Ditching Your Desktop GIS:
Using new open source JavaScript tools to create compelling web maps and perform essential geospatial analysis

Abstract:

This paper examines Web GIS from an applied research perspective to provide best-practice recommendations for GIS practitioners who face a bewildering array of technology choices for making online maps and performing web-based GIS analysis. While acknowledging the perceived benefits of an all-in-one solution like ArcGIS or a comprehensive API like Google Maps, the paper uses a prototype-based methodology based on a leading "GIS Essentials" text (Clemmer, 2010) to support the thesis that it is feasible to ditch your desktop GIS and use new open source JavaScript libraries and tools such as D3, Leaflet and Turf to create compelling web maps and perform essential geospatial analysis.

Keywords:

“Web GIS”, "JavaScript mapping" , neogeography, "HTML5 geolocation API",

"Google Maps API", "D3 mapping", Leaflet, Mapbox, GeoJSON, TopoJSON, "Turf JS"

Web Presentation:

>> http://www.milemarker3.org/geog/web-gis-690/

"The web changes everything, and GIS is no exception." (Fu, 2010) So begins the Web GIS Principles and Applications text that sets a theoretical framework for understanding this rapidly-changing field. This paper will examine Web GIS from a more practical perspective, and provide best-practice recommendations for GIS practitioners facing a bewildering array of technology choices for making online maps and performing web-based GIS analysis.

A Web GIS can be defined as the combination of using a Web browser to access and interact with geographic information systems (GIS) data and maps. Traditional computer desktop GIS systems are often pitted against Web GIS as competing platforms: "While the capabilities of Desktop GIS and their applications are undisputed, there has been a steady growth in the number of Web GIS applications and apps." (Kumar, 2014)

Broadly speaking, "Two developments have had a great influence on the spread of interactive maps on the web: The growth of open-source alternatives to proprietary web mapping software, and the publication of web mapping application programming interfaces (APIs) by companies like Google and Esri that enable third-party developers (particularly those without a GIS background) to embed maps in their websites." (Detwiler, 2016)

For traditional GIS industry players like local governments and public utilities, "GIS data and maps need...to be peeled from the organization's traditional data silos and made available on the web." (Sack, 2012) For new GIS projects, GIS practitioners should use the best technology for the job. Recently, leading geospatial professionals such as James Fee are asking, "Is GIS [too] Heavy?", when there are Web GIS alternatives out there: "GIS has become difficult to use. - they have no idea what they’re doing...By making it easier to use the software, they have dumbed it down and made it a beast." (Fee, 2015)

Isn't there a better way to perform Web GIS, a simpler and web-centric method, a more cost-effective way? This paper proposes the new crop of tools that use JavaScript, the scripting language embedded in all web browsers, as the method best suited for Web GIS: "Until recently, developing geospatial apps...required a comprehensive GIS service such as ArcGIS, Nokia Here, or Google

Maps. While these APIs are powerful, they are also expensive, onerous to learn, and lock the map developer to a single solution. Fortunately, there are now a wealth of useful, open source JavaScript tools for handling advanced cartography and geospatial analysis." (Pepple, 2015)

In fact, these tools have matured to the point where this paper will show that it is possible to ditch your desktop GIS and use open source Javascript libraries and tools to create compelling web maps and perform essential geospatial analysis.

Reviewing existing literature on the burgeoning field of Web GIS faces a steep challenge: "It is advancing too rapidly for books to keep up with, it is too new for many principles to gel, it is too diffuse to condense, and it is sometimes too technical to easily explain." (Fu, 2010) These facts often cause authors, especially academic ones, to focus on conceptual underpinnings at the expense of applied research. Or worse, some authors in the geographic academic community fall victim to distraction in the face of the rapidly changing landscape of Web GIS. These studies examine the field not on its own merits, but as an encroachment of their turf in the so-called neogeography vs paleogeography debate that raged from roughly 2006 to 2012.

For these authors, "the proliferation of Web-based geographic information technologies and the parallel opening up of the mapping enterprise to non-expert cartographers" (Leszczyski, 2014) represents a threat to the status quo. Instead of evaluating the new tools, they engage in a sociological navel-gazing analysis of the proclaimed "newness" of location-based technologies and smear the neogeographers, whose practices they say "usually do not conform to the protocols of professional practice." (Leszczyski, 2014)

What these criticisms fail to see is the potential of web technologies in the hands of trained GIS practitioners to radically transform the process of web-based cartography and geospatial analysis. Perhaps charting the rise of Web GIS through a historical lens can yield some more practical insights into this fast-moving industry that is disrupting traditional GIS at every turn.

The short history of Web GIS can be characterized as a prolonged period of intense innovation punctuated by a few watershed events. "Web GIS has grown into a rapidly developing discipline since its inception in 1993." (Fu, 2010)

The spark came in 1993 when the Xerox Palo Alto Research Center (PARC) released their Map Viewer, a piece of software that pre-dated the first web browser. MapQuest was the first consumer web service available when it came online in 1996, and in 1997 the U.S. Census Bureau released a web interface for its rich Topologically Integrated Geographic Encoding and Referencing (TIGER) dataset, using data from the 1990 census.

“This early generation of Web GIS used Dynamic HTML, Java and ActiveX to produce the first interactive web maps, and GIS vendors developed server-based software (ESRI's ArcIMS, Intergraph's GeoMedia Web Map) for their clients (mainly public sector agencies) to put their geographic data online.” Detwiler, 2016) It was good to get maps on the web, but there were major flaws: "Complicated user interfaces that were intimidating to the general public, and very slow performance, as the software had to redraw the map every time the layers or the scale changed." (Detwiler, 2016)

Then, in 2005 came the watershed release of Google Maps (and Google Earth), which completely changed the web mapping game. Through a new technology implementation known as AJAX (Asynchronous JavaScript and XML), online maps could be refreshed on the fly, with improved performance that was no longer about click and wait. Google Maps used graphic tiles that were sliced up to represent the current map view in what came to be known as a slippy map interface. When the map user zoomed or panned somewhere else, an AJAX request went out to grab a new set of tiles. In this way, Google Maps, "Created the modern idea of a web application where data was fetched in the background rather than having to be refreshed to get new data." (Detwiler, 2016)

This innovation sent MapQuest, Yahoo Maps, Esri, and all the other online mapping players of the time scrambling to keep pace, adding AJAX functionality of their own, and rushing to keep up with features like turn-by-turn directions, Street View, and more.

At this early stage, neither market-leader Google Maps nor established GIS vendor Esri released what is known in software development as an application programming interface (API) for web mapmakers to make use of their platforms. Instead, they shielded the public and developers from the details of directions routing and other GIS-like solutions they provided and were effectively, "Black boxes of geographical wonder." (Davis, 2007)

Eventually Google Maps and Esri both released public APIs for GIS developers to use the JavaScript language (and early on, other technology like Flash and Silverlight which would soon fall by the wayside) to do things like place point markers on maps, import data and use events to make web maps interactive. This is where the use of JavaScript for online mapping begins to take root, as it is a scripting language already embedded inside every web browser. These APIs ushered in a period of so-called "mapping mash-ups" where data attributes like real estate prices or crime rates are added to a tiled basemap to create basic interactive web maps.

And yet, the big consumer services and leading GIS vendors often failed to make effective and usable web-mapping APIs: "Online mapping sites built with proprietary software sold by GIS vendors may have also suffered from the fact that their interfaces typically mimicked desktop GIS software interfaces, intimidating non-GIS users." (Detwiler, 2016)

Perhaps these shortcomings of the commercial APIs are one of the reasons that the open source GIS movement began to hit its stride during this period. "Using any commercial API leaves you at the mercy of whatever changes the provider makes to the API or terms of service...if you are looking to reuse your code for non-Esri based websites then again an open API serves you best." (Chimp, 2011)

Led by the Open Source Geospatial Foundation (OSGEO), these open APIs followed the conventions of all open source software, making code freely available under a liberal license, and encouraging a collective effort to contribute enhancements and modifications to the code. Their stack of geospatial software includes the OpenLayers JavaScript library for online mapping (launched in 2007), the PostGIS geospatial database, and Geoserver (or MapServer) for serving up mapping applications in the form of web services. A web service can be defined as a focused task that a specialized computer (the server) knows how to do and allows other computers to invoke. The OSGEO servers specialize in Web Map Services (WMS): "A Web Map Service is an Open Geospatial

Consortium specification that defines a common interface for disseminating digital maps, rendered from spatial data, across the Internet." (Sack, 2012)

This OSGEO suite of open source web mapping tools is still used today. They do an adequate job, but they can be overly complex, usually take a great deal of time and effort to setup and integrate, and they often fall short in the creation of aesthetically pleasing online maps. They do free up GIS practitioners from the expense and restrictions of the big commercial services, but there's a new breed of open source JavaScript tools and libraries that are easier to use, produce great-looking online maps and provide even more Web GIS power. It is these tools that this paper will evaluate for their ability to fully replace desktop GIS systems.

The current generation of Web GIS started in 2012, when Google Maps completed something called Operation Ground Truth (I'm not kidding) and began charging for the use of their map tiles and data. "They [Google] said to their top-tier customers, oh by the way, we're going to start charging you some serious money for all these maps that you've been embedding in your websites for the last 5-6 years" (Amul, 2013)

This caused web cartographers to do a new round of due diligence to see what else was out there, and led to a new-found interest in a product called Open Street Map. Open Street Map (OSM) is a free worldwide online map service that started in 2004 due to lack of available mapping data in the U.K.. It is designed to be an online reference for all, like a Wikipedia for maps, that uses volunteered geographic information (VGI) to make the most accurate maps possible. Originally, OSM had slow uptake in the U.S. whose history includes government-mandated mapping from the U.S. Geological Survey (USGS) and the prominence of previously free mapping services such as Google Maps, but after 2012 the stage was set for some explosive growth in free and open source web mapping.

In the meantime, ever since the pioneering use of the AJAX technology with the Google Maps launch in 2005, the web has become a full web application platform with a lightweight programming model powered by new-found uses for JavaScript. In a sense, these new JavaScript libraries for making web maps are riding a larger tide of the pervasive nature of JavaScript in the web platform as a whole. The dominance of the web platform for modern software means the rise of the API; the rise of the API means the dominance of JavaScript as it is built into every web browser, both desktop and mobile. This has even spawned a maxim known as Atwood's Law which states, "Any application that can be written in JavaScript, will eventually be written in JavaScript."

And this ubiquity of JavaScript has started to makes waves in the geospatial realm as a whole. In a recent University of Wisconsin competitive analysis of emerging web map mapping technologies, the authors concluded, "In testing technologies for next generation web apps, we're quickly moving primarily toward JavaScript-based frameworks." (Donohue, 2012) Another commentator notes, "JavaScript has formed a strong relationship with the geospatial world in a manner that is reminiscent of the relationship between Python and desktop GIS." (Kumar, 2014) All of this is strong evidence for promoting JavaScript as the best scripting language for Web GIS: "I'm starting to think JavaScript, not Python, is the future of GIS scripting." (Fee, 2014)

The most recent crop of JS libraries also ushers in a new level of Web GIS capabilities: "Developers in the open source world who have been hard at work over the past several years creating alternative JavaScript libraries for web mapping...In most cases, these libraries can also pull in the big-name basemaps." The cross-compatibility of all these tools allows for a wide and growing variety of possibilities when it comes to making cool web maps." (Sack, 2012)

This central role of JavaScript in the most recent round of Web GIS innovations is given voice by a provocative conference talk titled: "GIS Is Not Dead, It's Coming For You And It's Been Drinking JavaScript." The speaker is Christopher Helm and the venue is the Free and Open Source Software for Geo Conference in 2013. According to Helm, "The world changes fast...yet, our [GIS] industry is stuck, and has generally shown itself to be slow to react to new ideas and paradigms that grow rapidly in other spaces." His talk discusses several ways that JavaScript and the web have, "Re-shaped GIS and are changing how we visualize, analyze and share geospatial data with each other and the world." This talk is really a call to action to drive the use of JavaScript for web GIS forward, positing new tools like Leaflet, Open Layers and D3 as nothing short of "Freedom for web cartography."

And now on to our own call to action. Let's see how much these new JavaScript tools for Web GIS can do. This new breed of JavaScript libraries represent a big change from using a traditional desktop GIS, but proves every bit as powerful. While it represents an a la carte approach, these libraries can also be thought of as modules that act in concert to solve Web GIS problems.

Here we will briefly review the components of a front-end focused suite of tools to operate on vector data to do Web GIS. These libraries have all become available in the past 4-5 years, and have the following roles: GeoJSON (2008) is the new universal dialect for geographic data transport, Data-Driven Documents (D3 - 2011) is used for Geographic Data visualizations and projections, Leaflet (2011) is a kinder, simpler web mapping API, Mapbox (2010) is a new open web mapping platform, and Turf (2014) represents true GIS analysis for web maps. All these technologies taken together provide a toolset for performing thematic data-driven web mapping on the front-end.

But first, know that in addition to those libraries, the Web platform has some built-in mechanisms that aid the quest for web mapping, namely the HTML5 Geolocation API. Built into the web browser software itself, the HTML5 Geolocation API allows for programmatic access to geographic location information associated with a device, and by itself makes web pages location aware. This is how all location-aware apps on your mobile devices know your location (if you allow it when the browser or device asks if you want to share your location). The basic JavaScript methods for acquiring and using location data in web pages include a getCurrentPosition method that returns the latitude and longitude of the web user and a watchPosition method to track a user's location for mobile apps. This built-in API already gives us a leg up in building out Web GIS functionality.

Another boost comes from a handy new geographic data format known as GeoJSON. GeoJSON is a format for encoding a variety of geographic data structures. "In GeoJSON, a vector feature and its attributes are represented as a JavaScript object, allowing for easy parsing of the geometry and fields." (Newton, 2014) Under the hood, GeoJSON is a flavour of JSON (JavaScript Object Notation) which is made up of a simple collections of name/value pairs, known in JavaScript parlance as an object.

For Web GIS, what's important to know is that a GeoJSON file contains all the geometric information required to draw geographic data. It also contains any data attributes, making it a vehicle as flexible as the proprietary Esri shapefile, which remains the most common format for geographic data.

But GeoJSON does shapefiles one better; It is a universal dialect that all other JS web mapping tools know how to speak, and because it is already in JavaScript form, it is easy to parse and use the data to produce compelling online maps.

A further innovation creates a TopoJSON file, an extension to GeoJSON that encodes topology and saves space over GeoJSON by encoding shared line segments called arcs. TopoJSON files are typically 80% smaller than GeoJSON files with no loss in geographic precision. TopoJSON utilities can merge multiple .shp files into a small, single TopoJSON file, which is then converted back to GeoJSON and then is ready for use in Web GIS applications.

Other open source tools take the pain away from converting data from and to different formats, a pain point for many GIS projects, web or otherwise. For example, GIS practitioners can use the ogr2ogr data conversion tool to convert data to and from many common formats, such as converting shapefiles, Google Earth KML, or .GPX file GPS points to GeoJSON for use with JavaScript web mapping libraries.

To make a map with this GeoJSON data, a good place to start is with D3 (which stands for Data-Driven Documents) a new JavaScript library based wholly on existing web standards. D3's geo module can translate the GeoJSON data that represents the complex geometry of say, U.K. country polygons and make it available to be rendered on a web page as a map.

D3 gives the geographer a canvas to draw vector map geography onto, using SVG shapes, mostly in the <path> tag; You then can style these vector points, lines and polygons using web standard CSS, JavaScript and HTML, with compelling results: "Five years ago, all the best interactive maps were mashups - informative but clunky overlays made with Google or OpenStreetMap, with limited potential for customization and interactivity. Flash forward and today's best maps are being made with

D3, a JavaScript library that uses the full potential of the web to animate, transform, and decorate." (Stockton, 2013)

In the larger web industry, D3 is used to create charts and graphs and for data dashboards. Now you can use the same data and create both chart and map views of your data, and realize mapping can be thought of as just another kind of data visualization.

But when used for mapping, D3 can also create choropleth maps with baked-in color scales from color brewer (http://colorbrewer2.org/), maps with data overlays, and web interactivity. Plus, D3 finally adds the idea of providing multiple geographic projections for mapping your data, breaking the tyranny of Web Mercator dictated for so long by first mover Google Maps and other web mapping APIs.

One thing not included with D3 is the map base layer of graphical tiles and that familiar slippy map interface so characteristic of web cartography. For this piece of web GIS, the new JavaScript library called Leaflet is a great open source fit.

Leaflet is a simply-designed and fully-featured web mapping API. It is more lightweight and easier to use than the comparable Google Maps or Open Layers API, is freely available for use in web pages, and speaks GeoJSON. Like other web map libraries, “The basic display model implemented by Leaflet consists of a basemap plus one or more translucent overlays, with vector objects displayed on top.” (Wikipedia, 2014)

And Leaflet gets high marks from members of the developer community: "Leaflet is simply the best option for working with the display of points, symbols, and all types of features on web and mobile devices" (Pepple, 2015) Also, Leaflet maps are tile-layer agnostic, so now your map library API can be different than your tile library API, opening up flexibility for mapmakers to use OSM, Google, Bing, or even Esri base layers.

Leaflet is also extensible, so you can use plugins that can help you develop extra features. For example, for GeoJSON point data with elevation (z) values, you could create an elevation profile to accompany a hiking or mountain biking map using the free leaflet.elevation plugin.

In addition to the individual libraries, new web mapping platforms are also starting to take shape.

Leading this charge is a company called Mapbox that adds services on top of Leaflet and Open Street Map that rivals the offerings of the traditional players: "Mapbox is battling Apple and Google for the future of location-based data with open source" (Patel, 2013)

Mapbox is "An attempt at an all-in-one web map creation, hosting, and publishing platform built on open-source technology." (Sack, 2012) It provides server space to host your tiled maps and development tools for adding custom data and custom cartographic design elements. Mapbox enhances the basic cartography of Open Street Map data, has multiple JavaScript APIs that complement Leaflet, and provides Geocoding and other services in a software as a subscription service model. Mapbox also publishes a large amount of code for free use by the open-source mapping community including the csv2geojson utility to convert csv files to geojson files.

And finally, lest you think that JavaScript is all fine and good for putting points on a map, but take the prevailing old school view that is a far cry from a full-blown GIS, I'd like to introduce you to Turf: Turf is a JavaScript library for spatial analysis. It is nothing short than GIS for web maps. It helps you analyze, aggregate, and transform data in order to visualize it in new ways and answer advanced questions about it. "One of the great things about Turf.js is that you can create a collection of features and then spatially analyze, modify (geoprocess), and simplify them, before using Leaflet to present the data." (Pepple, 2015)

Any ArcToolbox tool you can name is very likely represented in the well-documented Turf API, which covers the majority of what GIS practitioners use in their daily work. Join methods use turf.within() to select points of interest located within a geographic area; Measurement methods use turf.distance() to calculate a great circle distance on the Earth; Classification methods use turf.jenks() to classify data according to the jenks algorithm. Almost every function in Turf.js takes GeoJSON data as its input, and it performs all of its geospatial analysis on the client-side, so there's no need for a server to perform fully-functioning Web GIS.

In conclusion, ditching not only your desktop GIS, but also throwing off the yoke of the first movers and established players in the web mapping API space is not only possible, but for many developers, is starting to be a preferred best practice.

This a la carte JavaScript (JS) library approach to Web GIS requires a shift in mindset and experience in coding. It may not be for everyone, and some GIS practitioners may still find value in the GIS-in-a-box approach offered by the big, venerated GIS vendors. "An integrated bundle of compatible software can be useful for government agencies and public utilities." (Sack, 2012). And the new ways are also not without downside: "The piecemeal approach presents challenges as well. For starters, some pieces fit together better than others." (Sack, 2012)

That being said, the benefits of using free JavaScript tools to perform Web GIS are many, including interoperability, low cost and the ease of creating compelling web cartography. And the big mainstream players in the GIS and online mapping space are noticing: "In the proprietary software realm, Esri has dragged its feet on GeoJSON support, offering its own... geometry formats...However, Esri has informally shared an open source JavaScript library to convert between the two formats." (Quinn, 2014) Esri also has a plugin for Leaflet to use with ArcGIS services, as additional evidence that they are bending to the winds of the open source movement.

This true neogeography does however, assume a GIS practitioners ability to code up these solutions using web development languages and libraries. "The only barrier to all this potential is your own pain threshold for learning to write code"...Get over your fear of coding and start making better maps." (Stockton 2013) When you do so, you will be able to ditch the expensive dependency of your desktop GIS and ride the wave of Web GIS into a free and open source future.

Further Research: This paper on using JavaScript tools for Web GIS has focused on a front-end development technology stack, where JavaScript has a long history. Further research might tackle raster processing with open source tools (Leaflet/Mapbox or Google's Terra Bella / SkyBox), geospatial databases (MongoDB JavaScript datastore, PostGIS open source spatial database), Web GIS on the server side (Node.JS vs GeoServer, MapServer) or 3d Viewers using Web GL browser-based technology. Also, follow-up studies on this rapidly-moving field and its potential disruption to traditional GIS methods would be welcome additions to this preliminary research.

REFERENCES

Agafonkin, Vladimir. 2015. *How simplicity will save GIS.* FOSS4G 2015 Conference. https://vimeo.com/142115161 (last accessed 29 March 2016)

______. 2015. *Leaflet Quick Start Guide.* Leaflet JS. http://leafletjs.com/examples/quick-start.html (last accessed 29 March 2016)

______. 2015. *Interactive Choropleth Map.*Leaflet JS. http://leafletjs.com/examples/choropleth.html (last accessed 29 March 2016)

Bostock, Mike. 2015. D3 Geo Projections. https://github.com/mbostock/d3/wiki/Geo-Projections (last accessed 28 March 2016)

______. 2012. *Let's Make a Map*. https://bost.ocks.org/mike/map/ (last accessed 21 February 2016)

Bostock, Mike, V. Ogievetsky, and J. Heer. 2011. D3: Data Driven Documents. *IEEE Transactions on Visualization and Computer Graphics* 17 (12): 2301 - 2309.

Boulos, Maged, Jeffrey Warren, Jianya Gong and Peng Yue. 2010. Web GIS in practice VIII: HTML5 and the canvas element for interactive online mapping. *International Journal of Health Geographics* 9 (14).

Chimp, Mr. 2011. *How do various JavaScript mapping libraries compare?* GIS Stack Exchange. http://gis.stackexchange.com/questions/8032/how-do-various-javascript-mapping-libraries-compare (last accessed 24 Feb 2016)

Clemmer, Gina. 2010. *The GIS 20 Essential Skills*. Redlands, CA: ESRI Press.

ColorBrewer: Color Advice for Maps. http://colorbrewer2.org (last accessed 27 March 2016)

D3: Data Driven Documents. https://d3js.org (last accessed 21 February 2016)

Davis, Scott. 2007. *GIS for Web Developers: Adding 'Where' to Your Web Applications*. Raleigh, North Carolina: The Pragmatic Programmers, LLC.

Detwiler, Jim. 2014. *GIS Mashups for Geospatial Professionals* (online course Geog 863). The Pennsylvania State University. https://www.e-education.psu.edu/geog863/ (last accessed 29 March 2016)

Donohue, Richard, Timothy Wallace, Carl Sack, Robert Roth, Tanya Buckingham. 2012. *Keeping Pace with emerging web mapping technologies.* University of Wisconsin-Madison http://geography.wisc.edu/cartography/slides/emerging-web-mapping-donohue-et-al-nacis2012.pdf

Fee, James. 2015. *Is GIS Heavy?* Spatially Adjusted. http://spatiallyadjusted.com/spatial/2015/08/20/is-gis-heavy.html (last accessed 24 Feb 2016)

Flowers, Rebeckah. 2015. *Steve Coast on OpenStreetMap, HOT and paying it forward*. Directions Magazine. http://www.directionsmag.com/entry/steve-coast-on-openstreetmap-hot-and-paying-it-forward/446539 (last accessed 24 Feb 2016)

Fu, Pinde and J. Sun. 2010. *Web GIS: Principles and Applications*. Redlands, CA: ESRI Press.

Gannes, Liz. 2015. *Origin Story Google Maps: Ten Years of Google Maps, From Slashdot to Ground Truth.* Re/code: Tech News, Reviews & Analysis. http://recode.net/2015/02/08/ten-years-of-google-maps-from-slashdot-to-ground-truth. (last accessed 28 March 2016)

GeoJSON Format Specification. http://geojson.org/geojson-spec.html (last accessed 25 February 2016)

Geolocation API Specification https://www.w3.org/TR/geolocation-API/ (last accessed 25 February 2016)

Geospatial Data Abstraction Library (GDAL) ogr2ogr: converts simple features data between file formats. http://www.gdal.org/ogr2ogr.html (last accessed 27 March 2016)

Haklay, Muki, A. Singleton, and C. Parker 2008. Web Mapping 2.0: The Neogeography of the GeoWeb. *Geography Compass*, 2: 2011–2039

Helm, Christopher Helm. 2013. *GIS Is Not Dead, It's Coming For You And It's Been Drinking JavaScript* FOSS4G 2013 Conference. https://www.youtube.com/watch?v=F003DhZ3RRw (last accessed 29 March 2016)

Herlocker, Morgan. 2014. *Turf: GIS for web maps.* Mapbox Blog. https://www.mapbox.com/blog/turf-gis-for-web-maps/ (last accessed 29 March 2016)

Introduction to Turf.js.MapBox. https://www.mapbox.com/help/intro-to-turf/ (last accessed 29 March 2016)

JavaScript Basics for Geospatial Web Applications. 2012. Open Geo. http://presentations.opengeo.org/2012_javascript/javascript (last accessed 29 March 2016)

Khoury, Adam. 2014. *Dual Image Reveal Control Tutorial (JavaScript CSS HTML)* https://www.youtube.com/watch?v=eyEJvujXAwM (last accessed 29 March 2016)

Klammer, Ralf. 2013. *D3 Mapping Basics.* Digital Geography. http://www.digital-geography.com/d3-mapping-basics-draft-for-digital-geography-com/#.Vtcg3ZMrK-x (last accessed 28 March 2016)

Kumar, Muthukumar. 2014. *JavaScript for Geospatial applications: An Overview.* http://geoawesomeness.com/javascript-for-geospatial-applications-an-overview (last accessed 24 Feb 2016)

Lamsal, Aashis. 2011. *Choropleth mapping techniques for Web2.0.* GIS-Remote Sensing Tools, Tips, and Scripts. http://www.gisremotesensing.com/2011/05/choropleth-mapping-techniques-for-web20.html (last accessed 28 March 2016)

Leaflet: an open-source JavaScript library for mobile-friendly interactive maps. http://leafletjs.com (last accessed 25 February 2016)

Leszczynski, Agnieszka. 2014. On the Neo in Neogeography. *Annals of the Association of American*

Geographers. 104 (1): 60-79. Mapbox: Design and publish beautiful maps. http://www.mapbox.com (last accessed 27 March 2016)

Matthews, Sam. 2015. *Geocoding in the browser.* GIS Collective (GISC). http://giscollective.org/geocoding-in-the-browser (last accessed 28 March 2016)

Newton, Thomas, and O. Villarreal. 2014. *Learning D3.js Mapping: Build stunning maps and visualizations using D3.js*. Birmingham, UK: Packt Publishing Ltd.

OpenStreetMap: a map of the world. http://www.openstreetmap.org (last accessed 27 March 2016)

Palazzolo, Alan. 2013. *Simple Map D3: Easy "choropleth" style maps with D3.* http://code.minnpost.com/simple-map-d3 (last accessed 28 March 2016)

Pepple, Steve. 2015. *Using JavaScript to Create Geospatial and Advanced Maps.* Sitepoint.com. http://www.sitepoint.com/javascript-geospatial-advanced-maps/ (last accessed 21 Feb 2016)

Proj4js: JavaScript library to transform coordinates from one coordinate system to another. http://proj4js.org (last accessed 27 March 2016)

Quinn, Sterling. 2014. *Open Web Mapping* (online course Geog 585). The Pennsylvania State University. https://www.e-education.psu.edu/geog585/ (last accessed 29 March 2016)

Sack, Carl. 2012. *Web Mapping Services Introduction Education Series.* GIS Collective (GISC). http://giscollective.org/tutorials/web-mapping/wmsintro (last accessed 28 March 2016)

Sinton, Diana. 2016. *OSM: The simple map that became a global movement.* Directions Magazine. http://www.directionsmag.com/entry/osm-the-simple-map-that-became-a-global-movement/466280 (last accessed 29 March 2016)

Stockton, Nick. 2013. *6 Reasons to Get Over Your Fear of Coding and Start Making Better Maps.* Wired.com Map Lab. http://www.wired.com/wiredscience/2013/12/make-better-maps-with-d3/ (last accessed 21 Feb 2016)

TopoJSON Format Specification: An extension to GeoJSON that encodes topology. https://github.com/mbostock/topojson (last accessed 25 February 2016)

TURF: Advanced geospatial analysis for browsers and node. http://turfjs.org (last accessed 25 February 2016)