Migrating from Google Earth to Cesium
How to begin your closed-to-open source success story

Hannah Pinkos
Analytical Graphics, Inc.
hpinkos@agi.com
Overview

- Google Earth API deprecation
- Introduction to Cesium
- Cesium and Google Earth Differences
- Data visualization and KML
- Resources
Google didn’t provide any clear alternatives for 3D on the web.
Migrating to Cesium

Monthly Forum Posts

GE deprecation
http://cesiumjs.org/Cesium/Apps/HelloWorld.html
Cesium

JavaScript library for creating 3D globes and 2D maps

Open source software with Apache 2.0 License
  – Free for commercial use

Actively developed, monthly releases

Cross-platform, cross-browser, cross-device with no plugin
  – Runs on any system with WebGL support
  – Chrome, Firefox, Safari and Internet Explorer
  – Windows, Mac, Linux and Mobile
Success Story - CloudAhoy

Flight debriefing for pilots

http://cesiumjs.org/demos/cloudahoy.html

- Flight recreated from GPS data
- Cockpit video synced with Cesium display
- Benefited most from being able to run application on iPad
+ AGI hosts world terrain for people to use,
  + Quantized-mesh, an open format for efficient 3D terrain streaming and rendering
  + Height maps, another popular standard, so people can hook up their own terrain provider
Cesium Features

Three scene modes
- 3D globe
- 2D map
- Columbus view
  (2.5D)

Time Dynamic Visualization
Success Story – Virtual Cities Project

Urban historical information

– Recreate historic landscape
– Learn about city history
– Cesium timeline used to step through history of a site
Cesium runs natively on the web
  – No installation
  – No plug-in

Build app using HTML5
  – HTML for page structure
  – CSS for page styling
  – JavaScript for page control

+ Demo Hello World included in release to show how to use HTML CSS and JavaScript together to add Cesium to the page
Security

Browsers have discontinued NPAPI plugin support due to security vulnerabilities
  – NPAPI plugins run C/C++ programs with the same permissions as the browser
  – Unrestricted access to the local machine
  – Access resources
  – Ability to create new processes

Cesium runs in browser sandbox, so browser security extends to Cesium
+ May have to change some of the methods we are using to load in data.
+ Common practice in KML to load models with a KML file that loads other KML files that load models. We can’t load city making 1 request per building
+ Loading really large files can also slow down the application, since the entire thing has to be retrieved from a server.
Cross-Origin Resource Sharing (CORS)
– Web server access control to restrict servers from a different origin from reading data
– CORS is opt in, server has to have it enabled
– Info and tutorials: http://enable-cors.org/index.html

Offline
– Requires local imagery (and terrain)
– Change KML to point to local files

+ This could be a problem if imagery is on one domain and app is on another
+ Might run into this if using old KML files pointing to unmaintained servers that haven’t been updated to have CORS enabled.
+ Modern services all have CORS enabled
+ No problem if your data and your application are on the same domain.
For simple applications, you may be able to complete your migration by loading your KML files into Cesium.
Example 1: a bunch of placemarks with icons styled depending on the type of facility the placemark is representing

Example 2: KMZ file with embedded resources showing GDP per capita

Example 3: using google multi-track extension that uses GPS data recorded during a bike ride
In their migration process, they were able to reuse nearly all of their KML files, which made the migration go pretty smoothly.

They did, however, change how their wind vectors were being rendered in order to improve the speed of the application.

They also were able to implement a selection tool to show their colormap popups, which replaced their use of screen overlays.
Data Visualization - KML

KML is not designed for the web
- KML and KMZ files can grow very large and can be slow to load
- Datasets with complex NetworkLinks can make too many requests
- KML uses COLLADA models, convert to glTF to use in Cesium
Users occasionally find that KML doesn’t work exactly the same way in Cesium as it did in Google Earth.

+ Difficult to know exactly what behavior should be in certain cases because KML spec is not well defined, and Google Earth adds KML functionality that is not officially in the spec.

Good news: Cesium supports many features beyond KML:

- API for building app programmatically
- GeoJSON and TopoJSON
- New web-friendly open formats: CZML, 3D Tiles, glTF
If you are experiencing slowness with your KML file, one of the first things to try would be to convert it to a CZML file.

+ KML is inefficient, GE could get away with more because it could use more system resources
+ CZML is a JSON schema, which makes it easier to process than KML, which is an XML schema.
+ The schema was also designed specifically for running in Cesium on the web with performance in mind
Team at Tokyo Metropolitan University migrated their application called the Hiroshima Archive

This application details the events of the bombing of Hiroshima and the aftermath through photos and testimonials from survivors.

Saw performance improvement moving to CZML from KML
+ 3D tiles is an open format for visualizing massive datasets.
+ Uses an LOD structure to stream data, so only visible or ‘important’ pieces are loaded depending on your current view.
+ 3D tiles has support for a few different datatypes and will allow for things like rendering all the buildings and trees in a city, displaying point cloud data, and drawing large vector data sets.
+ COLLADA is flexible and verbose, which is great for modeling tools and desktop because there are a lot of C++ libraries that work well with COLLADA. But not so great for graphics engines because the files are large and can requires some processing before rendering.
+ We worked with Khronos to develop glTF,
+ 3D model format that is optimized for rendering 3D models on the web.
+ Cesium hosts an online COLLADA to glTF converter to make it easy for you to convert your existing COLLADA models to glTF.
Migration Resources - Web

Web Programming

- HTML to create application layout, add screen overlays, buttons and other form controls
- CSS to style HTML elements: color, height, width, etc
- JavaScript to initialize Cesium and control the page behavior

- developer.mozilla.org, w3schools.com
Migration Resources - Cesium

Cesium Programming
- cesiumjs.org - Tutorials, code examples and reference documentation
  - Getting started tutorial
    http://cesiumjs.org/tutorials/cesium-up-and-running/
  - Sandcastle app (examples and live coding)
    http://cesiumjs.org/Cesium/Apps/Sandcastle/index.html
  - API Reference Documentation
    http://cesiumjs.org/refdoc.html
Migration Resources - Cesium

Google Earth to Cesium

– [http://cesiumjs.org/for-google-earth-developers.html](http://cesiumjs.org/for-google-earth-developers.html)

– One-to-one code examples

– Two part tutorial series

– Guest blog posts with first hand experiences and tips
Migration Resources - Cesium

Cesium Forum

- [http://cesiumjs.org/forum.html](http://cesiumjs.org/forum.html)
- 900+ members
- 500+ monthly posts
- Supported by the Cesium team and by members of the community
Coming Soon to Cesium

- Full 3D Tiles support
- Vector data on terrain
- Shadows

- New: CesiumJS.com
  - Host, process and analyze geospatial data
  - Build, style and publish Cesium 3D map
Showcase your app

We want to hear your success story!

http://cesiumjs.org/demos.html
Thank you!

Questions?