are rendered again.

```
{
  time: 0.00576543211,
  rows: [
    {
      upper_left_x: 16742472,
      upper_left_y: -2589569,
      time_series: [
          0,
          8,
          8,
          8,
          8,
          12,
          12,
          12,
          16,
          ...
```

How we did it

Load all the information of the JSON files in memory using TypedArrays for better performance.

Render the tiles in the client using the codified information. Each time the timeline is moved, the tiles are rendered again.

```
{
  time: 0.00576543211,
  rows: [
    {
      upper_left_x: 16742472,
      upper_left_y: -2589569,
      time_series: [
          0,
          8,
          8,
          8,
          8,
          12,
          12,
          12,
          16,
          ...
```

How we did it

We process the boundary data in CartoDB and merged different tables in the same layer for improving the performance of the map when panning an zooming.

We don’t use UTFGrids for the interactivity layer on the boundaries. We render the region polygons with different colors in an invisible canvas and map those colors to the region_id. Getting pixel information with JS is really fast.

This is pretty sweet stuff

http://goo.gl/KXIF1

Detect new deforested areas on the client using satellite imagery

Deforestation Validation Tool

powered by Google Earth Engine

Andrew Hill
12 cells this month

Cell JG.65 - go back 2 NOTES

65 km² analysed DONE

DEF DEG

Compare view

NDFI settings

Ranges value 72 72

NDFI visibility

- Degradation
- Deforestation
- Forest
- Past deforestation

The challenges

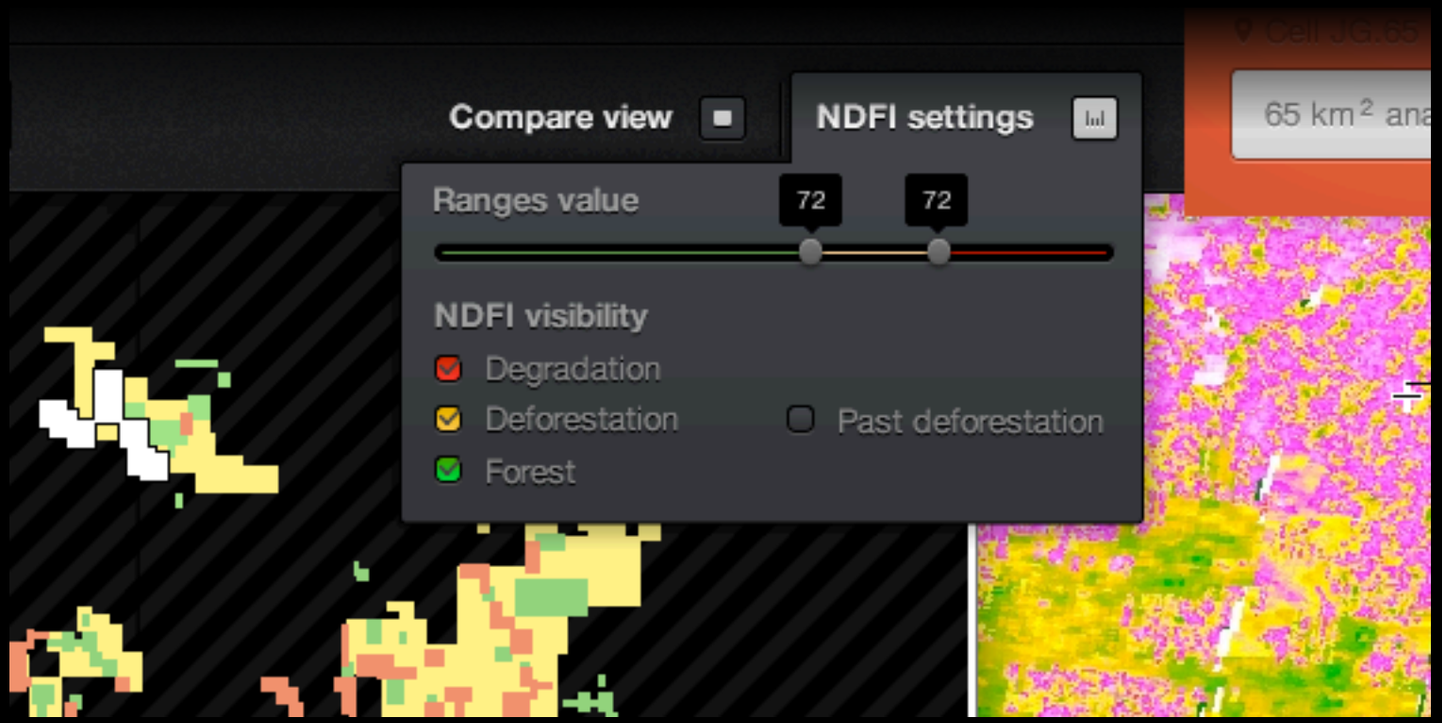
Detect deforested areas automatically depending on a dynamic range of values.

Allow humans to confirm / reject deforestation on this areas.

How we did it

Information comes encoded as tiles rendered with just one channel - grayscale.

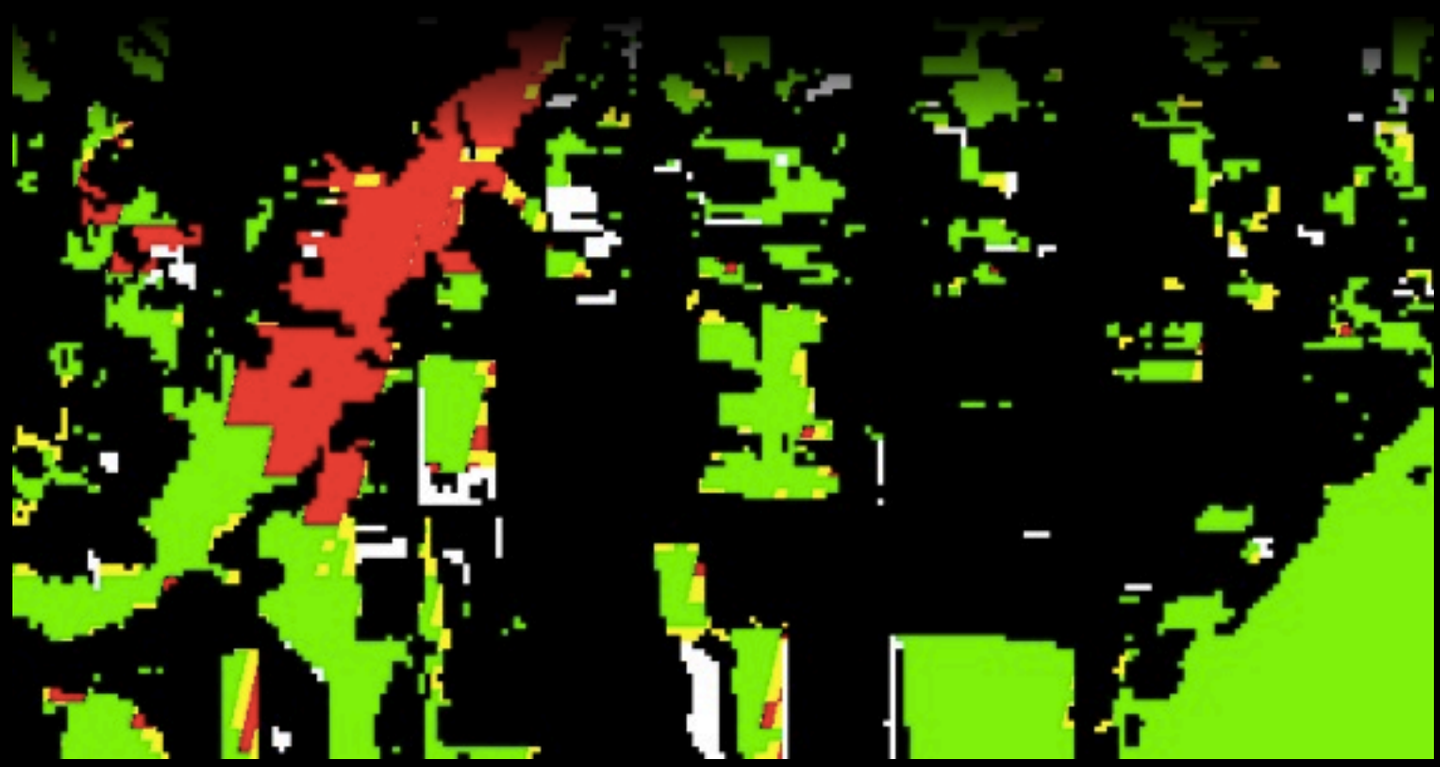
We render new tiles using canvas and assign colors to the different ranges of values determined by the user which allows us to see how the different areas change in real-time.



How we did it

We proved a magic-wand functionality that allows the users to create a polygon over an area with just a click.

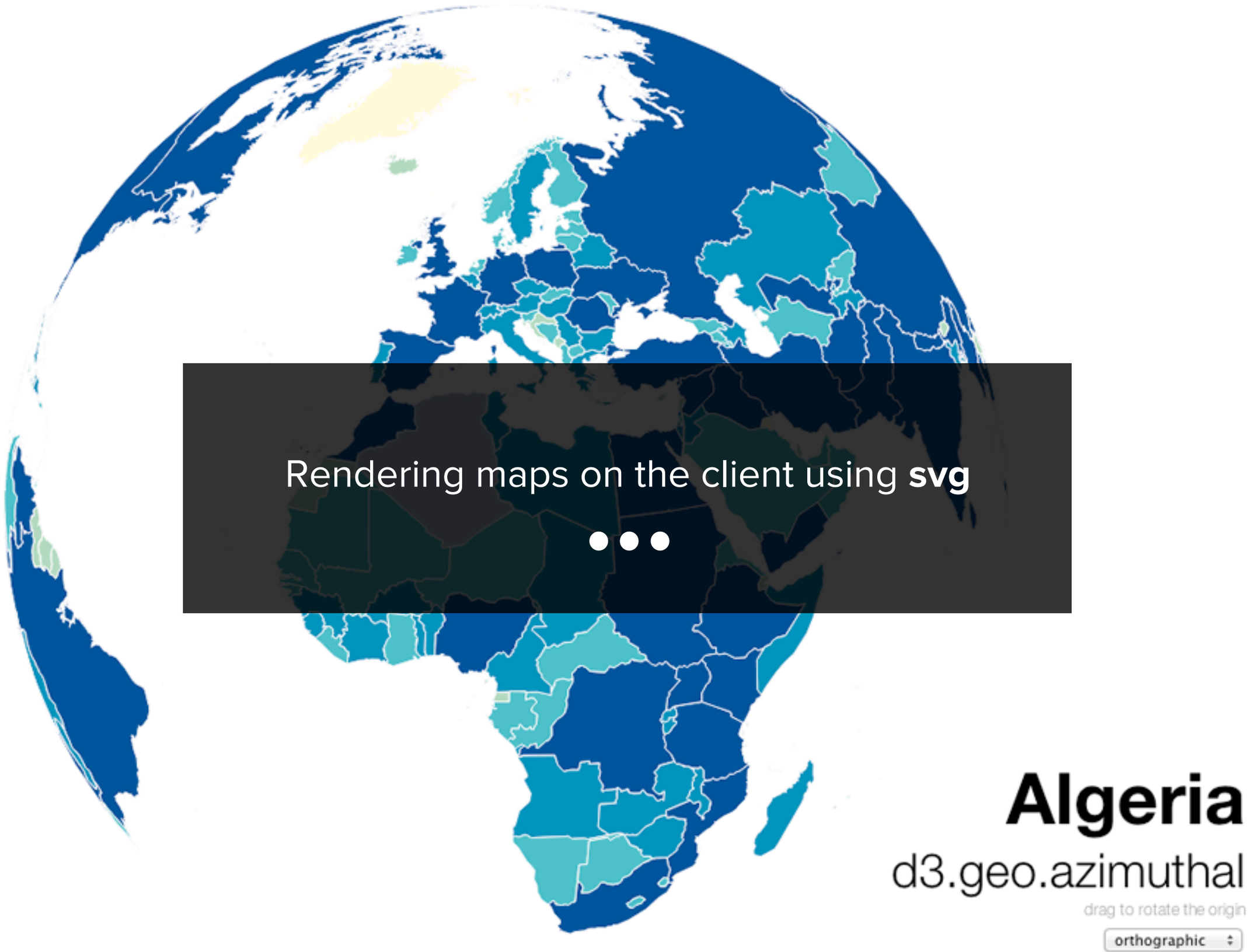
It just detect adjacent pixels with similar colors and draws a polygon round them using several simplifying algorithms.



What else?

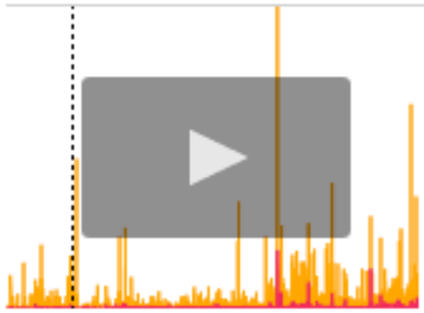
Rendering maps on the client using **svg**

• • •



Syria deaths

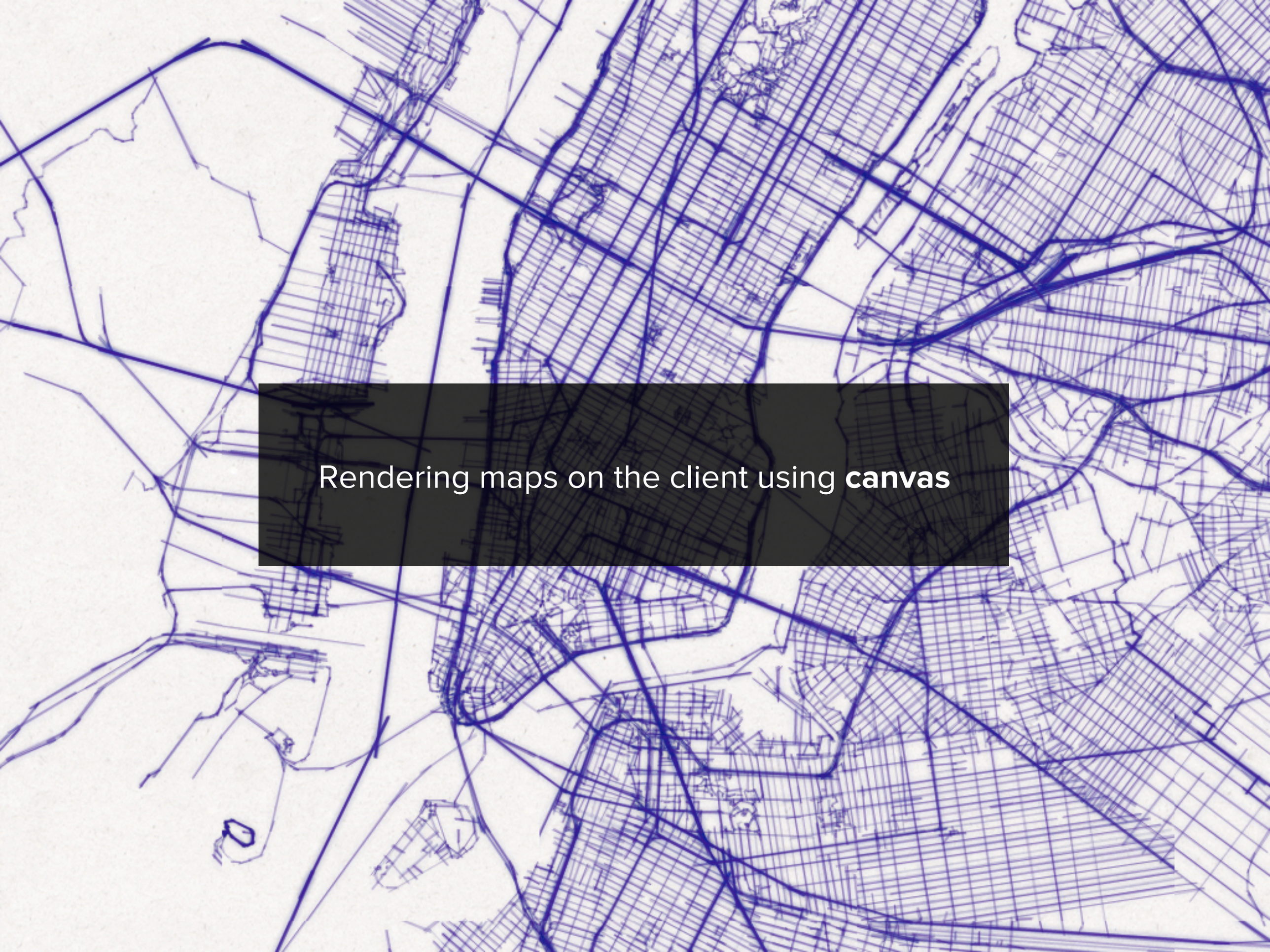
02/06/2011



1463 | 62 | 1525

- adult deaths
- child deaths <18 yrs
- total

Rendering maps on the client using **svg**



Rendering maps on the client using **canvas**

```
#roads {
  [zoom > 12] {
    line-color: #555;
    line-width: 0.3;
  }
  [type='motorway'],[type='motorway_link']
{
    line-width: 2;
  }
  [type='trunk'], [type='trunk_link'] {
    line-width: 5;
  }
  [type='primary'],   [type='primary_link']
{
    line-width: 5;
  }
  [type='secondary'],
[type='secondary_link'] {
    line-width: 2;
    line-color: #222;
  }
}
```

Rendering maps on the client using **canvas**

	total	avg	max	min
conversion_time	544.0	13.6	30.0	6.0
vertices	65972.0	1649.3	2755.0	864.0
primitive_count	25588.0	639.7	1120.0	297.0

You can find all this stuff at

vizzuality.com
https://github.com/Vizzuality/

cartodb.com
https://github.com/organizations/CartoDB
http://vizzuality.github.com/HTML5-experiments/
https://github.com/vizzuality/VECNIK

Thanks.