# Google Earth Tour Building

## For fun and profit

Joshua Tolley

End Point Corp.

June 7, 2012

# Whither the Data?

Everyone has geographic data

- Who browsed my website, from where? (GeoIP)
- Where do I ship most of my orders
- What, in fact, are the migration patterns of African and European swallows?

**Universal Truth**

Although you may have geospatial data, those data may be uninteresting.

# Getting Started

The process is pretty simple:

- Determine how you want to display the data
- Convert data to latitude and longitude (called "geocoding", this is often the most difficult part)
- Write a script to loop through the data points and display them (or do it by hand, if you're into that...)
- Celebrate

# What's Google Earth?

Google Earth is essentially Google Maps in 3D, on steroids.

# What's Google Earth?

- Desktop application for Linux (and Mac and Windows, if you insist)
- Free, but not open source ;(
  - This has caused us some problems, like getting bugs fixed
- There exists a paid "professional" version
  - Allows rendering of video, more flexible editing of large data sets, and a few other things
- Also exists in browser plugin form with JavaScript control; ostensibly this works only on Mac and Windows, though we're working to change that
- Began as a project by Keyhole software, which Google eventually bought

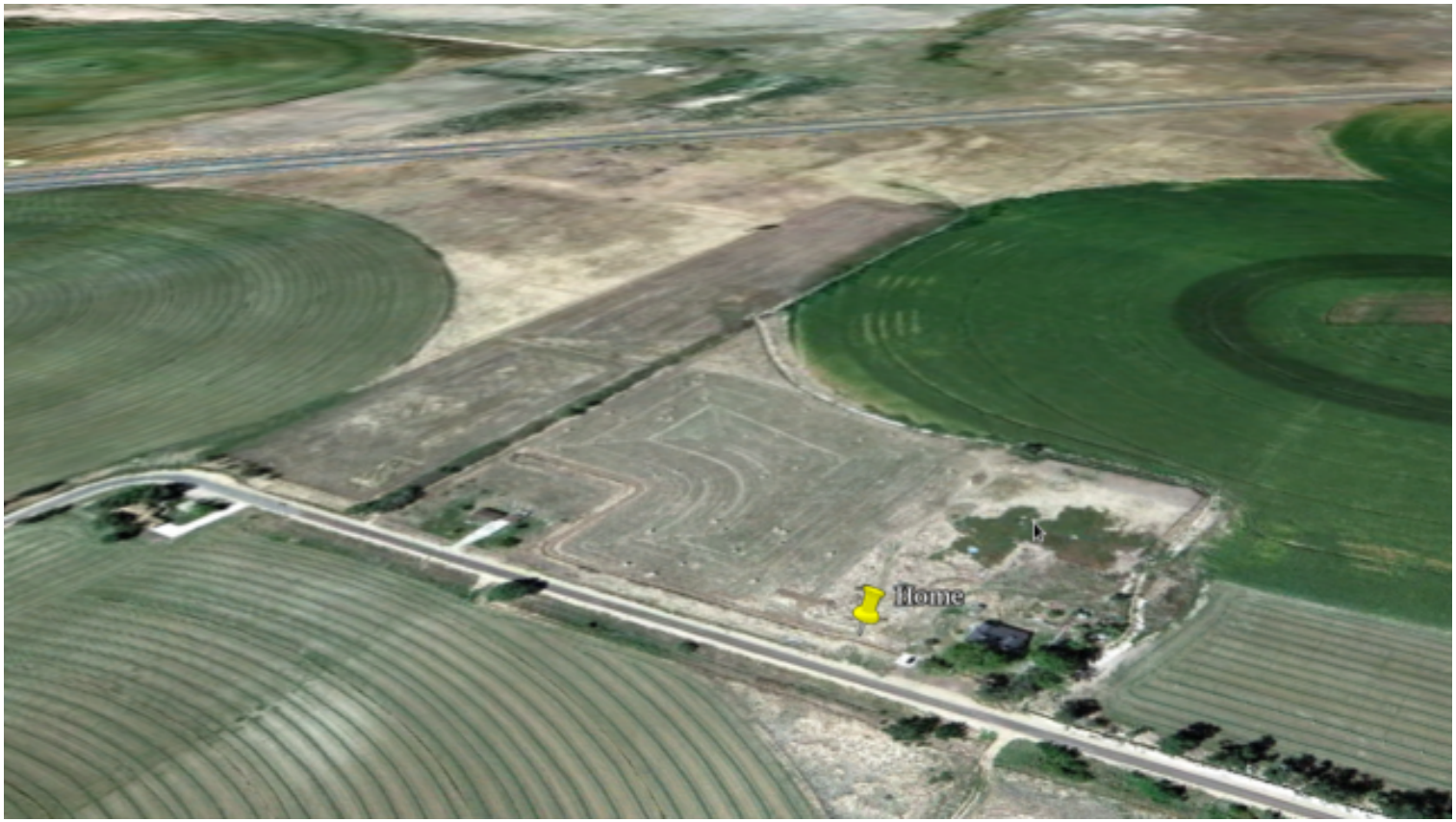
# How to use Google Earth

Google Earth accepts files written in Keyhole Markup Language

- XML-based
  - Mix between declarative XML, like XML Schema or XML config files, and procedural, like XSLT
- Can be created automatically through Google Earth. This is slow and inflexible
- Can be written by hand. This is like chewing glass
- Can be generated by various helper projects
  - Kamelopard: Ruby-based. I wrote most of it, and use it a lot.
  - PyKML: Python-based. More polished and consistent, but seemingly less capable than Kamelopard

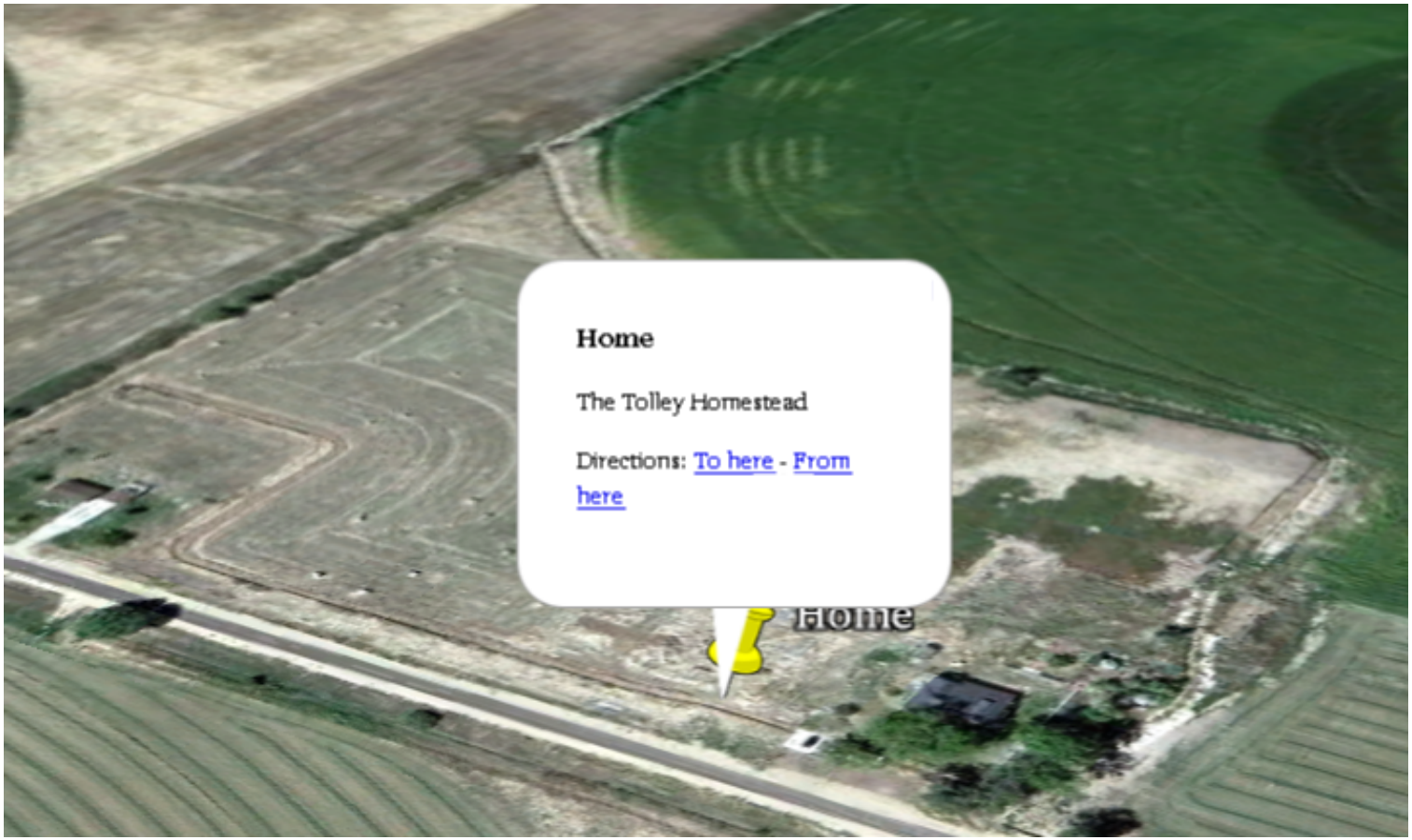
# An Example

I live on a small farm, where we are growing 12 acres of wheat and raising various poultry. It's here. This is a KML Placemark.



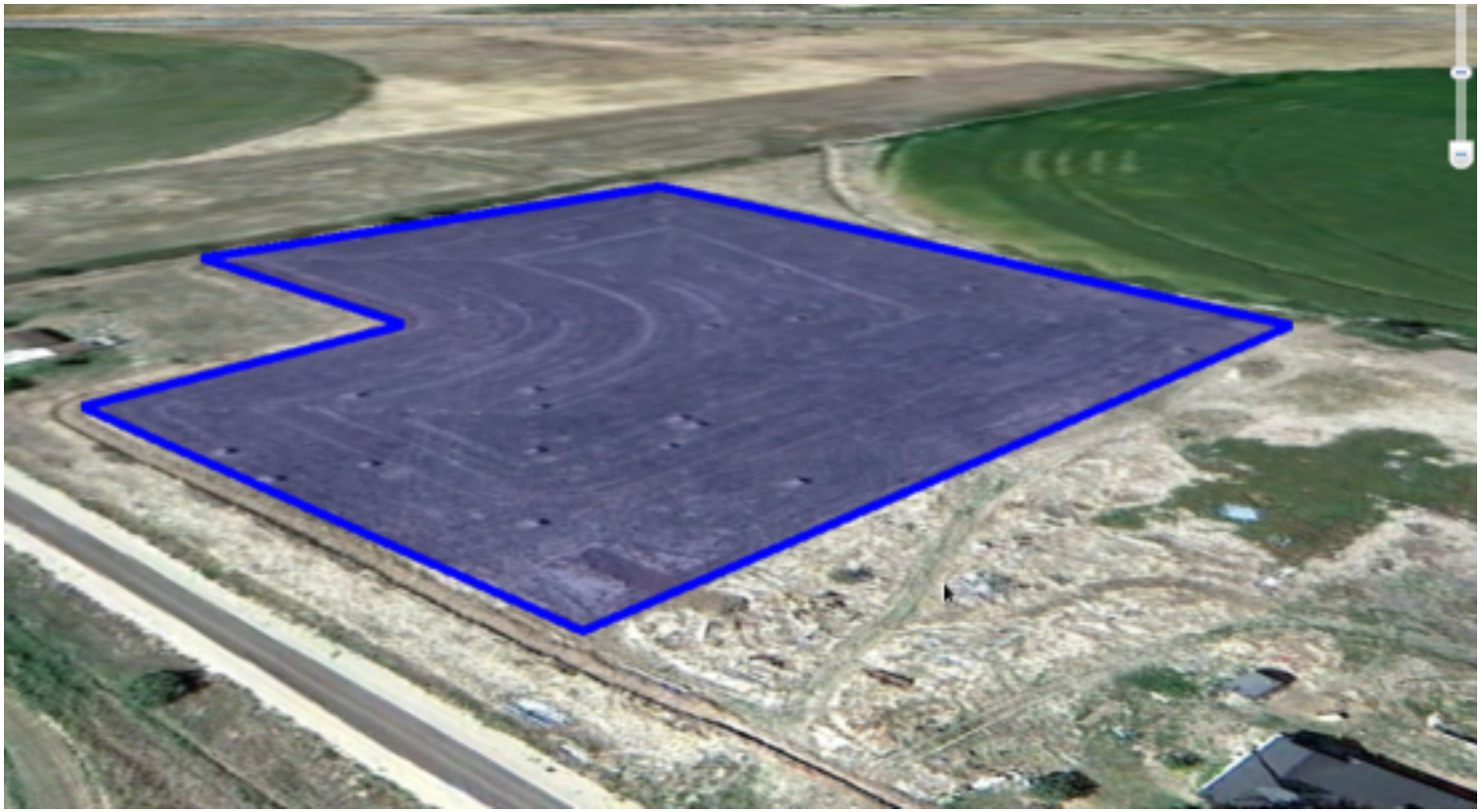
# An Example

These placemarks have descriptions, which can pop up in balloons, like this one. This can include CSS, images, or even Flash video. Icons, text, and balloons can all be styled at will.



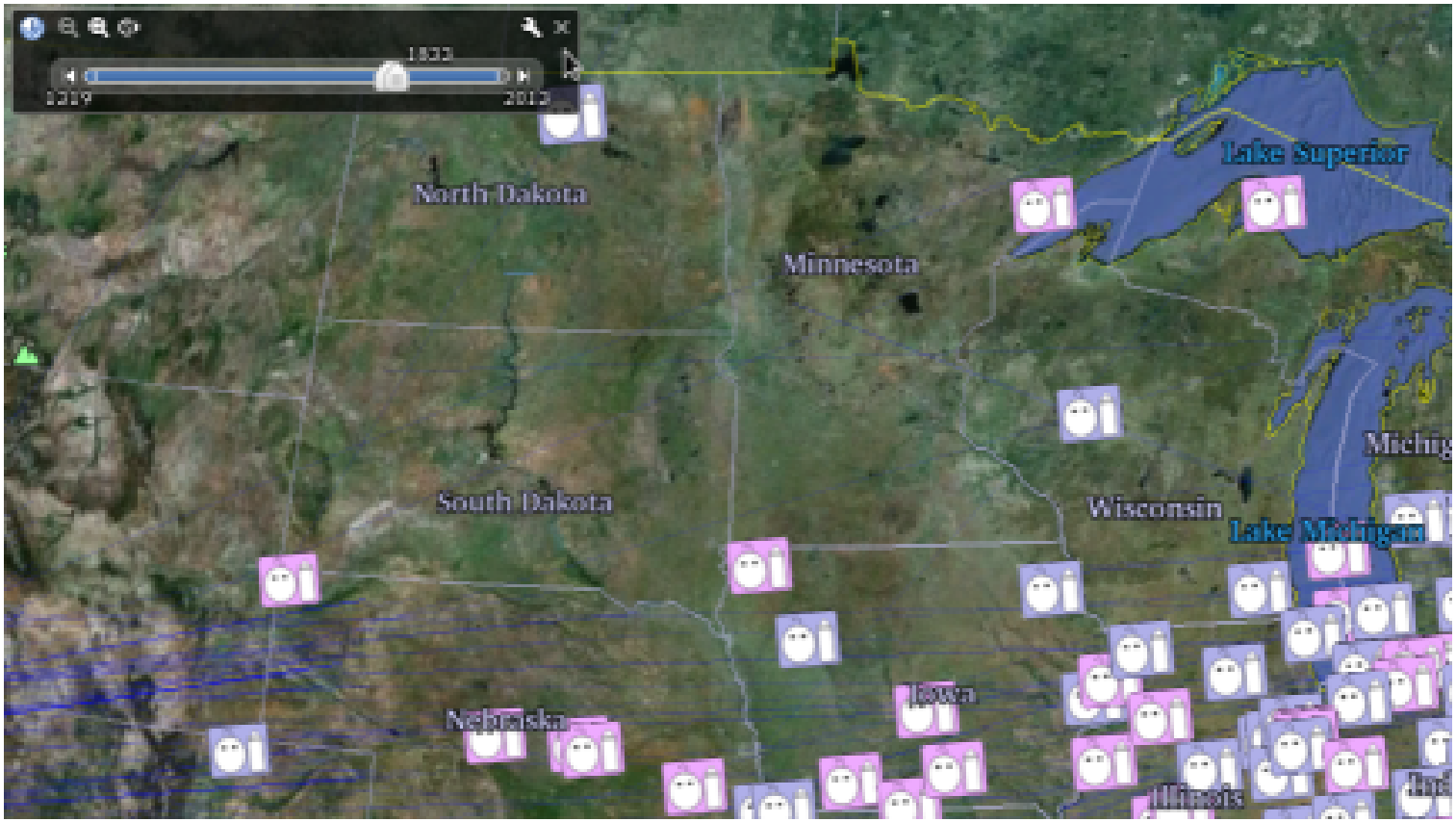
# An Example

As I said, we're growing wheat this year. This shows the wheat field, outlined with a KML polygon. KML also allows other objects, like lines and 3D models, in various styles.



# An Example

Some of these KML objects can include time data

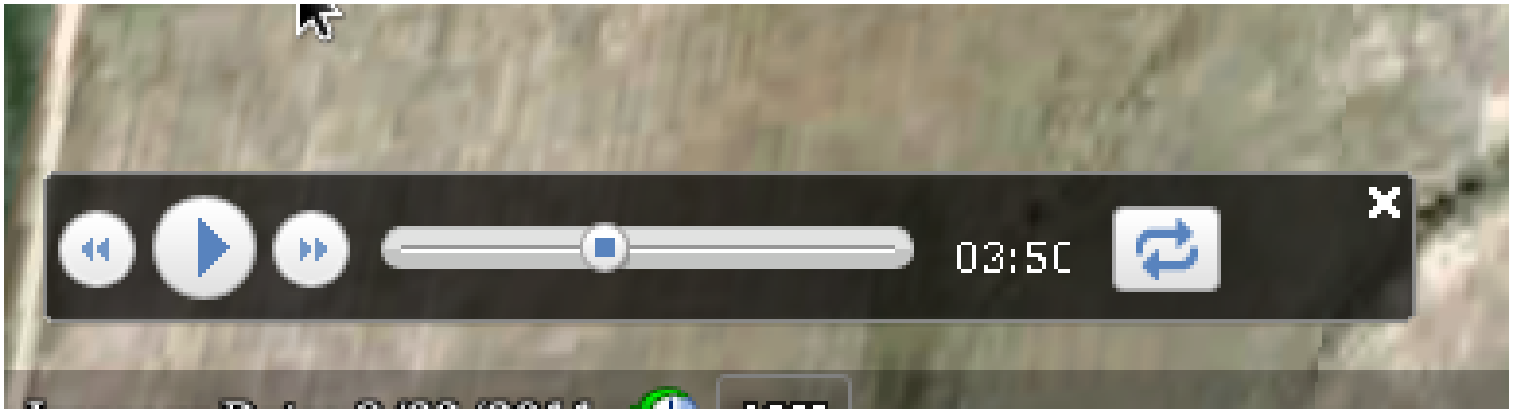


# An Example

Google Earth allows a few different kinds of added images in a scene, called “Overlays”. This is a Screen Overlay.

# An Example

Images, placemarks, overlays, etc. call be grouped and animated in a "Tour"




Tours navigate the viewer automatically, displaying and hiding objects at precise locations and for well-defined durations. They can include background audio.

# More Examples

This is useful for more than just pretty pictures.

- Ocean currents
- Tsunami shock waves
- Wind patterns
- Historical events

# Liquid Galaxy

Google Earth can talk to itself, to broadcast its view of the world. It can also receive these packets, and show related views. So if you put multiple instances together, you get a panoramic view.

# Liquid Galaxy

This panoramic view is called a Liquid Galaxy.

# Liquid Galaxy

End Point has done lots of work with Liquid Galaxies:

- They boot custom ISOs via PXE
- Tours and content can be controlled via touch screen
- We can monitor and repair galaxies remotely, easily
- Tour content can integrate Google Earth with other applications, such as mplayer

# Making tours

## Universal Truth

Writing XML in significant quantities by hand sucks.
Debugging and modifying it later redefines “suckage”.

# Making tours

Enter “Kamelopard”:

- Writes KML for you
- Ruby-based, for buzzword appeal
- Open source, so you can fix what’s broken
- Awkward name. It worked for PostgreSQL...

# Making tours

```
require 'rubygems'
require 'kamelopard'
require 'yaml'

f = folder 'Tour Resources'
data = YAML::load_file 'some_data.yml'
data.each do |d|
  p = point d[:longitude], d[:latitude]
  pl = placemark p, :desc => d[:desc]
  fly_to p, :duration => 4
  f << pl
end

write_kml_to 'doc.kml'
```

# Kamelopard

Kamelopard makes it easy (and succinct!) to generate KML for large data sets.

- Tour of End Point employees, taken straight from employee database
- FamilySearch mashup; show ancestral migrations
- "Smart" power meters' trouble messages vs. lightning strikes
- Fisheries' catch records plotted historically, also straight from the database

# Large datasets

KML can handle large datasets gracefully

- Regions: Data are loaded only when zoomed in close enough
- GroundOverlays: Data can be encoded into images that are mapped over the earth
- Combining the two, increasingly detailed images or sets of placemarks can appear as the user zooms in closer
- DataAppeal creates maps with various models in them, scaling and coloring them based on users' data

# NB! RE: LG

There are some important considerations with tours specifically for Liquid Galaxies

- Animations aren't broadcast, so only the master node will animate. Sometimes KML Regions can help mitigate this.
- Earth's ViewSync is currently broken. No historical data can be shown
- Launching tours is . . . convoluted. Build an HTML index for your tours.

# So...

- Google Earth is kinda neat
  - (though how might pictures of my backyard help the zombies advance their cause?)
- Liquid Galaxies are neat, too
  - They can make some pretty pictures
  - They can also show serious data

Questions?