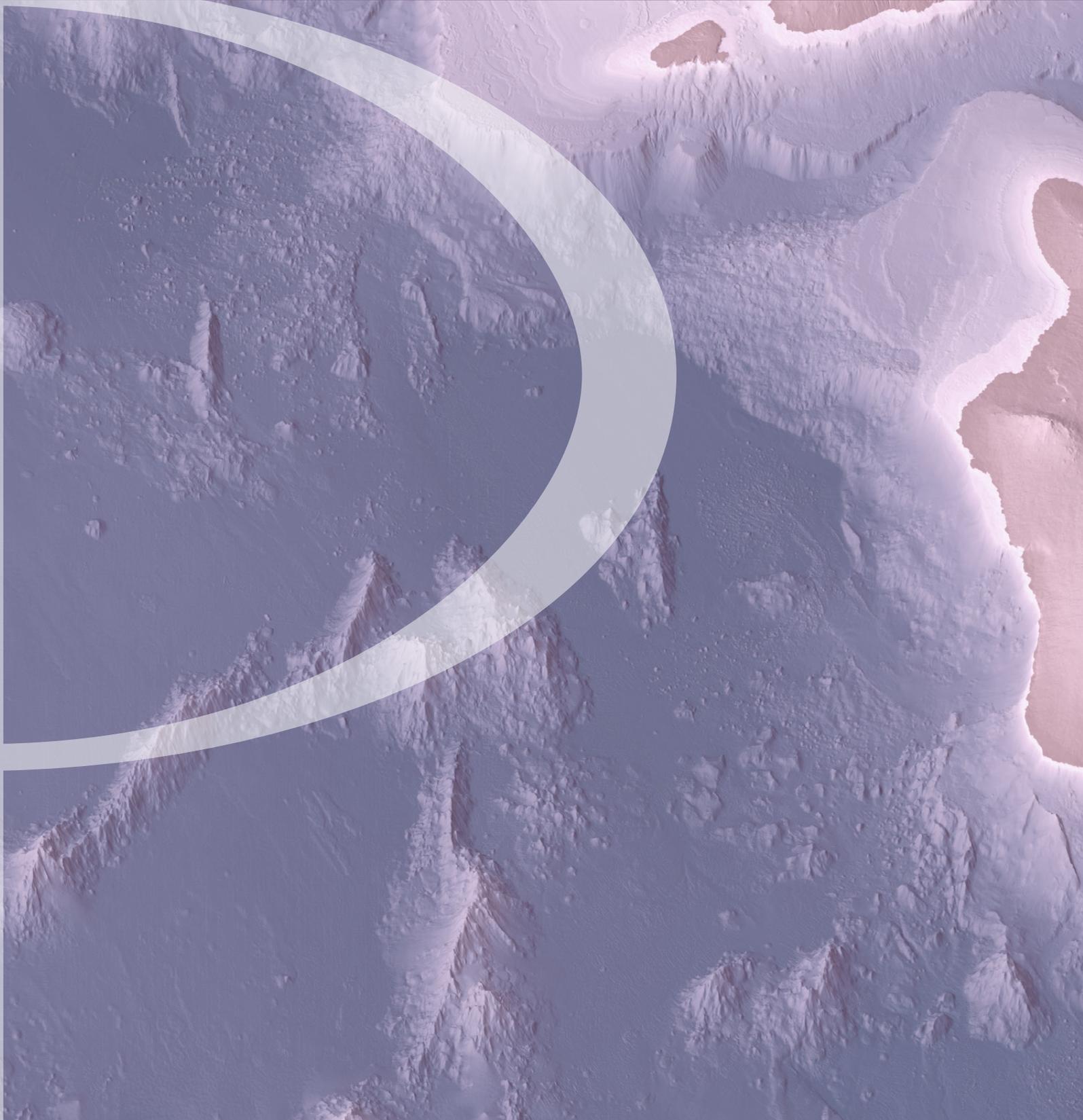


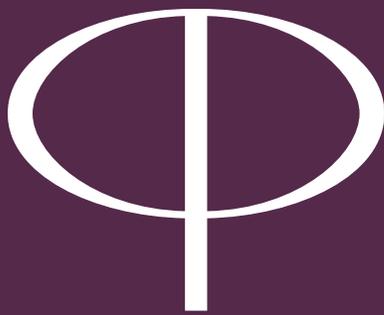


Cartographic Perspectives

The Journal of **nacis**

Number 76, 2013





IN THIS ISSUE

LETTER FROM THE EDITOR

Patrick Kennelly

3

PEER-REVIEWED ARTICLES

Mountains Unseen: Developing a Relief Map of the Hawaiian Seafloor
Tom Patterson

5

Identification of Partially Occluded Map Symbols
Jari Arvi Korpi, Mohammad Haybatollahi, Paula Ahonen-Rainio

19

CARTOGRAPHIC COLLECTIONS

Maps as the Art of Necessity: The Lilly Library War of 1812 Map Collection
Lori Dekydtspotter, Heiko Mühr

33

ON THE HORIZON

Time Series Proportional Symbol Maps with Leaflet and jQuery
Richard G. Donohue, Carl M. Sack, Robert E. Roth

43

PRACTICAL CARTOGRAPHER'S CORNER

Census Mapping Mashup
Paul Hunt

67

VISUAL FIELDS

Experimental Terrain Representation
Kelly Abplanalp

77

REVIEWS

London: A History in Maps *(reviewed by Patti Day)*

The World at Their Fingertips: Eighteenth-Century British Two-Sheet Double-Hemisphere World Maps *(reviewed by Ian Muehlenhaus)*

79

Napa Valley Historical Ecology Atlas: Exploring a Hidden Landscape of Transformation and Resilience *(reviewed by Lisa Sutton)*

Instructions to Authors

85

Cartographic Perspectives

The Journal of **nacis**

ISSN 1048-9053

www.nacis.org | [@nacis_cp](https://twitter.com/nacis_cp)

©2013 North American Cartographic Information Society

EDITOR

Patrick Kennelly
Department of Earth and
Environmental Science
LIU Post
720 Northern Blvd.
Brookville, NY 11548
patrick.kennelly@liu.edu

EDITORIAL BOARD

Sarah Battersby

University of South Carolina

Raechel Bianchetti

Michigan State University

Cynthia Brewer

The Pennsylvania State University

Mathew Dooley

University of Wisconsin–River Falls

Matthew Edney

University of Southern Maine
University of Wisconsin–Madison

Sara Fabrikant

University of Zürich

Amy Griffin

University of New South Wales–
ADFA

Fritz Kessler

Frostburg State University

Bernhard Jenny

Oregon State University

Mark Monmonier

Syracuse University

Ian Muehlenbaus

James Madison University

Margaret Pearce

University of Kansas

Michael Peterson

University of Nebraska at Omaha

Anthony Robinson

The Pennsylvania State University

Robert Roth

University of Wisconsin–Madison

ASSISTANT EDITOR

Daniel P. Huffman

somethingaboutmaps

daniel.p.huffman@gmail.com

SECTION EDITORS

CARTOGRAPHIC COLLECTIONS

Terri Robar

University of Miami Libraries
trobar@miami.edu

ON THE HORIZON

Andy Woodruff

Axis Maps
andy@axismaps.com

PRACTICAL CARTOGRAPHER'S CORNER

Alex Tait

International Mapping
Associates
alex@internationalmapping.com

VISUAL FIELDS

Laura McCormick

XNR Productions
laura@xnrproductions.com

REVIEWS

Lisa Sutton

lisarssutton@gmail.com

COPY EDITING: *Daniel P. Huffman*

ABOUT THE COVER: A color-adjusted detail from the unlabeled version of the Seafloor Map of Hawai'i, by Tom Patterson. See page 5 for article and more images, or visit www.shadedrelief.com/hawaii.

COPYRIGHT AND LICENSING: Unless otherwise noted, CP's contents are licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.



To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

The opinions expressed herein are those of the author(s), and not necessarily the opinions of NACIS.



LETTER FROM THE EDITOR

As I prepare to introduce readers to the content of issue 76, I recognize the changing role of the “Letter from the Editor” in an evolving journal. Past Editors would reflect on scintillating new content premiering on the gleaming pages of our paper journal. Now, I reflect on illustrious articles you may have seen some time ago on the bedazzling pages of our website, www.cartographicperspectives.org.

This transformation recognizes the importance of posting content after its acceptance by Section Editors or myself in as timely a manner as possible. No less should be expected with our commitment to blazing a trail through the landscape of open access publication. Authors love it. Readers presume it. Still, it was no easy task to take this spark of an idea and carefully enkindle it to the point that it shines.

For making this happen over the past three years, I owe my sincere thanks to Rob Roth, who stepped down from his duties as Assistant Editor after CP 75. Rob was always a strong proponent of making CP open access and promptly available. Moreover, he was willing and somehow able to commit his time and effort to this cause, despite competing pressures to complete a Ph.D. dissertation and begin a tenure-track faculty position. I feel CP owes much of its success in transforming itself into its current format to Rob’s efforts.

With this in mind, I managed to convince Rob to take on more, different work. Rob has agreed to serve on the Editorial Board of CP. Additionally, along with Anthony Robinson, Rob will serve as Guest Editor for a Special Issue of CP dealing with representation, cognition, and behavior. Look for articles on these hot topics in the full issue of CP 77, or even sooner on our website!

I’m also pleased that Daniel Huffman has agreed to take over Rob’s duties as Assistant Editor. Daniel’s efforts to date with CP have focused on layout and copyediting. With Daniel’s swelling credentials as Director of Operations at NACIS, I am glad he has agreed to take up the torch of keeping web content current. As Daniel is an Editor of the first two *Atlas of Design* volumes, I’m confident the web pages will be lustrous in design as well.

In other news, I’m happy to announce the winner of the CP Student Peer-Reviewed Paper competition for 2013. Nicholas Perdue was awarded the \$1,350 prize for his contribution “The Vertical Space Problem,” which appeared in CP 74 and offers an innovative approach to mapping multiple floor residential patterns. Nick feels the award motivated him to

transform his Master's thesis into an article. He also believes CP provided valuable insight and experience with the peer-review process. We hope the beacon of open access and timely publication continues to attract such quality submissions from students researching cartography.

Returning to CP 76, the peer-reviewed articles herein strive to bring information and representation to light. Tom Patterson takes inspiration from classic cartographers to represent the relief of the unseen seafloor of Hawai'i, in spite of bathymetric data laden with artifacts. Jari Korpi and his colleagues write about a different sort of cartographic challenge, testing the ability of users to identify partially occluded symbols on a map. These contributions focus on designing maps that are more appealing and useful to the greater cartographic community.

This issue's Cartographic Collections section, authored by Lori Dekydtspotter and Heiko Mühr, highlights the Lilly Library War of 1812 collection. They nicely integrate historical maps with a discussion of the emerging national geography in the early 19th century. The current Visual Fields showcases the work of Kelly Abplanalp, who combines hand-drawn linework with computer automated shading to produce evocative terrain maps of the Pacific Northwest.

On the Horizons features a contribution from Richard Donohue and his colleagues, who explain how Leaflet and jQuery can be used to create time series maps using proportional symbols. This tutorial should prove illuminating to anyone exploring mapmaking on this forefront of the technology. In the Practical Cartographer's Corner, Paul Hunt of the University of Nebraska at Omaha steps the reader through an innovative mashup that allows the user to explore the voluminous data available from the US Census Bureau. Finally, rounding off the issue are three reviews by Patti Day, Ian Muehlenhaus, and Reviews Section Editor Lisa Sutton, with each of the books reviewed offering a very different perspective on the history of mapmaking.

I hope you enjoy this issue of CP, regardless of when and how you are exposed to the various articles.

Patrick Kennelly
Editor

Mountains Unseen: Developing a Relief Map of the Hawaiian Seafloor

Tom Patterson
National Park Service
tom_patterson@nps.gov

The Seafloor Map of Hawai'i, a medium-scale relief map intended for lay audiences, posed production and design challenges typical of ocean-bottom mapping in general. The biggest problem was incomplete bathymetry data marred by artifacts. Fixing these bad data—filling voids and removing background noise—involved techniques similar to those used by cartographers for terrain mapping. Map design posed another challenge: how to depict a region on Earth that humans will never see. The Seafloor Map of Hawai'i uses plan oblique relief, which reveals the seafloor features with three-dimensional offset, a technique borrowed from National Geographic maps painted by Heinrich Berann and Tibor Tóth. Other challenges included selecting depth tints and relief colors based on the idea of cartographic realism and determining the names of seafloor features, many of which are unofficial and inconsistently identified.

INTRODUCTION

THE VOLCANIC HAWAIIAN ISLANDS are among Earth's most prominent mountains when measured from their bases on the ocean floor. The summit of Mauna Kea on the island of Hawai'i rises 10,000 meters, 1,100 meters taller than Mount Everest's height above sea level. Its sprawling neighbor, Mauna Loa, ranks as one of the most massive single mountains on Earth (Kaye and Trusdell 2002; Sager et al. 2013). Yet with 60 percent of their total height hidden beneath the Pacific, most people do not comprehend the size of Hawai'i's mountains. This paper discusses the *Seafloor Map of Hawai'i*, a new map that attempts to remedy this misperception. It depicts the Hawaiian Islands in their entirety from seafloor to summit with consistent detail throughout.

The *Seafloor Map of Hawai'i* overcame the challenges of representing seafloor topography derived from digital data on a regional, medium-scale map (1:897,000). General readers who do not use nautical charts or peruse scientific reports on oceanography are the target audience. The aim was to produce a “user-friendly” wall map of the Hawaiian seafloor that is the equivalent of a physical reference map for land. Accomplishing this posed questions: do data manipulation and relief presentation techniques developed for terrestrial mountains also apply to the ocean bottom

features? And, do the relief presentation techniques employed for small-scale seafloor maps, such as those found in *National Geographic* atlases, apply to larger-scale maps?

Compared to terrestrial mapping, the cartographic literature provides little information on how to present seafloor relief. For example, Eduard Imhof (1982, 205) enthuses about the potential of seafloor relief depiction:

Newer maps reveal and depict forms of astonishing variety. It would seem to be the natural next step to map submarine relief in three-dimensional shaded form in a similar manner to the land surface.

Yet, a few sentences later he dismisses the subject, ending the discussion:

Oblique hill shading, if used for underwater relief forms—and, hence, areas which are normally hidden both from light and our view—tends to produce unrealistic effects. In general it is probably more significant to provide good information on the depths of



© by the author(s). This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

the ocean floor than to portray the shapes there upon. (205)

More recent literature is similarly meager. Although examples of seafloor maps painted by skilled artists are available for reference, information on completely digital production methods is not. For example, former *National Geographic* artist Tibor Tóth offers samples of his beautiful seafloor art on his blog (Tóth 2009). Tóth also describes the “head-scratching situation” of rendering ocean bottom relief from digital data that were “full of serious imperfections.” His solution was to finish the work by hand:

It is conceivable that to someone with only computer based cartographic background this might have looked like a hopeless situation. To me this was where the years of conventional relief painting experience kicked

into gear. With the help of the amazing pressure sensitive WACOM tablet, and the various tools afforded by Adobe Photoshop (airbrush, smudge/dodge tools, and various filters), I produced a refined intermediate image.

The problem is that few cartographers have Tóth’s artistic ability or the time to devote to illustrating seafloor relief. Furthermore, compared to Tóth’s maps of entire oceans, manually producing medium-scale seafloor maps takes considerably more time because of the greater detail. Considering that water covers 71 percent of Earth’s surface, and that the body of bathymetry data is slowly growing, a discussion on digital production is overdue. Hawai’i, with its extreme undersea topography, plentiful medium-resolution data, and general interest to readers, offers a useful case study (Figure 1).



Figure 1: The Seafloor Map of Hawai’i measures 87.4 x 64 cm when printed and covers 433,000 square kilometers of area.

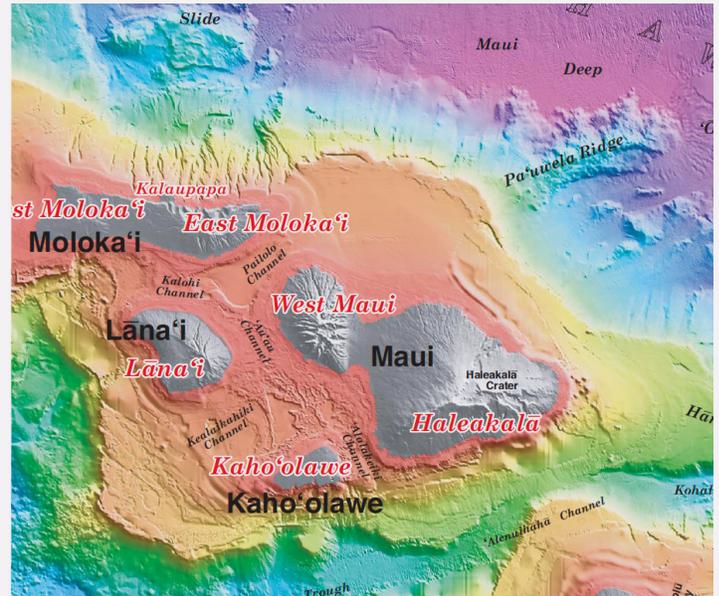
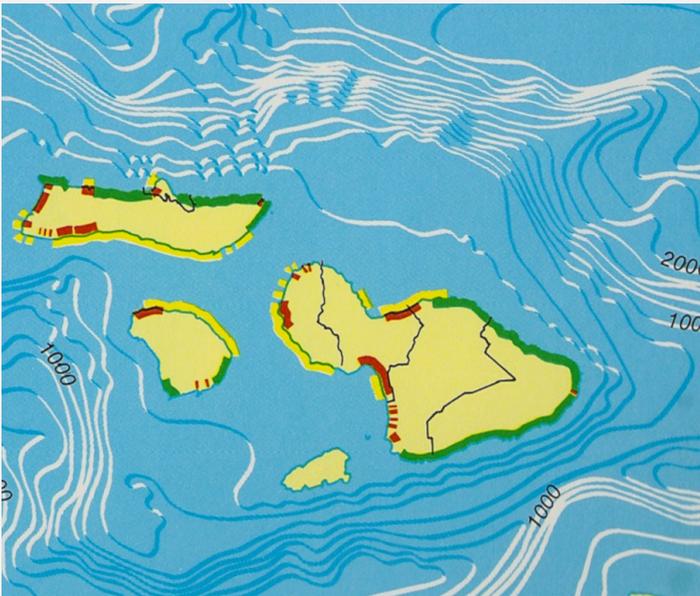


Figure 2: Excerpts of “Bathymetry and Shorelines” from the 1973 Atlas of Hawaii (left), and 2003 Hawaii’s Volcanoes Revealed (right). The pen and ink 1973 map depicts major features remarkably well compared to its 2003 digital counterpart.

The desire to depict the ocean bottom on maps of Hawai‘i is not new. It started with the first published maps of the islands—the coastal charts made by Cook, Kotzebue, Wilkes, and other early explorers—which used depth soundings (Fitzpatrick 1987). More extensive seafloor maps of Hawai‘i started appearing after the mid-20th century and became more common with the growing availability of bathymetry data from deep waters near the islands. Select examples are the “Bathymetry and Shorelines” map in the *Atlas of Hawaii*¹ (Moberly 1973) that employs the Tanaka method of illuminated isobaths (Tanaka 1950) at 1200-foot (366-meter) intervals (Figure 2, left). Cover art in the second edition of the *Atlas of Hawaii* (1983) features an oblique view from the east of the island chain emerging from the ocean depths. Manually plotted bathymetric profiles are the foundation for this airbrushed art by Everett Wingert. In 1985, Raven Maps published *Hawaii*, a state wall map with shaded relief, bathymetric tints (depth colors), and Raven’s signature hypsometric tints on land (Raven Maps & Images 1985). This was followed by *Hawai‘i* from *National Geographic* (2002), featuring ocean floor relief painted by hand in a style similar to the ocean plates found in their atlases and magazine supplement maps (see Figure 7 for atlas map examples).

The USGS map, *Hawaii’s Volcanoes Revealed* (Eakins et al. 2003) achieved a milestone by depicting the Hawaiian

seafloor from digitally rendered bathymetric data (Figure 2, right). The map was a little ahead of its time, however. Large areas of the seafloor, including key areas next to the islands, derive from low-resolution data, creating a discordant patchwork that detracts from its appearance. For example, the seafloor adjacent to the southeast coast of Maui is coarser than its surroundings (Figure 2, right). The map’s design is also an issue. As is often the case on ocean bottom maps made by scientists, a rainbow color scheme represents bathymetric zones; for example, shallow water is tinted warm red. The light source for the shaded relief originates from the northeast instead of the more conventional northwest, which increases the likelihood of readers perceiving the relief as inverted (Imhof 1982). The overall appearance of *Hawaii’s Volcanoes Revealed* is that of a research visualization, not a finished map.

In contrast, the *Seafloor Map of Hawai‘i* introduced here is a general reference map. Because of its online distribution, designing a map that could attract and hold the attention of a broad range of readers was a key consideration. Even the choice of *Seafloor Map of Hawai‘i* as the title strives for efficient web search results. Once retrieved, the map’s aim is to entice readers to explore, pausing occasionally to read text blurbs that explain Hawaiian seafloor features. The immediate message to readers is that most of Hawai‘i lies beneath the waves and only with a map can one visualize what is there.

1. Several cited publications spell Hawai‘i without the ‘okina diacritical mark, which is now standard.

Making the *Seafloor Map of Hawai'i* was a long-planned project that depended on the public release of high-resolution bathymetry data for the ocean bottom adjacent to the Hawaiian Islands, which finally occurred in May 2011.

The map took five weeks to produce as a part-time project in late 2011 and early 2012. As is typical of digital map production, data and design issues intertwine, although I will treat these issues separately so as to focus the narrative.

BATHYMETRY DATA OVERVIEW

FINDING AND MANIPULATING DATA to make the *Seafloor Map of Hawai'i* exemplifies the issues confronting global seafloor mapping as a whole: the ocean is huge, bathymetric surveying is painstakingly slow, and the available data are often poor or incomplete. Because water is a poor conductor of electromagnetic energy, techniques used to gather elevation data on land do not work for the ocean bottom. Even the surface of Mars, millions of kilometers from Earth, has more complete and detailed elevation data than the seafloor only a few kilometers below the ocean surface (Smith 2004).

The only complete global bathymetry dataset currently available derives from satellite altimetry measurements. Based on radar emitted from a satellite, this method detects slight variations in the sea surface height, compensating for waves and tides, to estimate the seafloor topography far below. (Sandwell and Smith 1997). Satellite altimetry data give a coarse snapshot of the seafloor at 2-arc second (~5-kilometer) resolution. It is the basis for the seafloor in the ETOPO2 world elevation dataset (National Geophysical Data Center 2006), and it is used as filler for areas with missing data in higher-resolution datasets, including SRTM30 Plus (Scripps Institute 2013), and GEBCO (2013).

HAWAI'I DATA

THE *SEAFLOOR MAP OF HAWAI'I* comprises bathymetry data from several sources. The cartographic challenge was patching together these disparate data to create a seafloor map that looked seamless. Map production involved rendering multiple pieces of terrain art from these data sources in Natural Scene Designer Pro and compositing the results in Adobe Photoshop.

The primary dataset was the Main Hawaiian Islands Multibeam Bathymetry Synthesis, version 19 (Hawaii Mapping Research Group 2011), a product of the University of Hawai'i at Mānoa, School of Ocean Earth

Since the 1970s, multibeam echo-sounders, a type of sonar towed by survey ships, have collected much higher resolution data. The time that a sound wave takes to reach the bottom and return determines the depth, taking into account the ship's constantly changing position on the surface. Continuous multibeam surveys record depths in swaths that become wider and detect more ocean bottom as the water deepens. Multibeam surveys are costly and time-consuming undertakings, however. After decades of effort, surveys are complete for less than 10 percent of the world seafloor, and they often appear randomly located with gaps in the coverage. It is estimated that the remaining 90 percent will take 120 ship-years of survey time to systematically complete using this same technology (Becker et. al. 2009). Broad continental shelves will take the most time to survey because of the relatively narrower multibeam swaths in shallow waters. Despite these difficulties, multibeam bathymetry is now available from the NOAA National Geophysical Data Center for most of the seafloor adjacent to US coasts, including Hawai'i (National Geophysical Data Center 2013a).

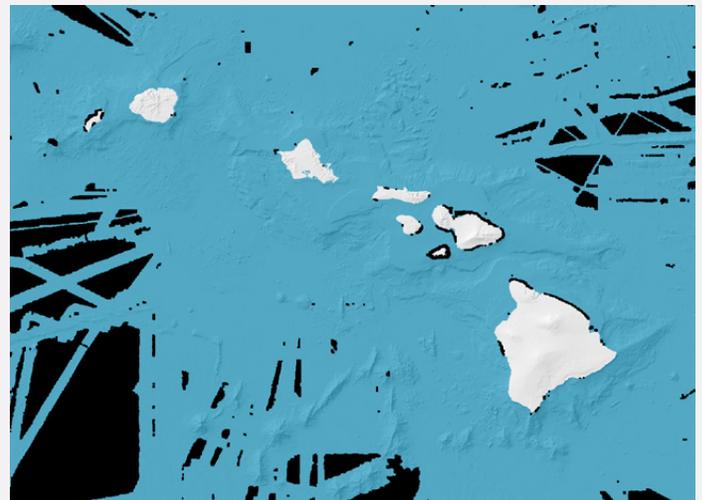


Figure 3: SOEST Main Hawaiian Islands Multibeam Bathymetry Synthesis, version 19, combines multibeam bathymetry (blue) with USGS DEMs on land (light gray). Black indicates gaps in the bathymetry coverage.

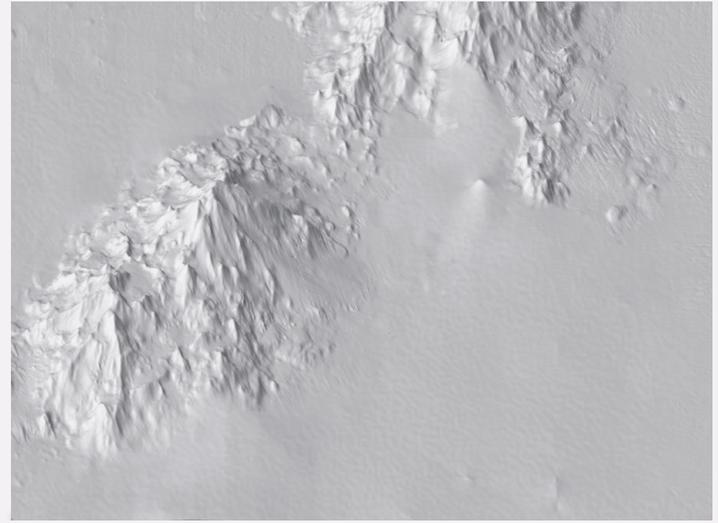
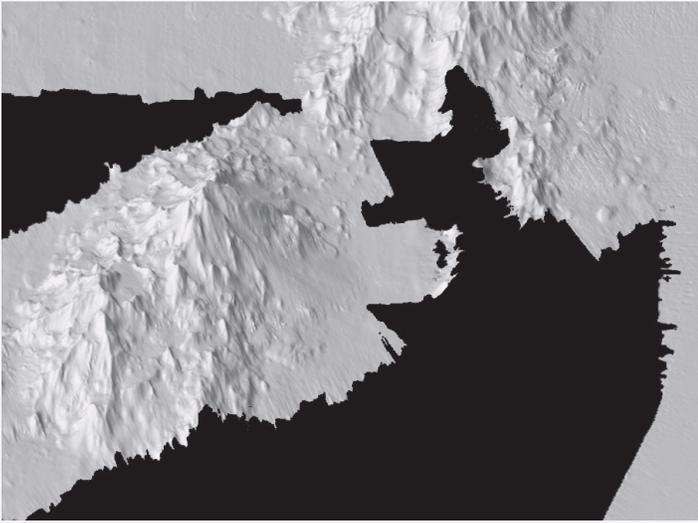


Figure 4: Filling gaps in the SOEST data (black area, left) involved filling in areas with coarser satellite altimetry data from a NOAA DEM (right).

Science and Technology (SOEST). It offers multibeam bathymetry obtained from many surveys merged with USGS Digital Elevation Model (DEM) coverage on land at a spatial resolution of 1.8 arc seconds (43 meters). The SOEST data are far from perfect, however. Gaps in the

data account for 12 percent of the total area of the *Seafloor Map of Hawai'i*. These include narrow strips next to the island shorelines and larger gaps in deep waters on the map periphery (Figure 3, black areas).

DATA MANIPULATION

FILLING THE GAPS in the SOEST data was achieved using methods similar to those used with Shuttle Radar Topography Mission (SRTM) data. This popular elevation dataset of land areas on Earth is also plagued with numerous gaps, vexing cartographers who need to make seamless shaded relief maps. There are two methods of plugging the gaps: by interpolating nearby elevation values or using a second dataset, if one is available (Tait 2010). For the SOEST data, a second DEM developed for the NOAA Tsunami Inundation Program (National Geophysical Data Center 2013b) provided the solution. This DEM, a composite made from multiple sources to provide unbroken seafloor coverage, also had problems. Terrestrial elevation data were absent. More troublesome, the overall quality and resolution of the NOAA DEM were less than that of the SOEST data. For example, shallow areas derived from multibeam data displayed terracing artifacts.

Gaps were filled by rendering both data sources as shaded reliefs and compositing them in Photoshop. A layer mask on the SOEST relief allowed the NOAA relief to show through where there were gaps, thus creating a shaded relief with continuous coverage. Feathering the edges on the layer mask diffused the abrupt seams between the

two reliefs. In deep areas, giving the satellite altimetry data a barely perceptible pebble texture further facilitated the blending (Figure 4).

Even where no gaps existed, the SOEST data required considerable manipulation to make it presentable on the *Seafloor Map of Hawai'i*. The main issue was noisy artifacts embedded in the multibeam bathymetry, the result of surveys conducted over several decades merged as a single dataset. The “Frankenstein” stitches between these different data are often more noticeable than the topography itself, especially on abyssal plains (Patterson 2008). Even data deriving from a single multibeam survey often has a noisy texture that disguises subtler seafloor features.

Generalizing these noisy data made it more acceptable for mapping. Reducing the resolution from 1.8 to 6 arc seconds (43 to 144 meters) and applying smoothing to the data in Natural Scene Designer Pro removed most artifacts (Figure 5). Eliminating the larger artifacts required additional manual touchups. In Adobe Photoshop, placing a blurred copy of the relief on a layer below the original and painting repeatedly with a soft brush on a layer mask erased the worst imperfections. Challenges with this technique

were distinguishing seafloor features from artifacts and minimizing damage to features when making touchups. When the identity of an artifact was in doubt, the compromise solution was to diminish instead of remove it. A technique similar to that described above was used to make CleanTOPO2, a small-scale bathymetry dataset of Earth (Patterson 2013).

Fortuitous circumstances also minimized the visual impact of coarse-resolution data on the final map, which occurs mostly in the southwest corner of the map. This area is now conveniently covered up by the map legend. Elsewhere on the map, notes inform readers where areas of generalized seafloor data occur.

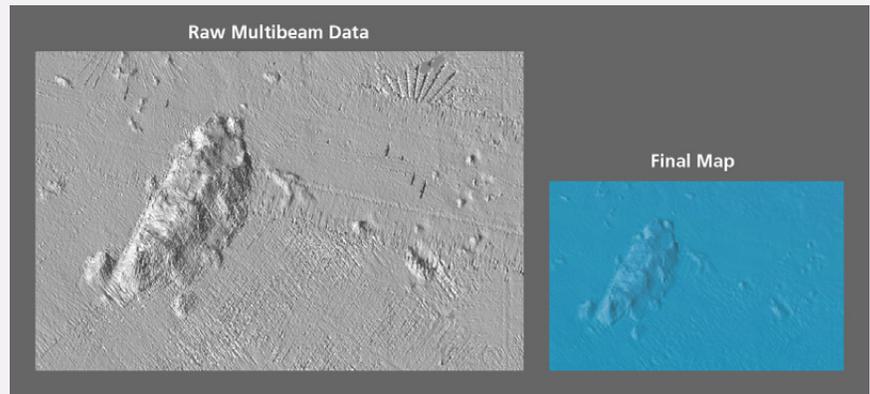


Figure 5: Raw multibeam data contain numerous artifacts (left) that data manipulation partially removed (right). Printing the final map at smaller scale and in blue further minimized the artifacts.

DESIGN OVERVIEW

MAKING THE *SEAFLOOR MAP OF HAWAII* posed design challenges that apply to all seafloor maps regardless of scale. Unlike other remote and inhospitable areas on Earth, such as high mountains visible from valleys below, maps of the seafloor depict places that will never be fully seen in their entirety. Light penetrates ocean water to a depth of 200 meters, and perpetual darkness cloaks what lies beyond. What is seen of this dim, alien world is limited to close-up glimpses from deep-sea submersibles. How then should the seafloor appear on a map when we can only imagine what it looks like?

One design approach is to use cartographic realism, which draws inspiration from natural world observations for depicting physical features on maps (Patterson 2002). For



Figure 6: Detail of the Seafloor Map of Hawaii without most labels. The seafloor east of Hilo, composed of pillow lava, is more rugged than the broad shield volcanoes that characterize the land.

example, tree canopies are green, as are forested areas on maps. What hampers applying cartographic realism to seafloor maps is our limited exposure to views of the seafloor. Mud flats and sandy shoals primarily exposed at low tide suggest that the ocean bottom is uniformly soft and all its relief variation is gentle, and should be reflected as such on maps. The reality is not so simple, however. Although the continental shelves and abyssal plains where sediments collect are indeed gentle, many other areas are not. For instance, the undersea cliffs, canyons, seamounts, and lava slopes near the Hawaiian Islands often exceed the ruggedness of adjacent areas on land (Figure 6). Portraying these features on the *Seafloor Map of Hawaii* required a change from the “land-dweller” mindset. A gently sloped ocean bottom would not do.

Color is another topic of debate for seafloor mapping. Using rainbow colors such as those found on *Hawaii’s Volcanoes Revealed* (Figure 2, right) is one approach that has been traditionally used if visualizing nuanced depths trumps all else. Given the lack of light in the deep ocean, and the tendency of dark tints to be perceived as lower in elevation, one could rationalize portraying the bottom as dark gray and black. Beyond making an avant-garde design statement, the problem with these monochromatic tints is having only 256 grayscale levels to depict depths ranging from 0 to -10,900 meters. Graphically speaking, it is a stretch. Printing black shaded relief on black depth tints also presents challenges.

As a practical matter, most ocean bottom maps, including the *Seafloor Map of Hawai'i*, depict depths with gauzy blue tints progressing from light shallows to dark depths (Figure 6). Slightly varying the hue alleviates the problem of having enough tonal range to represent the depth range. A light to dark blue color sequence mimics the way we see deepening water from boats, bridges, or when swimming. There are also graphical advantages. Because blue is the

most visually recessive hue and dark values appear lower, readers will likely perceive the bathymetry as occupying the lowest areas on a map. Blue depth tints blend harmoniously with gray shaded relief. Additionally, in terms of attracting an audience, it helps that—according to a Rutgers University study—blue is overwhelmingly the most popular color (2013).

HEINRICH BERANN'S INFLUENCE

THE LAST FIFTY YEARS have seen a sizeable number of small-scale ocean bottom maps published for general audiences, which provided design references for making the *Seafloor Map of Hawai'i*. The pioneer in this effort, Austrian artist Heinrich Berann, painted a series of ocean maps in the 1960s and 1970s for *National Geographic* and the US Navy based on data compiled by Marie Tharp and Bruce Heezen (Lawrence, 1999). It is perhaps not a coincidence that the artist who popularized alpine panoramas, a map genre that merges with landscape painting, set the early standard for seafloor maps (Figure 7).

Berann drew on his artistic background to paint highly distinctive ocean bottom maps (1968; 1977). His minimalist color palette presents a world of contrast: land areas basking in sunshine set starkly against the blue-gray ocean bottom, a somber underworld. Depth tinting is barely present. Berann instead paints the mid-ocean ridges, rents in Earth's crust producing new seafloor rock, in dark gray, in contrast with the ocean basins that are lighter blue. The three-dimensional topography exaggerates features on the ocean bottom more so than those on land. The ocean appears as if drained of water, exposing its chiseled continental shelves, deeply etched canyons, soaring seamounts, and fractured mid-ocean ridges to the reader.

Of the 12 ocean-bottom maps painted by Berann, eight use a light source originating from the southeast (lower right) to illuminate seafloor features, the rest use a southwest (lower left) light source. Like his alpine panoramas, illumination striking seamounts and other high features casts shadows across the seafloor. This is of course impossible so far below the surface, but so too is seeing the ocean bottom without water. The upcoming section on bathymetric tints discusses an alternative to depicting drained oceans.

More recently *National Geographic* has modified Berann's style, led by Tibor Tóth, a former staff artist and now retired freelancer for that organization. For several years conventional shaded relief had replaced three-dimensional relief depiction on *National Geographic* seafloor maps (Figure 7, third from top), but now three-dimensional relief is popular again (Figure 7, bottom). Compared to Berann's pieces, Tóth employed less vertical exaggeration, used brighter colors, and did not darken the mid-ocean ridges.

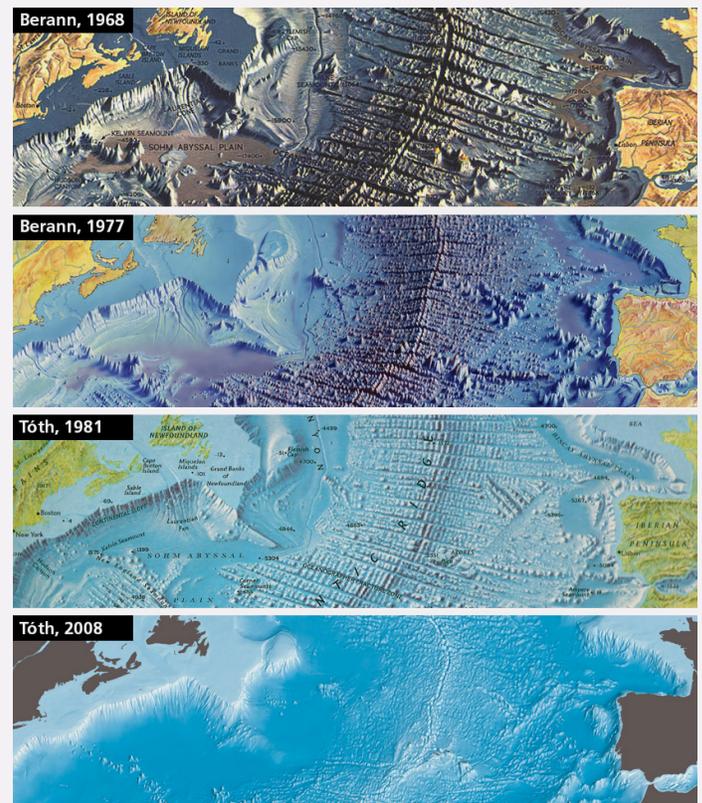


Figure 7: Small-scale ocean bottom maps. Examples courtesy of *National Geographic*, except for Berann, 1977, published by the US Office of Naval Research.

Tóth's work also shows Berann's strong influence, including restrained use of depth tints, the presence of cast shadows (Tóth applies these with a lighter touch), and a preference for southeast lighting, although he acknowledges that southwest illumination works just as well (Tóth 2009). More significantly, Tóth also painted seafloor topography. His method in recent years was to render digital bathymetry as a three-dimensional relief and then paint over it in Photoshop using a Wacom tablet and stylus (Tóth

2008; 2009). He painted not of preference but because of the necessity to remove imperfections or to add detail to poor-quality bathymetry data, especially at larger scales.

Although the design of the *Seafloor Map of Hawai'i* borrows heavily from both Berann and Tóth, it differs from them most noticeably in using only digital methods for relief presentation.

PLAN OBLIQUE RELIEF

PLAN OBLIQUE RELIEF is a type of projection that gives an impression of three-dimensional relief rendered from digital elevation models (DEMs). One of the parameters of plan oblique relief projection is a vertical offset, which produces an appearance similar to the terrain depicted in panoramas. It depicts mountains projecting upwards toward the top of the map and valleys downward (Jenny and Patterson 2007). By contrast, conventional shaded relief assumes that the position of the reader is directly above the map and depends entirely on light and shadows to model the terrain features (Figure 8).

There are tradeoffs to using plan oblique relief. On the positive side, it brings drama and realism to maps. Readers see terrain with a vertical dimension and in partial profile much like mountains appear from a scenic overlook or out of an airplane window. The relatively simple undersea

topography of Hawai'i—conical seamounts, blocky landslide debris, and steep-sided terraces around the islands—is suited to plan oblique relief presentation. These features appear to pop up from the ocean floor.

Lighting is another advantage. Because plan oblique relief illumination comes from either the front left or front right, compared to back left with conventional shaded relief, shadows fall on slopes facing away from the reader. This makes the terrain and overall map lighter, improving the readability of labels. The lighter relief also combines well with dark bathymetric tints in deep areas.

On the down side, plan oblique relief can hide parts of a map. As in any 3D image, there is a front side and back side to objects. In the case of plan oblique relief, south slopes face the reader and are more visible than steep north

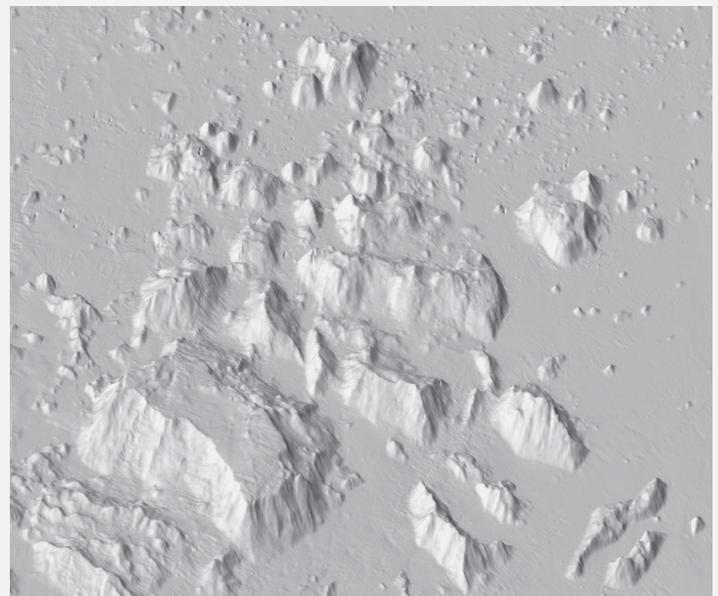
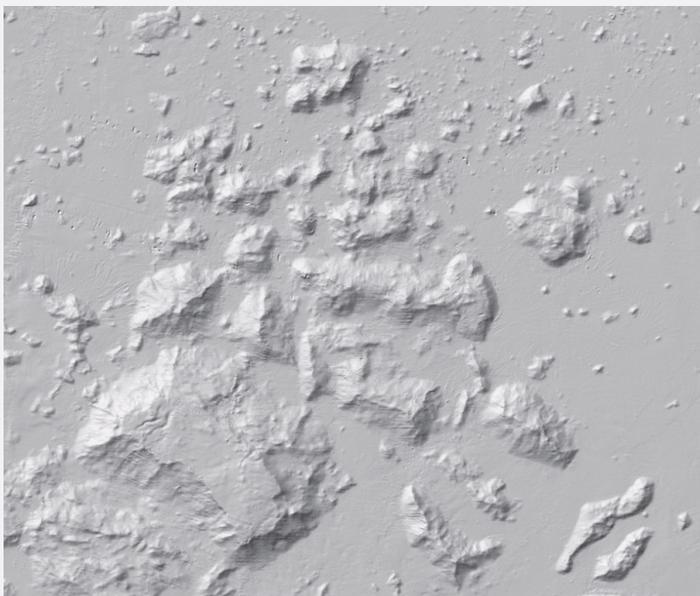


Figure 8: Conventional shaded relief of the Nu'uanu Slide (left) compared to plan oblique relief (right). Note that illumination originates from the upper left on the shaded relief and lower left on the plan oblique relief.

slopes, which can disappear from view. This issue, however, is less critical on the empty seafloor than on land as there are no roads, towns, rivers, etc. to obscure. In addition, seafloor features do not experience aspect-related weathering. Unlike terrestrial mountains in the northern hemisphere that have dramatic north faces due to glaciers, seafloor features are apt to be equally interesting on all sides.

Map lines are another problem with plan oblique relief. For example, latitude and longitude lines on a plan oblique relief map would not appear as a grid but would mirror the topography, going up and down with the changing elevation. Because a rectangular grid would only apply at sea level, a small area on the *Seafloor Map of Hawai'i*, it is not used. A related problem is that map georeferencing and reprojection is not advisable. For example, transforming plan oblique relief from a cylindrical to conic projection would tilt the three-dimensional topography inward. Like the panoramas that they mimic, plan oblique relief maps are pictorial.

BATHYMETRIC TINTS

THE *SEAFLOOR MAP OF HAWAII* employs very conventional bathymetric tints (depth colors) starting with light blue-green in the shallowest water and progressively darkening to gray-blue in deep areas (Figure 9). Class breaks occur at 1,000-meter intervals, except for extremely shallow and very deep areas that tend toward flatness and therefore have more class breaks for better definition. Depth tints on the *Seafloor Map of Hawai'i* blend continuously into one another, and combine with the relief shading and illumination. The point of the map is to emphasize relative depths.

Selecting tints to represent depths can challenge mapmakers accustomed to working with hypsometric tints. Unlike on land where high elevations are relatively rare, much of the world ocean consists of very deep basins. On the *Seafloor Map of Hawai'i*, for example, the maximum depth is -5,795 meters and depths between -4,000 and -5,000 meters predominate. Finding a pleasing blue tint to represent this depth class was critical, while avoiding gaudy or excessively dark hues that would overwhelm the map when viewed at full size. On the other hand, because the map is distributed online and appears as one of many thumbnail images on search pages, selecting eye-catching colors was an important consideration.

The *Seafloor Map of Hawai'i* uses plan oblique relief rendered with a beta version of Natural Scene Designer Pro 6.0. The pitch setting in this software application, ranging from -10 to -90 degrees, determines the amount of vertical exaggeration. For the Hawaiian seafloor, a pitch setting of -25 degrees yielded more vertical exaggeration than the default -45 degrees, but much less than that found on Berann's smaller-scale maps. Light originates from the west-southwest (245 degrees) at an angle of 45 degrees above the horizon. This light direction purposely departs from Berann's favored direction from the southeast (lower right). By selecting a light source closer to that of conventional shaded relief (northwest or upper left), the intent was to present the relief in a manner most familiar to readers. The Hawaiian Islands trending from southeast to northwest also render very well with perpendicular light coming from the southwest. The light angle at 45 degrees above the horizon generates fewer shadowed slopes than those of Berann's maps. Cast shadows also are not present on the *Seafloor Map of Hawai'i*, giving it a lighter overall appearance.

Compared to the subdued hues favored by Berann, the *Seafloor Map of Hawai'i* is brighter and more saturated, partially in consideration of the geographic area being mapped. Hawai'i is not the North Atlantic. A more significant departure from Berann, and to a lesser extent his successor at *National Geographic*, Tibor Tóth, is the prominent use of bathymetric tints. This is largely due to advances in map production. Whereas compositing blended

Depth in meters	RGB color	CMYK color
Sea level to -100	226r 240g 241b	10c 1m 4y 0k
-100 to -200	198r 229g 231b	21c 1m 8y 0k
-200 to -500	170r 208g 219b	33c 7m 11y 0k
-500 to -1,000	143r 195g 217b	42c 10m 9y 0k
-1,000 to -2,000	115r 179g 207b	53c 15m 11y 0k
-2,000 to -3,000	95r 169g 204b	61c 19m 10y 0k
-3,000 to -4,000	75r 159g 195b	69c 23m 13y 0k
-4,000 to -5,000	56r 148g 185b	75c 29m 16y 0k
-5,000 to -5,500	43r 138g 166b	80c 33m 26y 1k
-5,500 to -6,000	42r 130g 152b	81c 36m 32y 3k

Figure 9: Bathymetric tints used on the *Seafloor Map of Hawai'i*. The beige background in the illustration is the island color on the map.

bathymetric tints and the modulated light and shadows of shaded relief is easy to do with digital technology, a traditional artist would face challenges. There are just too many constantly changing colors to depict accurately with an airbrush or paintbrushes.

Now that digital production is the norm, combining bathymetric tints and shaded relief offers both conceptual and practical advantages to mapmakers. Shades of blue becoming darker with depth suggest an ocean filled with translucent water rather than drained (Figure 6). Consequently, readers see a less extreme departure from reality. Darker bathymetric tints in deep areas accentuate the aerial perspective effect, enhancing the apparent three-dimensionality of topography (Imhof, 1982). These darker tints also

disguise shaded relief created from poor-quality data in deep waters.

Land areas on the *Seafloor Map of Hawai'i* are muted beige, which downplays the importance of the islands compared to the ocean bottom. The beige nevertheless has enough warmth to provide visual relief from the cool blues everywhere else on the map (Figure 9). Shaded relief on land is slightly blue-gray, which softens its appearance. Lowlands received slight darkening to accentuate figure-ground contrast between the island shapes and shallow water, eliminating the need for shoreline casings. In fact, lines are entirely absent from the map. The island of Hawai'i, which is comprised of shield volcanoes with gentle slopes, received extra shaded relief darkening.

MAP FINISHING

THE *SEAFLOOR MAP OF HAWAII* comes in two versions, one with spot depths and elevations indicated in meters, and the other in feet. A liberal sprinkling of spot depths focuses attention on key seafloor features, such as seamount summits and deep troughs. The map also identifies island high points. Draping NOAA (2013) nautical charts with depth soundings on the rendered plan oblique relief provided a placement guide. This method, however, often proved inadequate because soundings on the nautical charts did not precisely coincide with relief generated from bathymetric data, which is presumably more accurate. For example, a spot depth might fall mid-slope on a seamount instead of its highest point. Sampling the bathymetry data to obtain spot depths proved a better technique.

Labeling undersea features proved more difficult than those on land. The National Geospatial-Intelligence Agency (2014) maintains the GEONet Names Server for US and international waters, but the coverage is sparse for Hawai'i. Altogether the GEONet Names Server accounts for 39 of the 84 undersea place names found on the *Seafloor Map of Hawai'i*, mostly seamounts southwest of the island of Hawai'i. The remaining names are unofficial, largely taken from the USGS map *Hawaii's Volcanoes Revealed*. They typically describe physiographic features and geologic events, such as the Clark 1 Slide west of the island of Kaho'olawe (Figure 10). Other geology-related names include the Moloka'i Fracture Zone, Southwest O'ahu Volcanic Field, Hawaiian Arch, and numerous ridges, slides, and slumps.

Researching undersea feature names for the map revealed varied terminology. For example, some maps identified the Hawaiian Trough, a region of extremely deep water adjacent to the Hawaiian Islands, as the Hawaiian Deep or Hawaiian Moat. In this case, the GEONet Names Server identified the feature as a trough, which settled the decision. For names not on the GeoNet server, a helpful reference was "Policies and Guidelines for the Standardization of Undersea Feature Names," a document published by the US Board on Geographic Names (2005), which lists definitions of undersea feature designations.

Many large undersea features don't have names. The massive Nu'uuanu Slide northeast of O'ahu consists of more than a dozen mountain-sized fragments (Figure 6), only one of which has a name. Even more conspicuous, no name was found for the large seamount northwest of the Kaua'i (partially cropped on the left map margin) that rises almost to the surface. The opposite problem, too many place names, occurs only off the north shore of Moloka'i. Of the one dozen submarine canyons found here, tight space on the map permitted the labeling of only two. Many Hawaiian seafloor features take their names from adjacent places on land. For example, off the northeast coast of Maui, Pa'uwela Ridge takes its name from a point, and Hāna Slump and Hāna Ridge from a town (Figure 10). These appear on the *Seafloor Map of Hawai'i* with diacritical punctuation as approved by the US Board on Geographic Names (2013) for their terrestrial namesakes.

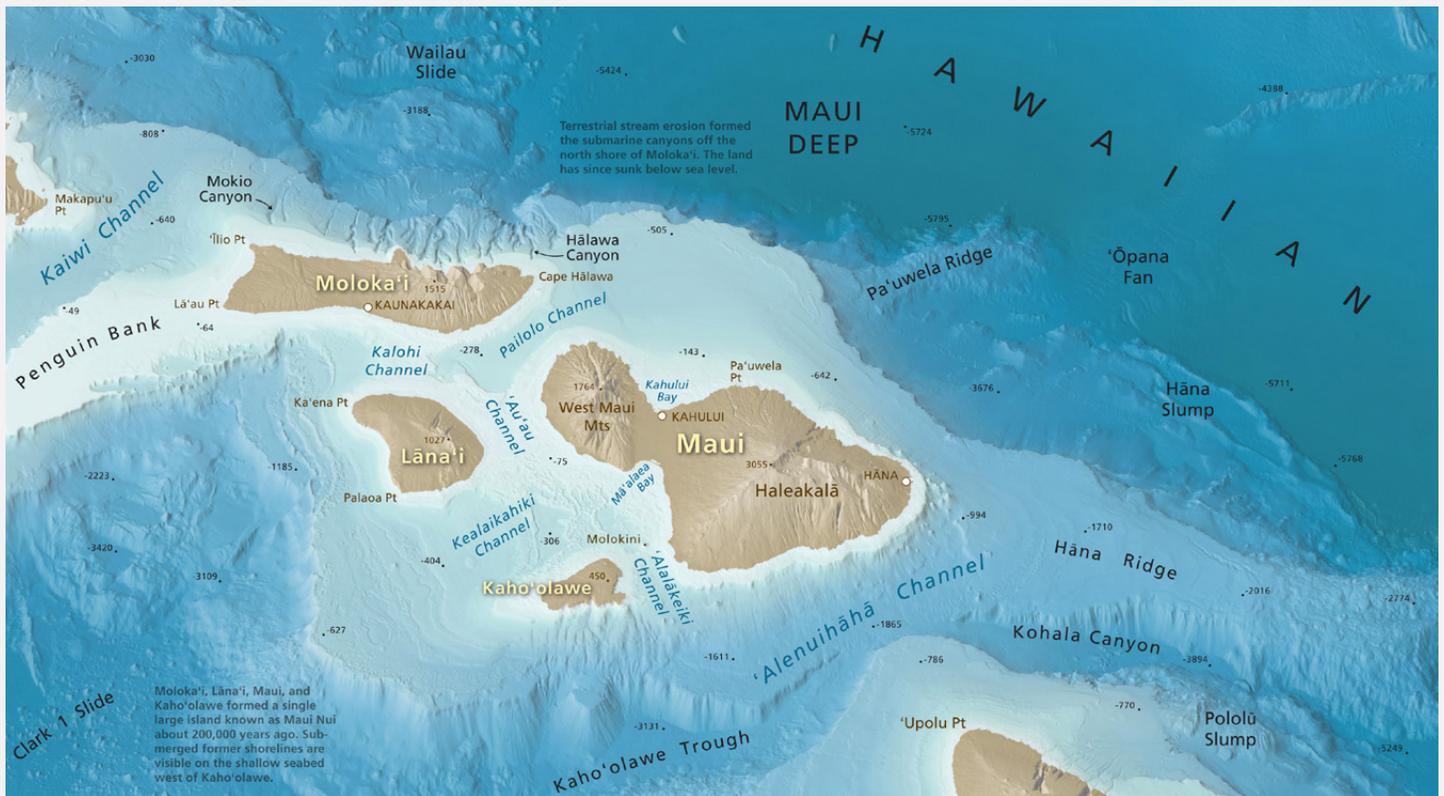


Figure 10: Excerpt of the Seafloor Map of Hawai'i with labels and metric spot depths.

Finally, the *Seafloor Map of Hawai'i* uses the Plate Carrée or Geographic projection. The bathymetric data used to make the map were originally in this cylindrical projection.

Since maps of tropical areas at medium scale have little distortion regardless of the projection used, changing it to another projection was unnecessary.

CONCLUSION

THE *SEAFLOOR MAP OF HAWAII* represents a map type that will slowly become more common with the ongoing collection of high-resolution bathymetry data. It demonstrates that with digital production it is now possible to make medium-scale seafloor maps that take inspiration from the small-scale ocean maps hand painted by artists over the last 50 years. More significantly, digital methods bring unrivalled detail to ocean floor maps and expand the design possibilities. For example, merging bathymetric tints with plan oblique relief brings color and realism to undersea maps that will likely appeal to general audiences. The intent is for readers to see the Hawaiian Islands as an enormous mountain range.

The compilation of this map also calls attention to the problem of incomplete and suboptimal bathymetry data that require considerable manipulation to become presentable on a map. As with all types of mapping, as scale and

detail increase, so does the magnification of data problems. The solutions employed by cartographers to fill voids in SRTM data in mountainous areas also apply to bathymetry data. Yet even with these repairs the data are often too irregular and noisy for clean map presentations. The available solutions are not ideal: smoothing bathymetry data, reducing the map scale, printing shaded relief on the seafloor lightly, and emphasizing depth tints. As a last resort and if time permits, touching up the data manually yields a much improved map presentation.

An irony of nearly all seafloor maps, including the *Seafloor Map of Hawai'i*, is the inevitable presence of land somewhere on the map. Perhaps to be meaningful to land-bound humans, this is a necessary requirement. It nevertheless brings up the question of how to design seafloor maps without land, which involves issues such as ubiquitous blue tones, a lack of known names, and no land to provide a

frame of reference. Considering that most of Earth is underwater, and how few large- and medium-scale maps of

the seafloor exist, cartographers still have much work to do—and many discoveries to make.

This article originally appeared in the Proceedings of the 8th ICA Mountain Cartography Workshop, September 1–5, 2012, Taurewa, New Zealand.

REFERENCES

- Becker, J., D. Sandwell, W. Smith, J. Braud, B. Binder, J. Depner, D. Fabre, et al. 2009. "Global Bathymetry and Elevation Data at 30 Arc Seconds Resolution: SRTM30 PLUS." *Marine Geodesy* 32:355–371. doi:[10.1080/01490410903297766](https://doi.org/10.1080/01490410903297766).
- Berann, Heinrich. 1968. "Atlantic Ocean Floor." *The World of H. C. Berann*. Accessed September 28, 2013. http://www.berann.com/panorama/archive/index.html#Ocean_Floor_Maps.
- . 1977. "World Ocean Floor." *The World of H. C. Berann*. Accessed September 28, 2013. http://www.berann.com/panorama/archive/index.html#Ocean_Floor_Maps.
- Eakins B., J. Robinson, T. Kanamatsu, J. Naka, J. Smith, E. Takahashi, and D. Clague. 2003. *Hawaii's Volcanoes Revealed*. <http://pubs.usgs.gov/imap/2809/>.
- GEBCO. 2013. Accessed September 28. <http://www.gebco.net/>.
- Fitzpatrick G. 1987. *The Early Mapping of Hawaii*. London: Kegan Paul International.
- Hawaii Mapping Research Group. 2011. "Main Hawaiian Islands Multibeam Bathymetry Synthesis." *University of Hawaii at Manoa*. Accessed September 28, 2013. <http://www.soest.hawaii.edu/HMRG/Multibeam/index.php>.
- Imhof E. 1982. *Cartographic Relief Presentation*. Edited by H. J. Steward. Berlin: de Gruyter.
- Jenny B., and T. Patterson. 2007. "Introducing Plan Oblique Relief." *Cartographic Perspectives* 57:21–40. doi:[10.14714/CP57.279](https://doi.org/10.14714/CP57.279).
- Kaye G., F. and Trusdell. "Using GIS to estimate the total volume of Mauna Loa Volcano, Hawaii." Presented at the Geological Society of America, Cordilleran Section, 98th Annual Meeting, Corvallis, Oregon, 13–15 May, 2002.
- Lawrence D. 1999. "Mountains under the sea: Marie Tharp's maps of the ocean floor shed light on the theory of continental drift." *Mercator's World* 6:36–43.
- Moberly R. 1973. "Bathymetry and Shorelines." *Atlas of Hawaii*. Edited by R. W. Armstrong. Honolulu: University of Hawaii Press.
- National Geographic. 2002. *Hawai'i*. Washington, DC: National Geographic Society.
- National Geophysical Data Center. 2006. "2-minute Gridded Global Relief Data (ETOPO2v2)." <http://www.ngdc.noaa.gov/mgg/global/etopo2.html>.
- . 2013a. "Integrated Models of Coastal Relief." Accessed September 28. <http://www.ngdc.noaa.gov/mgg/coastal/coastal.html>.
- . 2013b. "NOAA Tsunami Inundation Digital Elevation Models (DEMs)." Accessed September 28. <http://www.ngdc.noaa.gov/mgg/inundation/>.
- National Geospatial-Intelligence Agency. 2014. "GeoNames WMS Viewer." Accessed July 15. <http://geonames.nga.mil/namesviewer>.
- NOAA. 2013. "Office of the Coast Survey." Accessed September 28. <http://www.nauticalcharts.noaa.gov>.
- Patterson T. 2002. "Getting Real: Reflecting on the New Look of National Park Service Maps." *Cartographic Perspectives* 43:43–56. doi:[10.14714/CP43.536](https://doi.org/10.14714/CP43.536).

- . 2008. “Creating a National Geographic-Style Physical Map of the World.” *Mountain Mapping and Visualization: Proceedings of the 6th ICA Mountain Cartography Workshop, 11–15 February 2008, Lenk, Switzerland*, 155–161. Zurich: ETH Zurich.
- . 2013. “CleanTOPO2” *Shaded Relief*. Accessed September 28. <http://www.shadedrelief.com/cleantopo2/index.html>.
- Raven Maps & Images. 1985. *Hawaii*. Medford, OR: Raven Maps & Images.
- Rutgers University. 2013. “How does Gender Color our Preferences and Decisions on the WWW?” Accessed September 28. http://comminfo.rutgers.edu/professional-development/childlit/gender_project/home.htm.
- Sager W., J. Zhang, J. Korenaga, T. Sano, A. Koppers, W. Widdowson, and J. Mahoney. 2013. “An Immense Shield Volcano within the Shatsky Rise Oceanic Plateau, Northwest Pacific Ocean.” *Nature Geoscience* 6:976–981. doi:10.1038/ngeo1934.
- Sandwell D., and W. Smith. 1997. “Exploring the Ocean Basins with Satellite Altimeter Data.” Accessed September 28, 2013. http://topex.ucsd.edu/marine_grav/explore_grav.html.
- Scripps Institute of Oceanography. 2013. “SRTM30_Plus: SRTM30, Coastal & Ridge Multibeam, Estimated Topography.” Accessed September 28. http://topex.ucsd.edu/WWW_html/srtm30_plus.html.
- Smith W. 2004. “Introduction to This Special Issue on Bathymetry from Space.” *Oceanography* 17(1):6–8. doi:10.5670/oceanog.2004.62.
- Tait A. 2010. “Some Useful Digital Elevation Datasets.” *Cartographic Perspectives* 67:63–74. doi:10.14714/CP67.116.
- Tanaka K. 1950. “The Relief Contour Method of Representing Topography on Maps.” *Geographical Review* 40:444–456. doi:10.2307/211219.
- Tóth T. 2008. “Accidental Cartographer.” *Mountain Mapping and Visualization, Proceedings of the 6th ICA Mountain Cartography Workshop, 11–15 February 2008, Lenk, Switzerland*, 241–247. Zurich: ETH Zurich.
- . 2009. “Ocean Floor Relief.” Accessed February 12, 2012. http://web.me.com/tgraphix/Tóth_Graphix_Site/Tóth_Graphix_Blog/Entries/2009/2/8_OCEAN_FLOOR_RELIEF.html (site no longer exists).
- US Board on Geographic Names. 2005. “Policies and Guidelines for the Standardization of Undersea Feature Names.” Accessed September 28, 2013. earth-info.nga.mil/gns/html/GUIDELINES05.doc.
- . 2013. “United States Board on Geographic Names.” Accessed September 28. <http://geonames.usgs.gov/domestic/index.html>.

Identification of Partially Occluded Map Symbols

Jari Korpi
Aalto University
jari.korpi@aalto.fi

Mohammad Haybatollahi
Aalto University
mohammad.haybatollahi@aalto.fi

Paula Ahonen-Rainio
Aalto University
paula.ahonen@aalto.fi

Maps should be legible at all scales, and the information density of a map should be adapted to fulfill this goal. However, there are situations in which overlapping symbols might not be easily avoided. These kinds of cluttered or over-plotted situations often occur today in geovisual analytics and in map mash-ups created using Web 2.0 technologies. In this research project, we examine via a user test the extent to which occluded symbols can still be identifiable. Specifically, we tested how different levels of occlusion affected the accuracy and response time of finding symbols that varied in either color hue, abstract shape, or pictogram. The results of the test show that the efficiency of the symbols decreases when the symbols become partially occluded. Still, even half-occluded complex shapes can be identified quite accurately. Symbols varying in color hue seem to tolerate occlusion the best.

KEYWORDS: map symbols; symbol design; occlusion; overlapping symbols; visual variables; pictograms; map congestion

INTRODUCTION

THE DIGITAL MAPS USED in modern media consist of data that are created and updated continuously by web users with the help of Web 2.0 technologies (Graham 2010). This mass of data is often presented without cartographic editing, for example in simple map mash-ups, in which all data are displayed on a background map without any control over their overlap. This kind of map publishing easily leads to crowded maps containing numerous overlapping symbols. Some of these symbols cannot be identified at all, cluttering the map without conveying the intended information. This problem is common in maps produced using social networking tools (Field & O'Brien 2010).

Generalization is an essential part of mapmaking. It aims to reduce complexity in order to make the map legible and aesthetically pleasing. The collision or overlap of symbols is one of the conditions which determines *when* generalization is required (McMaster & Shea 1992). In map mash-ups, aggregating map symbols has been found to be a powerful method for solving the problem of overlap (Burigat & Chittaro 2008; Delort 2010). Filtering the data according to the relevance of the thematic objects is also

considered important for reducing the visual complexity of map displays (Swienty et al. 2008).

Map displays meant for geovisual analytics tend to be more complex than is cartographically appropriate (Kraak 2010). In exploratory analysis, the user browses large data resources and chooses the items to be shown on the map by querying a database. The user needs to be sure that all of the items fulfilling the query conditions are displayed, especially in cases where individual items are important to the analysis. In these situations, methods such as aggregation or automatic filtering cannot be used, even if the map display is cluttered, because they may hide necessary information and seriously mislead the user during the course of the analysis.

For cases in which methods that remove individual items cannot be used, displacement of the symbols remains one of the few acceptable methods of generalization (Korpi & Ahonen-Rainio 2013). However, it may not be possible to find space for a large number of non-overlapping symbols while keeping them close to their correct locations. In such situations, controlled overlap might be a solution,



© by the author(s). This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

in which symbols are allowed to overlap slightly, and only heavily overlapping symbols are displaced. One symbol peeking out from behind another symbol can make an important contribution to detecting a pattern during analysis. Allowing symbols to partially overlap on a map would represent a trade-off between the efficiency of identification and not losing the information. Then, an appropriate question would be: *What is a reasonable limit for the degree of occlusion?* For our purposes, a definite limit occurs when the symbols can no longer be efficiently identified, but only serve to clutter the map.

There has been little research on the perception of overlapping or occluded map symbols within the field of cartography. Groop and Cole (1978) have studied how accurately people can judge the sizes of partially occluded circles in proportional symbol maps. However, occluded object recognition has been studied quite extensively in the existing psychological literature. Evidence from such research suggests that early visual processes can handle occlusion information quite well (Rensink & Enns 1995), and this information is then available during the object recognition phase (Wolfe & Horowitz 2004). This suggests that we have mechanisms to recognize occluded objects quite easily. We are also capable of recognizing meaningful objects from very small and low-resolution thumbnail images (Torralba 2009). On the other hand, Rosenholz et al.

(2007) claim that human object recognition performance decreases due to occlusion.¹

However, we did not find that researchers have drawn any conclusions on the effect of the *degree of occlusion* that could help us answer our question about the efficiency of partially occluded map symbols. To bridge this gap, we organized a user test where the task was to search for and *identify* visible and partially occluded symbols on a map. In terms of our question about the reasonable limit of occlusion, the design of the symbols also has relevance because:

1. Some visual variables can be *perceived*, whereas others have to be *inferred* when partially occluded;
2. With regard to visual variable of *shape*, the complexity of the shape affects the symbol identification.

In the next section, we discuss related work on visual variables and occluded shape recognition. After that, we assess the above two claims more thoroughly and form hypotheses based on them. Following the theoretical section, a user test and its results are presented. Finally, we discuss the results in light of theories on occluded object recognition before drawing our final conclusions. The current study aimed at finding practical solutions for visualizing news data on an interactive map. We were especially interested in the capacity of pictograms in map symbols.

1. In cartography, the term *symbol identification* is used and has a meaning similar to *object recognition* in psychology (Keates 1989).

RELATED WORK

VISUAL VARIABLES

MAP SYMBOLS CARRY INFORMATION via visual variables, which were first introduced by Bertin ([1967] 1983). Bertin's seven variables were shape, size, hue, value, orientation, texture, and location. Other cartographers have suggested including additional variables, such as saturation, transparency, crispness, and resolution (MacEachren 1995). Psychologists have also studied the visual variables of objects and introduced slightly different ways of organizing them. For example, Kosslyn and Koenig (1992) categorized the variables as object properties (e.g., color, texture, shape) and spatial properties (e.g., size, orientation).

Different visual variables have different capabilities and serve different purposes when visualizing data. Bertin ([1967] 1983) classified visual variables on the basis of whether they are "selective" or not. For example, color hue is selective because it allows the eye to isolate all elements in one category and disregard other categories. Shape cannot do this and is, therefore, nonselective. Color hue is also the most efficient variable when a uniquely characterized symbol needs to be found on a map (Lloyd 1997). This kind of visual search has been extensively studied in cognitive psychology, and models have been built to describe human visual search mechanisms (e.g., Treisman & Gelade 1980; Wolfe 2007). Based on a number of visual search studies, Wolfe and Horowitz (2004) classified visual attributes

according to their ability to guide attention. They included color, size, orientation, and motion in the best, “undoubtedly” category. They included such variables as shape in the second best, “probable” category. Within Gestalt psychology, color hue is viewed as one of the strongest similarity grouping principles (Quinlan & Wilton 1998).

OCCLUDED SHAPE RECOGNITION

In natural scenes, most of the objects are partially hidden behind other objects. We are used to operating in this kind of an environment, and we can easily recognize objects based on their visible parts. We understand that partially visible objects lie behind other objects, and we do not need to interpret the visible part of the object alone as a whole. This ability is already gained by 2–4 months of age (Valenza et al. 2006). Since the perceptual completion of occluded objects is such an inseparable mechanism in the processes of object recognition, it has been well studied in psychology and neuroscience.

In the psychological literature, two different approaches, *local* and *global*, have been proposed for explaining the processes of recognizing partially occluded objects. Local theories state that people mainly connect the discontinuities between partially occluded contours by finding the simplest continuous function between the points of occlusion (Kellman & Shipley 1991), or on the basis of T-junctions (Rubin 2001). Global approaches suggest that shape regularities, such as symmetries, dominate during the perceptual completion of objects (van Lier 1999; de Wit et al. 2005). Evidence has also been found for the relevance of both influences (van Lier et al. 1995; Tse 1999).

Kellman (2001) separates global and local processes in object recognition. A global process—that is, when the recognition is based on global symmetry or on the familiarity of the object—is a higher level, top-down cognitive process compared to the perceptual local, bottom-up contour interpolation process. A global process requires that the observer have an idea what the occluded object looks like.

Also, the gestalt *past experience*² rule states that our visual perception is tuned to search for familiar objects in a particular scene (Wertheimer 1958). Furthermore, the context can reduce the required visible detail in object recognition. For example, a cat can be recognized from the very tip of its tail peeking out from behind a sofa, if it is already known beforehand that a cat is somewhere in the room (Kosslyn & Koenig 1992).

In a map context, the processes of object recognition can be reviewed by comparing abstract and pictographic symbols. The basic distinction is that a pictographic symbol mimics its object whereas an abstract symbol does not, which means that the identification processes for the two types of symbols differ from each other. Map symbols are identified by matching a symbol on the map against the symbols in the legend or by remembering the meaning of the symbol based on previous experience (Keates 1989). The meaning of an abstract symbol has to be learned beforehand so that the symbol can be correctly identified without using the legend, whereas the meaning of a pictographic symbol can be correctly inferred without previous learning. Because of this advantage, pictographic symbols are usually used in maps designed for novice or occasional users, such as tourists (Kostelnick et al. 2008).

A pictographic symbol loses its advantage of intuitiveness if the conceptual relationship between the symbol and the object it represents is not familiar to the map reader—that is, if the relationship has to be learned similarly to that of abstract symbols (Korpi & Ahonen-Rainio 2010), because it cannot be based on causal reasoning, which is typical of humans (Sloman 2009). Then, a pictogram is nothing more to the reader than a visually complex abstract symbol, and this complexity is the disadvantage of pictographic symbols. Humans can identify simple shapes significantly faster than complex shapes (Alluisi 1960). In a map context, readers can identify pictographic symbols more accurately than abstract symbols, but they can identify abstract symbols more quickly than pictographic symbols (Forrest & Castner 1985).

2. Some sources use the term *familiarity*.

THEORETICAL FRAMEWORK

WHEN CONSIDERING THE EFFICIENCY of occluded map symbols, the visual variable used to visualize the symbols and, more specifically, how the visual variables are rendered

on the display are important. Such variables as transparency, color hue, value, and saturation do not need contour information; rather, they can be assigned to each pixel, and

therefore we refer to them here as *surface-based* variables. Size, shape, orientation, crispness, and resolution cannot be assigned to each pixel but require the contour information in order to be visualized. Therefore, we refer to size, shape, orientation, crispness, and resolution here as *contour-based* variables.

In the case of overlapping symbols, surface-based variables can be *perceived* even when only part of the symbol is visible, but contour-based variables must be *inferred* based on the visible information. The map reader does not need to see the whole symbol in order to see its color. But when the reader tries to identify the shapes of partially occluded symbols, they cannot really be seen; rather, the reader has to rely on cues, such as the fact that a curved line is a part of a circle and that straight angles belong to a square. This suggests that, when they are partially occluded, identifying contour-based variables is more demanding than identifying surface-based variables. However, in practice map symbols are small and the visible areas of partially occluded symbols are obviously even smaller. The human ability to discriminate between color hues weakens when the field size gets smaller than half a degree of visual angle (Ware 2000). Therefore, color variables may not benefit from their perceptible character in the case of heavy occlusion.

USER TEST

WE DESIGNED A USER TEST to investigate the degree to which occluded symbols are still identifiable. Specifically, we tested *surface-based* versus *contour-based* visual variables and *abstract* versus *pictographic* symbols in the case of occluded symbols. To this end, we tested how effectively (accurately) and efficiently (fast) three symbol types—*color hue* (surface-based variable), *abstract shape*, and *pictogram* (contour-based variables)—could be identified on a map at different occlusion levels. The effectiveness and efficiency of completely visible symbols were compared to the effectiveness and efficiency of partially occluded symbols. The target symbol on a map appeared as either totally visible or as partially occluded by other symbols. The occlusion levels used in the test were chosen on the basis of the hypothetical limits of each case. For the shape-varied symbols (i.e., abstract shapes and pictograms), the tested occlusion levels were 25 and 50 percent; we assumed that in general, more shape information should be visible than missing, although

When considering the processes of occluded-object recognition, abstract symbols can be identified with local perceptual processes whereas the identification of pictographic symbols requires higher level cognitive processes. In terms of using the symbols on crowded maps, the question is then whether this further slows down the process of identifying pictographic symbols compared to that of identifying partially occluded abstract symbols. When thinking about it intuitively, pictographic symbols lose more of their efficiency when partially occluded. For example, Slocum et al. (2005) state that pictographic symbols might be more difficult to interpret than abstract symbols when the symbols overlap. On the other hand, some studies reveal that global processes are taken into account relatively early in the visual system (Sekuler et al. 1994), which suggests that pictographic symbols might not lose any more of their efficiency when partially occluded than abstract symbols do.

On the basis of the fact that surface-based variables can be *perceived* from partially occluded symbols and that contour-based variables have to be *inferred*, we formed a hypothesis that *surface-based variables cope with symbol overlap better than contour-based variables* (Hypothesis 1). On the basis of the need for cognitive processes to identify partially occluded pictographic symbols, we formed a hypothesis that *abstract symbols cope with symbol overlap better than pictographic symbols* (Hypothesis 2).

there may be strong individual differences between symbols in this respect (Kosslyn & Koenig 1992). The tested occlusion levels were 50 and 75 percent in the case of color hue, because we hypothesized that the occlusion tolerance would be higher with color hue. Bedford and Wyszecki (1958) found that color discrimination is still quite accurate with field sizes of 12 minutes of visual angle, which is slightly larger than 25 percent of the symbol size used in our test. The task of the subjects was to interpret a map that visualized news items with point symbols representing five different news categories, and to find a unique symbol on the map. The response time and accuracy of the responses were measured.

SUBJECTS

In total, 40 subjects participated in the test. Twenty-seven of the subjects were undergraduate students in

geoinformatics, while the rest were graduate students and researchers. The subjects represented a range of nationalities and educational backgrounds. The majority of the subjects were familiar with geographic information systems and spatial data, but their level of experience varied considerably. The age of the subjects ranged from 23 to 58. We tested the red-green color blindness of the subjects using an Ishihara test picture, and none of the subjects proved to have this kind of color deficiency.

TEST MATERIAL

The test maps consisted of a background map and 53 partially occluded or completely visible point symbols. The point symbols represented news items in five different news categories: each category had a different symbol, which varied either in terms of *color hue*, *abstract shape*, or *pictogram*. The five different symbol colors were based on the easily separable color schemes for qualitative data designed by Harrower and Brewer (2003). The abstract shapes and pictograms were designed for the test and drawn using Adobe Illustrator. The symbols used in the test are shown in Figure 1. The diameter of the symbols on the screen was 4 mm, an ordinary symbol size for digital maps and similar tests. The maps and legends used in the test were constructed using Avenza MAPublisher and the test software was implemented in Java. The user interface of the test software is shown in Figure 2.

In each case, one of the categories had only one news item on the map. This was the target symbol that the subject was supposed to find. The test question was: “Which one of the following symbols is presented only once on the map?” This question was used instead of letting the subject search for a given target symbol because in this way the subject had to examine the entire map area and the effect of target location could be minimized. In Figure 2, the

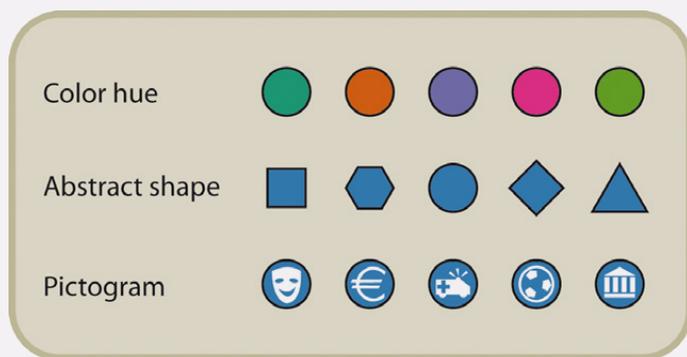


Figure 1: Symbols used on the test.

target symbol is the orange *economy* symbol (surrounded by a black square). Another news category appeared twice on the map. Below, we use the term “second best guess” for this alternative. The purple *culture* symbol in the figure is the “second best guess” (surrounded by a black ellipse). The other three categories appeared several times on the map (as the *politics*, *sports*, and *accident* symbols), and these were “incorrect alternatives.” The background map (Figure 2) was designed so that it would not confuse the subject’s perception of the thematic symbols. Water areas, main routes, and urban sprawls were represented with light colors, and no place names were given.

Since it was assumed that the task could be solved by ruling out symbols that appeared more than once on the map, the target symbol and one of the two symbols used for the “second best guess” were always equally occluded. This was done to ensure that the response was based on identifying the occluded symbols. Since the task was designed so that the subject had to examine the entire map area as a means of ensuring their response, it was assumed that the locations of the symbols on the map would not affect the results. For each occlusion level (0, 25, 50, and 75 percent), there were three alternative locations for the target symbol and the two symbols for the “second best guess.” All of the alternative locations were quite close to the center of the map, but the target symbol always appeared in different locations on each test map presented to the subjects in order to prevent them from learning about the locations. The two symbols used for the “second best guess” were always located close to, but not quite next to, each other.

The background map and the spatial distribution of the symbols were constant. This means that a specific location always contained a symbol, but the visual variable, the occlusion level, whether the symbol was a target symbol, and the location of the target symbol varied from map to map. Based on all the possible combinations, a set of 45 maps were constructed. These maps were divided into five subsets of 9 maps each; each set included all nine conditions—that is, they included three symbol types in three different occlusion levels. The test was a within-subjects design, meaning that all of the subjects saw all of the tested conditions. However, not all of the subjects saw the exact same maps, due to the variations in the target locations. Also, the order of the maps varied from set to set. A set size of 9 maps was chosen to keep the overall test short in terms of time since the test session also contained other test tasks that are not presented in this paper.

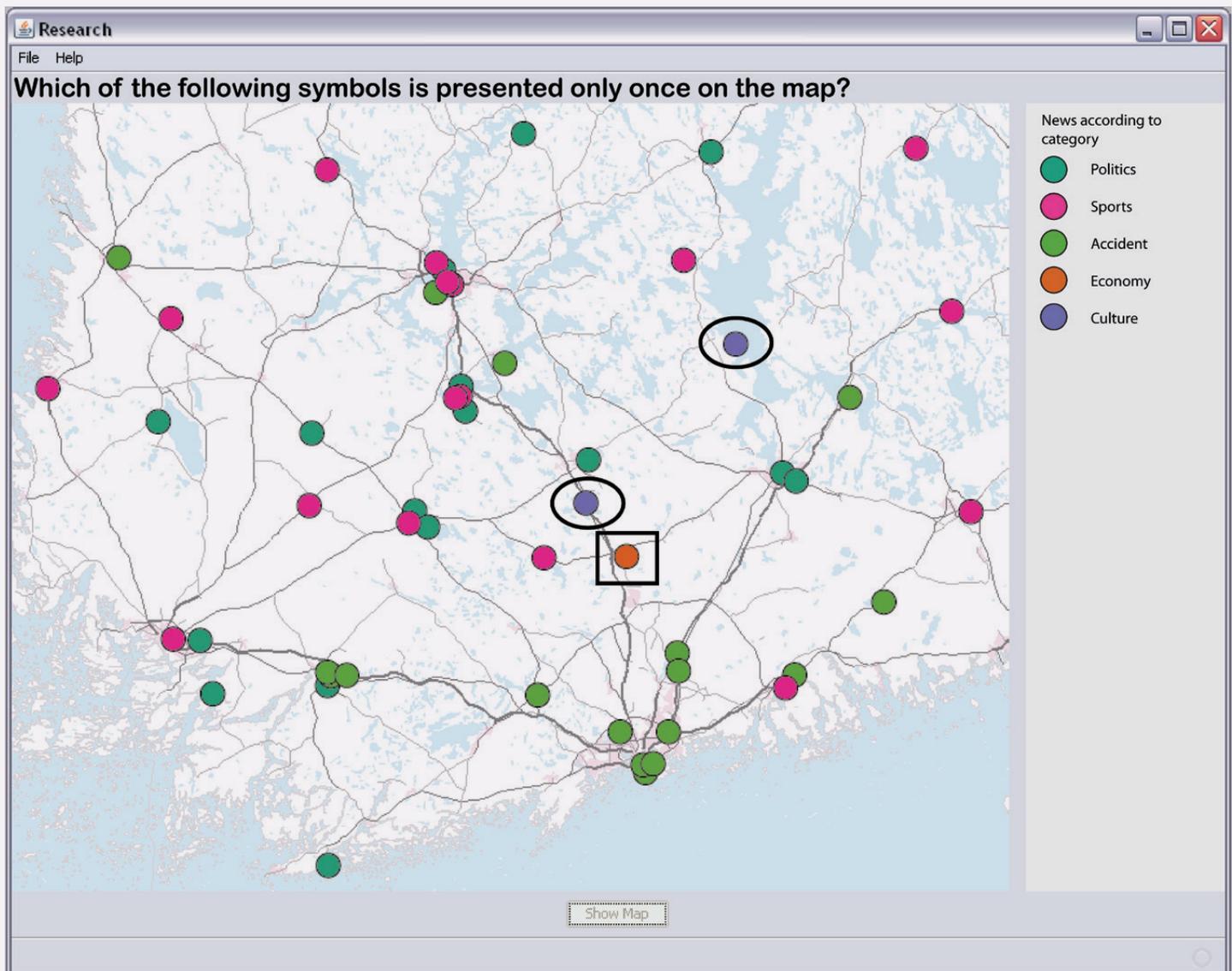


Figure 2: The user interface of the test software. A subject has opened the map and the timing has started. The subject must give a response by clicking on a symbol in the legend. In this case, the target symbol is totally visible. The black square and ellipses were not visible for the subjects.

TEST PROCEDURE

FIVE SUBJECTS at a time took the test with similar laptops in a meeting room. Each of the laptops had a different test set of 9 maps. The theme of the test maps, the task, and the test routine were first introduced to the subjects. The subjects were motivated to respond as quickly and as accurately as possible, since response time and accuracy were being measured. After the short briefing session, the

subjects were allowed to start the test at their own pace. During the test routine, the question and the legend showing the possible solutions were first displayed. By clicking a button, the subject rendered the map visible and activated the timer (Figure 2). The timer stopped when the subject clicked a symbol in the legend, which caused a new question with new possible solutions to be shown. The subject was able to rest after answering each question before making the next map visible.

RESULTS

To test the effects of the occlusion levels (0, 25, 50, and 75 percent) on the efficiency of three types of map symbols,

we used repeated-measure ANOVAs with IBM-SPSS Statistics version 21. The three symbol types shared two

Symbol (level of occlusion)	Mean (ms)	Correct		Incorrect	
		SD (ms)	N (%)	N (%)	N (%)
Color (0)	6973.55	3523.41	38 (95%)	2 (5%)	
Color (50)	9483.59	4071.59	40 (100%)	0 (0%)	
Color (75)	11681.18	5754.51	33 (85%)	6 (15%)	
Abstract (0)	15580.08	7003.90	36 (92%)	3 (8%)	
Abstract (25)	19826.43	8640.99	35 (95%)	2 (5%)	
Abstract (50)	23325.32	9724.47	38 (97%)	1 (3%)	
Pictogram (0)	25059.64	10705.85	33 (85%)	6 (15%)	
Pictogram (25)	26646.07	9913.96	27 (75%)	9 (25%)	
Pictogram (50)	33332.56	14781.93	34 (85%)	6 (15%)	

Table 1: Descriptive statistics.

common occlusion levels (i.e., 0 and 50 percent); they yielded the main data for comparing the decrease in the efficiency of the different symbol types when they were occluded. We analyzed these occlusion levels using two-way repeated measure ANOVA. Additionally, we tested the color hue when 75 percent of the symbol was occluded in order to assess the theoretical limit of occlusion tolerance, and we tested the shape variables when they were 25 percent occluded in order to analyze the effects of minor occlusion. Therefore, we also performed the analysis with one-way repeated measure ANOVA for each symbol type separately.

Prior to the analysis, we screened and checked the data for any violations with respect to assumptions about the analysis of variance (e.g., independence of cases, normality, and homogeneity of the variances). There were a few outliers in the data, which caused some of the variables to be slightly skewed and marginally violated the assumption of normality. Removing the influential outliers based on a procedure suggested by Tukey (1977) and Hoaglin et al. (1986) provided an acceptable level of normal distribution so that all variables could pass the Kolmogorov-Smirnov test of normality. In addition, we found that the residuals in all

variables coincided with the normal line in a P-P Plot and, thus, fit the assumption of normally distributed residuals. The variables also passed Mauchly's sphericity test concerning the two-way repeated measure ANOVA. The results of the analyses are presented in three stages below.

DESCRIPTIVE ANALYSIS

Table 1 and Figure 3 show the results of the comparative analysis done in the study. The results indicate that the majority of the responses were correct: They ranged from 100 percent to 75 percent. The subjects gave slightly more incorrect responses in the case of pictograms than for the other two symbol types, but the difference was not statistically significant. The accuracy of the responses did not seem to drop dramatically for any of the three symbol types at any of the tested occlusion levels. As reaction times of incorrect responses cannot be related to any particular cognitive or perceptual processes, we continued the analysis using only the correct responses.

For all three symbol types, the mean response times for correct responses followed a steady pattern of lengthening as the level of occlusion increased. In the case of color hue, the time difference in the mean response times between totally visible and half-occluded symbols was noticeably smaller (2.5 s) than in the case of abstract shapes (7.7 s) and pictograms (8.3 s). This supports the hypothesis that surface-based variables cope with symbol overlap better than contour-based variables but not the hypothesis that abstract symbols cope with symbol overlap better than pictographic symbols.

Comparisons	Mean Difference (ms)	Std. Error
Color with Abstract	8622.61*	1080.75
Color with Pictogram	18701.50*	1857.17
Abstract with Pictogram	10078.89*	1822.88

Table 2: Pairwise comparison of the mean at baseline level (totally visible symbols).

Note: N = 28 for all pairwise comparisons. * $p < 0.001$

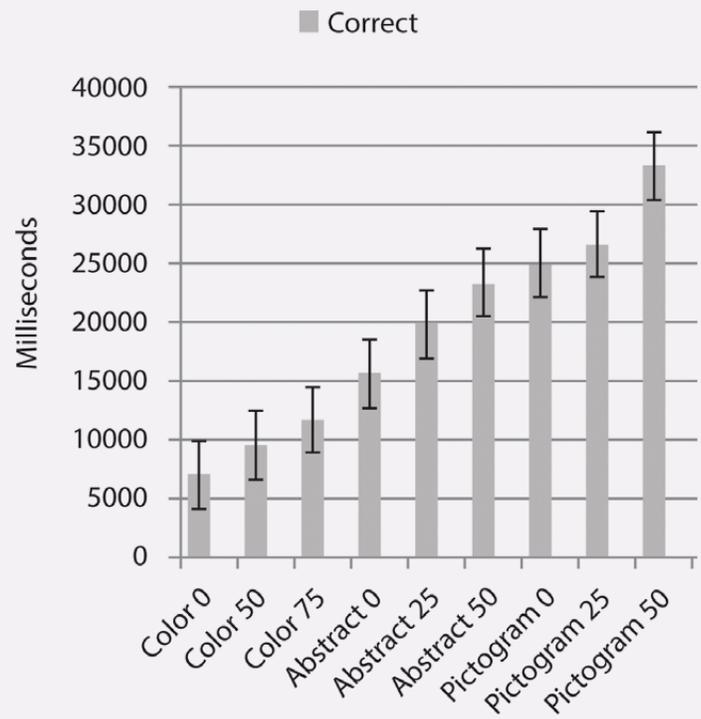
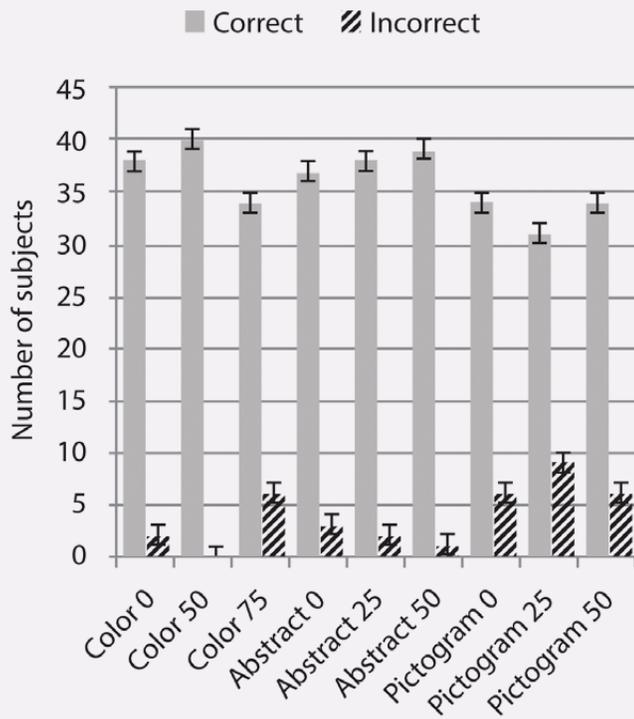


Figure 3: Graph comparison of the frequencies of the correct and incorrect responses (left), and the mean response times of correct responses (right).

EFFICIENCY OF TOTALLY VISIBLE VS. HALF-OCCLUDED SYMBOLS

Using two-way repeated measure ANOVA, we compared the response times at the baseline (i.e., totally visible symbols) between the three types of symbols. The results show significant differences between the three baseline cases for totally visible symbols ($F_{(2, 28)} = 60.86, p < 0.001$). Table 2 shows that the pairwise comparisons of the means for each of the three baseline cases differed significantly ($p < 0.001$). The results therefore denoted that in a totally visible situation, subjects identify color hue more efficiently than they do abstract shapes, and they identify abstract shapes more efficiently than they do pictograms.

In order to test the differences between the response times of the 50 percent occluded cases and the baseline, we implemented a two-way repeated measure analysis, which included two levels of occlusion (i.e., 0 and 50 percent) and the three symbol types. When it came to the levels of occlusion, the means of the response times at 0 and 50 percent occluded levels differed significantly from each other ($F_{(2, 22)} = 137.90, p < 0.01$). The post-hoc pairwise analysis indicated that the occluded levels (0 and 50 percent) differed significantly from each other for all three symbols types ($p < 0.001$). Therefore, the results support the findings of descriptive analysis in that occlusion significantly

decreases the efficiency of the map symbols, at least when the symbols are half-occluded. No significant interaction effect between the occlusion levels and symbol types was found ($F_{(2, 22)} = 1.08, p = 0.35$).

SEPARATE ANALYSES FOR EACH SYMBOL TYPE

Because the tested occlusion levels for color hue were different than for the other symbol types, we used one-way repeated measure ANOVA to perform separate comparisons for each symbol.

Color hue: The result showed that the level of occlusion significantly affected the efficiency with which subjects identified the symbols ($F_{(2, 31)} = 23.58, p < 0.001$); the means for all three occlusion levels (0, 50 and 75 percent) differed significantly.

Abstract: The result showed that the level of occlusion significantly affected the efficiency with which subjects identified the symbols ($F_{(1.46, 29)} = 8.02, p < 0.01$)³; post-hoc analysis showed that the means for all three occlusion levels (0, 25 and 50 percent) differed significantly from each other

3. In the case of abstract symbols, Mauchly's test indicated a violation of the assumption of sphericity ($\chi^2_{(17)} = 12.48, p < 0.01$). Therefore, the degree of freedom was corrected using the Greenhouse-Geisser estimates test.

($p < 0.01$). This result suggests that occluding abstract symbols by 25 percent decreases their efficiency significantly compared to totally visible symbols.

Pictogram: In the case of pictograms, we cannot reject the null hypothesis indicating that there are no differences between the three occlusion levels (0, 25 and 50 percent) ($F_{(2, 17)} = 1.29, p = 0.29$). When repeating the analysis with only two of the levels included in the one-way repeated measure ANOVA, the results indicated a significant difference between baseline and 50 percent occlusion ($F_{(1, 28)} = 23.58, p < 0.01$). In other words, the mean of the time that participants spent identifying the pictogram at a 50 percent occlusion level ($M = 30725.47$ ms) was significantly greater than the baseline ($M = 26068.84$ ms). This result is in line with the results for the *two-way repeated measure ANOVA*. No significant differences were found between the means of the responses at baseline with 25 percent occluded levels ($F_{(1, 22)} = 1.44, p = 0.24$) and at 25 percent with 50 percent occluded levels ($F_{(1, 22)} = 3.08, p = 0.09$). This result suggests

that occluding pictographic symbols at a 25 percent level does not decrease their efficiency significantly compared to totally visible symbols. The task of identifying pictograms in a display while visualizing dozens of symbols is difficult, and other factors may have affected the process of identifying pictograms and bedimmed the effect of occlusion. However, it seems that pictograms do not lose any more of their efficiency than abstract symbols when occluded, which is in line with the findings of the descriptive analysis.

We used one-way ANOVA to test whether different subgroups of participants performed differently based on having received different sets of maps. The only (marginally) significant difference was detected in case of totally visible symbols visualized with color hue ($F_{(4, 33)} = 3.48, p < 0.05$); post-hoc follow-up analysis showed that, in that case, those who saw set 3 were significantly slower than those who saw sets 1, 4, or 5. One-way ANOVA did not indicate other significant differences between the subgroups.

DISCUSSION

SURFACE-BASED VS. CONTOUR-BASED VARIABLES

OUR EXPERIMENTAL RESULTS suggest that the *efficiency* of the symbol decreases when the symbol is partially occluded, regardless of whether the visual variable is *surface-based* or *contour-based*. However, this decrease was not evident until the symbols were half-occluded, since we did not test 25 percent occluded color hue and the results for the 25 percent occluded pictograms lacked statistical significance. The results also show that the symbols can still be identified quite *effectively* (accurately) when half-occluded regardless of the visualization used. This suggests that the human ability to recognize occluded objects applies well to map reading.

On the basis of comparing the lengthening of response times from totally visible symbols to half-occluded symbols, the efficiency of *surface-based* variable color hue (2.5 s mean time) decreased less than the efficiency of *contour-based* shape variables (abstract shape 7.7 s; pictogram 8.3 s). This supports the first hypothesis: *surface-based variables cope with symbol overlap better than contour-based variables*. This means that while all symbols seem to tolerate occlusion, the visual variable used in the symbols indeed

affects the symbols' ability to maintain its efficiency while occluded. Furthermore, our categorization of *surface-based* and *contour-based* variables seems to be valid when assessing the visual variables' ability to tolerate occlusion, and there is a difference in efficiency between *perceiving* and *inferring* occluded map symbols in practice.

ABSTRACT VS. PICTOGRAPHIC SYMBOLS

The subjects located abstract symbols faster and slightly more accurately than pictographic symbols on the test maps, but no drop in accuracy occurred when the abstract or pictographic target symbols were occluded. In these two cases, the response times lengthened in similar fashion (abstract shape 7.7 s; pictogram 8.3 s) when the symbols were half-occluded. This suggests that the need for global processes when identifying partially occluded pictograms does not lengthen the response times any more than identifying partially occluded abstract shapes that only require local processes during the identification process; hence, the second hypothesis—*abstract symbols cope with symbol overlap better than pictographic symbols*—is not supported by the results. Therefore, in practice pictographic symbols can tolerate symbol overlap as well as abstract symbols. Our

results support the studies claiming that global processes dominate in object recognition, whereas the dominance of the local processes is not supported, at least when object recognition is applied to a complex map reading task. Another explanation for the efficiency of the pictograms when occluded is that the context facilitates the search because the process of identifying complex pictograms on a map is facilitated by the user's familiarity with a limited set of possible map symbols.

When designing the pictograms, the possibility of overlap means that it is at least as important to design the pictograms separate from one another as to make them characteristically recognizable. The pictograms used in our test were designed first of all to be separable from one another. It is obvious that if pictograms resemble one another, then users might not correctly discriminate between them and other symbols that are partially occluded. Five different pictograms appeared on the test maps, but in practice the number may be considerably higher. For example, the symbol sets used in crisis management include dozens of symbols (ANSI 2006; GICHD 2005). When the number of different pictograms increases, the task of designing the pictograms easily separable from one another becomes more complicated.

VISUAL VARIABLES IN MAPS USED FOR GEOVISUAL ANALYTICS

The results also support the findings of visual search studies on the sovereign efficiency of color hue (e.g., Wolfe & Horowitz 2004; Lloyd 1997). In our test, subjects were able to even identify a 75 percent occluded color hue more quickly than they could completely visible shapes. Similarly, our results support the findings of studies that subjects identify simple shapes more quickly than they do complex shapes (e.g., Alluisi 1960; Forrest & Castner 1985). In our test, subjects identified a 50 percent occluded abstract shape as quickly as they did a completely visible pictogram. However, our results differ from those of Forrest and Castner (1995) who found that subjects identified the pictographic symbols *more accurately* than they did abstract symbols. In our study, only five different symbols appeared on the map, whereas the maps used by Forrest

and Castner had several different symbols, which reduced the possibility of mistakes in our case.

In practice, our results stress the fact that efficient variables should be used on maps intended for geovisual analytics. However, efficient variables cannot be varied to represent many different values. Subjects can only discriminate between relatively few different color hues or abstract shapes, as shown in the difference in accuracy for the abstract symbols when comparing the results of this study to those of the study by Forrest and Castner (1995). In this respect, pictograms have no equal. For example, more than just a few different symbols are needed when using map symbols to represent news topics or news content. Therefore, the strengths of different visual variables should be used in the case of geovisual analytics. Our suggestion is to use the attention-guiding and selective variable of color hue in combination with an illustrative pictogram, so that the color hue represents the higher level category and the pictogram represents a specific sub-level category. In this way, the color will help guide the search (Wolfe 2007) and the map reader can more easily access the information represented by pictograms.

NEEDS FOR FURTHER STUDY

The number of test maps presented for each subject was relatively small because of time constraints imposed by other experiments in the test session. Therefore, the number of factors that we were able to test was limited, and two issues should be studied further. First, more occlusion levels would need to be tested to better determine the level of occlusion at which the loss of efficiency reaches a critical point. For example, we assume that the abstract and pictographic shapes would no longer be effectively identifiable at the level of 75 percent occlusion, but to be sure it would need to be tested. Furthermore, we could fully compare the performance of different visual variables with each other if we tested the same occlusion levels for all types of symbols. Second, we used color hue to represent a *surface-based* variable and shape to represent a *contour-based* variable. Although color hue outperformed shape when they were partially occluded, more variables need to be tested to categorically state whether this superiority of color hue extends to other surface-based variables.

CONCLUSIONS

THE RESULTS OF OUR TEST give evidence for decreased *efficiency* of map symbols that are partially occluded. Therefore, overlapping symbols should generally be avoided in cartography. On the other hand, our results suggest that symbols can be identified quite accurately when partially occluded. This means that valuable observations can be made on the basis of partially occluded symbols on maps intended for *geovisual analytics*. Therefore, instead of excluding some symbols from a map due to a lack of space, symbols in crowded locations can be arranged so that they partially overlap with one another.

The question of the *reasonable limit* for occlusion cannot be answered unambiguously on the basis of our test, since we did not reach an occlusion level where the effectiveness of the symbols dropped dramatically. Also, the map symbols' ability to tolerate occlusion depends on the visual variable used. *Surface-based* variables seem to outperform *contour-based* variables because the efficiency of color drops less than the efficiency of the shape variables when the symbols are half-occluded. Therefore, the maximum level of occlusion is also likely to be higher with surface-based

variables than with contour-based variables. In our test, the symbols were identifiable at all tested occlusion levels. The most occluded level was 75 percent with color hue and 50 percent with abstract and pictographic shapes. These percentages are likely close to the usable maximums for each case with a symbol size of 4 mm. This result underpins the capacity of color as an efficient visual variable and suggests that it should be used in maps for *geovisual analytics*, where the map display tends to become crowded.

While the visual variable used seems to affect the efficiency of a map symbol when partially occluded, the complexity of the shape does not. Readers can identify an abstract shape more efficiently than a pictogram on a map containing several symbols regardless of whether or not the symbols are partially overlapping or totally visible; however, partial occlusion does not further weaken the efficiency of pictograms compared to abstract shapes. The efficient visual variable of color hue should be used in combination with pictograms to help readers access the information represented by the pictograms.

REFERENCES

- Alluisi, E. A. 1960. "On the use of information measures in studies of form perception." *Perceptual and Motor Skills* 11:195–203. doi:[10.2466/pms.1960.11.2.195](https://doi.org/10.2466/pms.1960.11.2.195).
- ANSI. 2006. "ANSI INCITS-415 2006 Homeland Security Mapping Standard—Point symbology for emergency management." Washington, DC: American National Standards Institute.
- Bedford, R. E., and G. W. Wyszecki. 1958. "Wavelength discrimination for point sources." *Journal of the Optical Society of America* 48(2):129–135. doi:[10.1364/JOSA.48.000129](https://doi.org/10.1364/JOSA.48.000129).
- Bertin, J. [1967] 1983. *Semiology of Graphics: Diagrams, Networks, Maps*, trans. W. Berg. Madison: University of Wisconsin Press.
- Burigat, S., and L. Chittaro. 2008. "Decluttering of icons based on aggregation in mobile maps." *Map-based Mobile Services – Design, Interaction and Usability*, edited by L. Meng, A. Zipf, and S. Winter, 13–32. Berlin-Heidelberg: Springer.
- Delort, J.-Y. 2010. "Visualizing large spatial dataset in interactive map." *Proceedings of the 2nd International Conference on Advanced Geographic Information Systems, Applications, and Services*. St. Maarten, Netherlands Antilles.
- Field, K., and J. O'Brien. 2010. "Cartoblography: experiments in using and organizing the spatial context of micro-blogging." *Transactions in GIS* 14(1):5–23. doi:[10.1111/j.1467-9671.2010.01210.x](https://doi.org/10.1111/j.1467-9671.2010.01210.x).
- Forrest, D., and H. W. Castner. 1985. "The design and perception of point symbols for tourist maps." *The Cartographic Journal* 42:11–19. doi:[10.1179/caj.1985.22.1.11](https://doi.org/10.1179/caj.1985.22.1.11).
- GICHD. 2005. "Cartographic recommendations for humanitarian demining map symbols in the Information Management System for Mine Action (IMSMA)." Geneva: Geneva International Centre for Humanitarian Demining.

- Graham, M. 2010. "Neogeography and the palimpsests of place: Web 2.0 and the construction of a virtual earth." *Tijdschrift voor economische en sociale geografie* 101(4):422–436. doi:[10.1111/j.1467-9663.2009.00563.x](https://doi.org/10.1111/j.1467-9663.2009.00563.x).
- Groop, R. E., and D. Cole. 1978. "Overlapping graduated circles: Magnitude estimation and method of portrayal." *Canadian Cartographer* 15(2):114–122. doi:[10.3138/Q5Q5-N244-8462-NG25](https://doi.org/10.3138/Q5Q5-N244-8462-NG25).
- Harrower, M., and C. A. Brewer. 2003. "ColorBrewer.org: an online tool for selecting colour schemes for maps." *The Cartographic Journal* 40(1):27–37. doi:[10.1179/000870403235002042](https://doi.org/10.1179/000870403235002042).
- Hoaglin, D. C., B. Iglewicz, and J. W. Tukey 1986. "Performance of some resistant rules for outlier labeling." *Journal of American Statistical Association* 81:991–999. doi:[10.1080/01621459.1986.10478363](https://doi.org/10.1080/01621459.1986.10478363).
- Keates, J. S. 1989. *Cartographic Design and Production. 2nd edition*. London: Longman.
- Kellman, P. J. 2001. "Separating processes in object perception." *Journal of Experimental Child Psychology* 78:84–97. doi:[10.1006/jecp.2000.2604](https://doi.org/10.1006/jecp.2000.2604).
- Kellman, P. J. and T. F. Shipley. 1991. "A theory of visual interpolation in object perception." *Cognitive Psychology* 23:141–221. doi:[10.1016/0010-0285\(91\)90009-D](https://doi.org/10.1016/0010-0285(91)90009-D).
- Korpi, J., and P. Ahonen-Rainio. 2010 "Cultural constraints in the design of pictographic symbols." *The Cartographic Journal* 47(4):351–359. doi:[10.1179/000870410X12911337964923](https://doi.org/10.1179/000870410X12911337964923).
- . 2013 "Clutter reduction methods for point symbols in map mashups." *The Cartographic Journal* 50(3): 257–265. doi:[10.1179/1743277413Y.0000000065](https://doi.org/10.1179/1743277413Y.0000000065).
- Kosslyn, S. M., and O. Koenig. 1992. *Wet Mind: The New Cognitive Neuroscience*. Toronto: Maxwell Macmillan.
- Kostelnick, J. C., J. E. Dobson, S. L. Egbert, and M. D. Dunbar. 2008. "Cartographic symbols for humanitarian demining." *The Cartographic Journal* 45(1):18–31. doi:[10.1179/000870408X276585](https://doi.org/10.1179/000870408X276585).
- Kraak, M.-J. 2011. "Is there a need for neo-cartography?" *Cartography and Geographic Information Science* 38(2):73–78. doi:[10.1559/1523040638273](https://doi.org/10.1559/1523040638273).
- van Lier, R. J. 1999. "Investigating global effects in visual occlusion: from a partly occluded square to the back of a tree-trunk." *Acta Psychologica* 102:203–220. doi:[10.1016/S0001-6918\(98\)00055-9](https://doi.org/10.1016/S0001-6918(98)00055-9).
- van Lier, R. J., P. A. van der Helm, and E. L. J. Leeuwenberg. 1995. "Competing global and local completions in visual occlusion." *Journal of Experimental Psychology: Human Perception and Performance* 21:571–583. doi:[10.1037/0096-1523.21.3.571](https://doi.org/10.1037/0096-1523.21.3.571).
- Lloyd, R. 1997. "Visual search processes used in map reading." *Cartographica* 34(1):11–32. doi:[10.3138/F342-1217-827R-0777](https://doi.org/10.3138/F342-1217-827R-0777).
- MacEachren, A. M. 1995. *How Maps Work: Representation, Visualization and Design*. New York: Guilford Press.
- McMaster, R. B., and K. S. Shea. 1992. *Generalization in Digital Cartography*. Washington, DC: Association of American Geographers.
- Quinlan, P. T., and R. N. Wilton. 1998. "Grouping by proximity or similarity? Competition between Gestalt principles in vision." *Perception* 27:417–430. doi:[10.1068/p270417](https://doi.org/10.1068/p270417).
- Rensink, R. A., and J. T. Enns. 1995. "Preemption effects in visual search: evidence for low-level grouping." *Psychological Review* 102(1):101–130. doi:[10.1037/0033-295X.102.1.101](https://doi.org/10.1037/0033-295X.102.1.101).
- Rosenholz, R., Y. Li, and L. Nakano. 2007. "Measuring visual clutter." *Journal of Vision* 7(2):1–22.
- Rubin, N. 2001. "The role of junctions in surface completion and contour matching." *Perception* 30:339–366. doi:[10.1068/p3173](https://doi.org/10.1068/p3173).
- Sekuler, A. B., S. E. Palmer, and C. Flynn. 1994. "Local and global processes in visual completion." *Psychological Science* 5:260–267. doi:[10.1111/j.1467-9280.1994.tb00623.x](https://doi.org/10.1111/j.1467-9280.1994.tb00623.x).

- Slocum, T. A., R. B. McMaster, F. C. Kessler, and H. H. Howard. 2005. *Thematic Cartography and Geographic Visualization, 2nd edition*. Upper Saddle River: Pearson Prentice Hall.
- Sloman, S. 2009. *Causal Models: How People Think About the World and Its Alternatives*. New York: Oxford University Press.
- Swienty, O., M. Jahnke, H. Kumke, and S. Reppermund. 2008. "Effective visual scanning of geographic information." *Visual Information Systems, Web Based Visual Information Search and Management*, edited by M. Sebillio, G. Vitiello, and G. Schaefer, 19–30. Berlin-Heidelberg: Springer. doi:[10.1007/978-3-540-85891-1_5](https://doi.org/10.1007/978-3-540-85891-1_5).
- Torralba, A. 2009. "How many pixels make an image?" *Visual Neuroscience* 26:123–131. doi:[10.1017/S0952523808080930](https://doi.org/10.1017/S0952523808080930).
- Treisman, A., G. Gelade. 1980. "A feature-integration theory of attention." *Cognitive Psychology* 12(1):97–136. doi:[10.1016/0010-0285\(80\)90005-5](https://doi.org/10.1016/0010-0285(80)90005-5).
- Tse, P. U. 1999. "Complete mergeability and amodal completion." *Acta Psychologica* 102(2–3):165–201. doi:[10.1016/S0001-6918\(99\)00027-X](https://doi.org/10.1016/S0001-6918(99)00027-X).
- Tukey, J. W. 1977. *Exploratory Data Analysis*. Reading, MA: Addison-Wesley
- Valenza, E., I. Leo, L. Gava, F. Simion. 2006. "Perceptual completion in newborn human infants." *Child Development* 77(6):1810–1821. doi:[10.1111/j.1467-8624.2006.00975.x](https://doi.org/10.1111/j.1467-8624.2006.00975.x).
- Ware, C. 2000. *Information Visualization: Perception for Design*. San Francisco: Morgan Kaufmann.
- Wertheimer, M. 1958. "Principles of perceptual organization" *Readings in Perception*, edited by D.C. Beardslee and M. Wertheimer Princeton. NJ: D. Van Nostrand Co.
- de Wit, T. C. J., K. R. Mol, and R. van Lier, 2005. "Investigating metrical and structural aspects of visual completion: priming versus searching." *Visual Cognition* 12(3):409–428. doi:[10.1080/13506280444000373](https://doi.org/10.1080/13506280444000373).
- Wolfe, J. M. 2007. "Guided Search 4.0: Current Progress with a model of visual search." *Integrated Models of Cognitive Systems*, edited by W. Grey, 99–119. New York: Oxford.
- Wolfe, J. M., and T. S. Horowitz. 2004. "What attributes guide the deployment of visual attention and how do they do it?" *Nature Reviews Neuroscience* 5:1–7. doi:[10.1038/nrn1411](https://doi.org/10.1038/nrn1411).

Maps as the Art of Necessity: The Lilly Library War of 1812 Map Collection

Lori Dekydspotter
Lilly Library, Indiana University
lodekydt@indiana.edu

Heiko Mühr
Wells Library, Indiana University
hmuehr@indiana.edu

IN HIS EXAMINATION of the War of 1812, J. C. A. Staggs notes that “Many of the problems that we encounter today in understanding the War of 1812 arise from the fact that the political geography of North America has changed greatly since 1815” (2012, 19). Places central to the conflict have either been renamed or have disappeared completely from later maps, making contemporary maps produced during or shortly after the conflict invaluable. This essay highlights a collection of such maps that are held at Indiana University’s rare books and special collections library, The Lilly Library.

The War of 1812 had a significant impact on a nation eagerly expanding its borders. Many areas of the United States were affected by the conflict, including the Indiana Territory. Indiana would go on to attain statehood in 1816, but not before key events unfolded and battles took place on its soil. Territorial governor William Henry Harrison negotiated the Treaty of Fort Wayne in 1809, forcing Native American delegates to accept substantial land cessions to the United States government. Angered by the sale of yet more land, Shawnee leader Tecumseh and 400 armed

followers traveled in the summer of 1810 to meet with Harrison in the territorial capital of Vincennes. Tecumseh contended that the Fort Wayne Treaty was illegitimate and demanded that Harrison nullify it. After a heated and unsuccessful exchange, Tecumseh threatened to ally with the British if his demands were not met (Langguth 2007, 166).

In November 1811, after tensions and violence had increased, Harrison successfully led 1000 American troops in a battle at Prophetstown, located near the confluence of the Tippecanoe and Wabash Rivers. The defeat at the Battle of Tippecanoe was a setback for the Native American forces that opposed the continuing cessions of their territory. Undeterred, Tecumseh’s Confederacy went on to ally with British forces in Canada. Consequently, the borderlands between Canada and the United States became the primary theater of conflict in the opening stages of the War of 1812. Fighting centered on Lakes Ontario and Erie, both Upper and Lower Canada, and the Old Northwest, consisting of Ohio and the Michigan and Indiana Territories. Later in the war, significant fighting took place in the Chesapeake Bay area and near New Orleans.

THE WAR OF 1812 BICENTENNIAL PROJECT

IN PREPARATION FOR the War of 1812 bicentennial, Indiana University Library embarked on a collaborative, year-long project involving the Lilly Library, the Indiana University Digital Library Program, and the Wells Library Technical Services Department. A website, *The War of 1812*, was created (Figure 1), which gathers a wealth of information on the war through a combination of essays and digitized primary source materials. Hundreds of manuscripts, books, maps, and prints relating to the War of 1812 are available online in digitized form. All of the

physical materials are located in the Lilly Library, Indiana University Bloomington.

The materials in the War of 1812 collection were acquired by Indiana University principally from 1942 to 1969, from a variety of sources and through the cooperative efforts of historians, librarians, and booksellers. The maps in particular came to the Lilly Library through collector, historian, and Indiana native, Richard Elwell Banta (1904–1977). Banta was involved in the book trade for much of his life, selling antiquarian and trade books to academic libraries



© by the author(s). This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

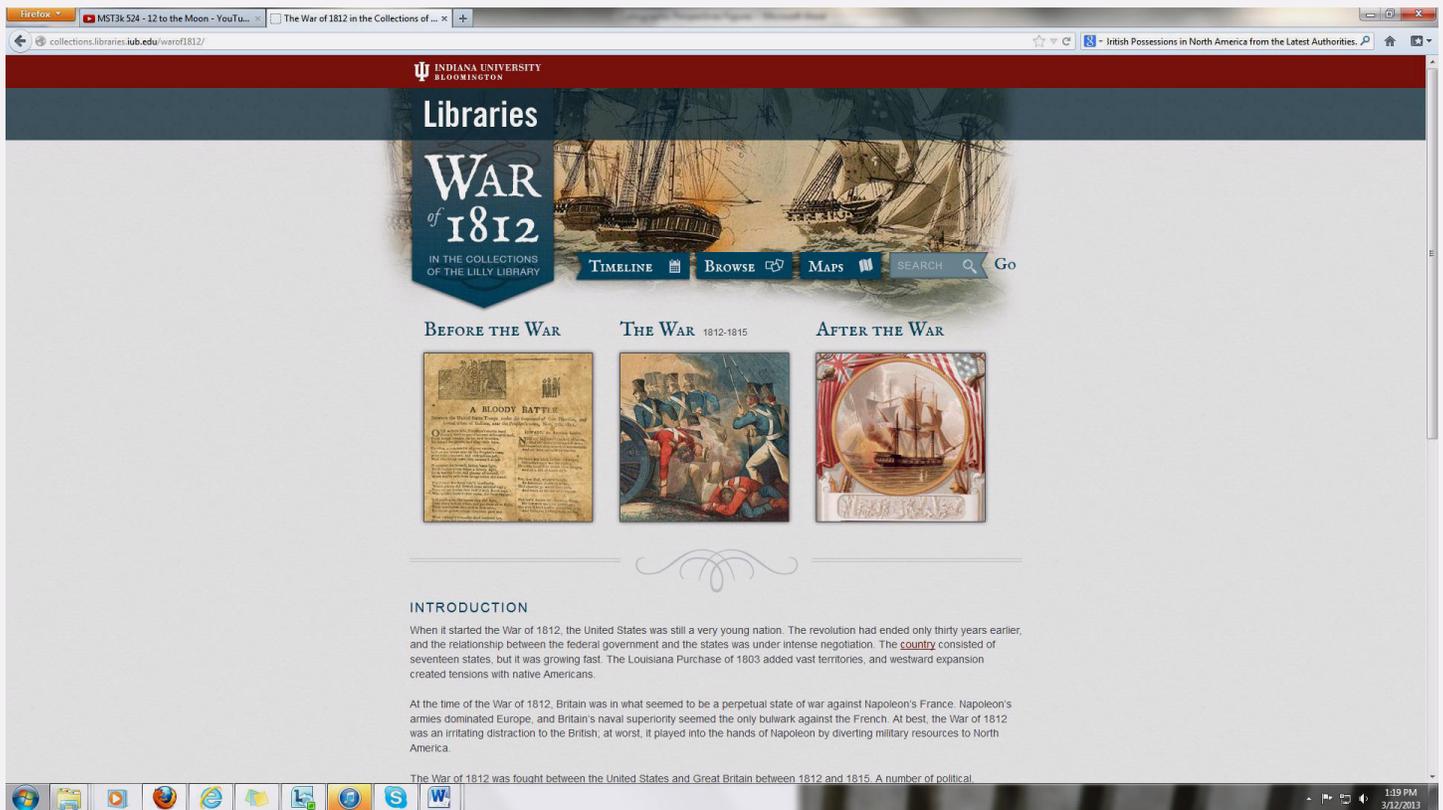


Figure 1: The War of 1812 website.

throughout the country. Although not formally trained in the subject, Banta became interested in the history of the Old Northwest, reading widely in the field and contributing to journals, periodicals, and encyclopedias (Thompson 1974, 28).

This portion of the collection consists of manuscripts and prints of coastal and theater-of-war maps. Some depict military activities during the war. The Lilly Library cataloged most of the printed items and manuscript collections; however, the Wells Library Technical Services Department contributed greatly by creating the War of 1812 finding

aid for the almost 5,000 individual items included in the collection. The 23 maps that were selected from the collection required specialized description and cataloging. In addition to the maps, correspondence, log books, legal documents, diaries, speeches, copybooks, orderly books, and receipts also received enhanced descriptions and were likewise digitized. Indiana University Libraries Digital Projects & Services (www.dlib.indiana.edu) digitized most of the collection items in the Digital Media and Imaging Center located in Wells Library; however, the most fragile and rare materials were digitized by staff located in the Lilly Library.

A NATIONAL GEOGRAPHY EMERGES

AS RICHARD W. STEPHENSON EXPLAINS, “War, like necessity, has been called the mother of invention. The same might be said of cartography, for with every war there is a great rush to produce maps to aid in understanding the nature of the land over which armies will move and fight, to plan engagements and the deployment of troops, and to record victories for posterity to study and admire” (1989, 1). The maps in the Lilly Library’s War of 1812 collection

represent the conflicts that took place on both land and water.

In an early survey of the history of American mapmaking, Erwin Raisz wrote that “the map of America has been gradually built up through the centuries. The early charts of the Spanish, Italian, Dutch, French, and English navigators, and the maps of the explorers and army officers, however important, are American in their subject only. Their

maps were engraved and published in Europe” (1937, 373). In contrast, state maps were among the earliest examples of American mapmaking (Figure 2). Based on original surveys, these maps were compiled, drafted, engraved, printed, and published by Americans.

American publishers issued state maps and accompanying gazetteers, eventually binding state maps together in an atlas format. These maps and atlases graphically represented the territory of the United States. Publishers also Americanized maps and geography texts by introducing American prime meridians. An increasingly opinionated and connected geographic discourse emerged, influenced by geopolitical events. Raisz identifies this era as “The

Emancipation of American Cartography (1780–1820)” (1937, 378).

Martin Brückner (2006) has argued that the newly-produced maps afforded the young nation an understanding of itself as a geographical whole. Brückner documents a vibrant eighteenth century culture of geography consisting of plat maps and surveying manuals, decorative wall maps, gazetteers, geography primers, and atlases. He demonstrates that the rise in popularity of maps and geography texts ushered in a new geographic literacy among ordinary Americans. John Rennie Short (2001) likewise viewed the publishing of these maps, atlases and geographical texts as the emergence of a national geography. He identified three



Figure 2: Carey's Louisiana state map, 1818.



Figure 3: Carey's British Possessions in North America from the Latest Authorities, 1814.

key players in this process: Jedidiah Morse, the author of the first comprehensive geography of North America, and Philadelphia map publishers Mathew Carey and John Melish, singling out Melish as “a full-time geographer, mapmaker and map publisher” whose “adult life was devoted to what he termed the topographic representation of the new republic” (127).

Matthew Carey (1760–1839), a Dublin-born immigrant, established a print shop and publishing house in Philadelphia. In 1795, he published *Carey's American Atlas*, the earliest atlas of the United States. Carey organized an elaborate cottage system of craftsmen for engraving,

printing, and coloring his maps, utilizing the best independent talent available. Some scholars of cartographic history have argued that the publication of *Carey's American Atlas* ushered in a “golden age” of American cartography:

Nearly all of the atlases of this era were printed with copperplates, a method that produced maps far more durable than those of later years. Also known as the intaglio process, copperplate engraving allowed fine, delicate lines that were easily maintained and updated; subtle and variable area

tones; and small, neat lettering. (Schulten 2001, 21)

In his seminal study, *American Maps and Map Makers*, Walter W. Ristow underlines Carey's role as "a pioneer in atlas publishing," but points out that Carey was not primarily concerned with publishing cartographic works. In contrast, "John Melish, a native of Scotland, was the first American publisher to concentrate his efforts wholly on producing maps, atlases, and geographical publications" (Ristow 1985, 21).

John Melish (1771–1822) first visited the United States while working for a Glasgow cotton merchant. Melish eventually relocated to Philadelphia and based his first major publication, a fact-filled two-volume work, *Travels in the United States of America in the Years 1806 & 1807, and 1809, 1810 & 1811* (1812), on the copious notes he had taken during his travels as a cotton merchant and on a trip to the Old Northwest. Eight maps illustrated the

volumes and Melish subsequently turned to map publishing. By 1814, he identified himself as a "Geographer and Map Seller" and had published several independent maps, geographies and gazetteers.

While engraving a map of the United States, the frontispiece for volume one of *Travels*, Henry Schenck Tanner "suggested the propriety of drawing a general map of the seat of war, and proffered the use of a very ample set of maps in his possession." The *Map of the Seat of War in North America* (Figure 4) was consequently also engraved by Tanner (1786–1858). Melish reports the map "sold so rapidly that the first plate was soon worn out, and a new one has since been brought forward, enlarged and much improved" (Melish 1813, 3).

That same year, Melish published several new maps which showed other areas that would likely be affected by the spread of war. *The Military and Topographical Atlas of the United States*, released in late 1813, consisted of eight such

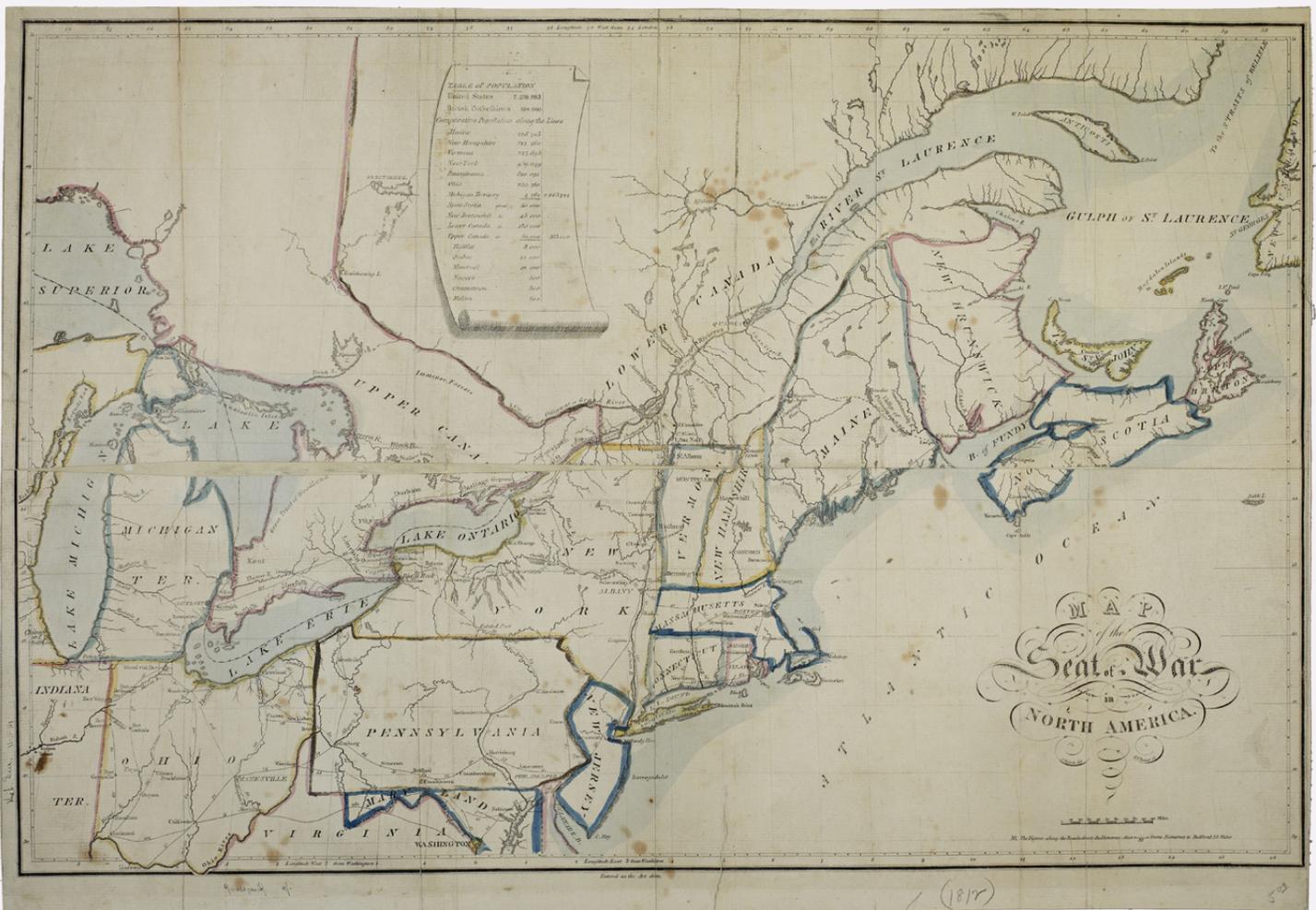


Figure 4: Tanner's The Map of the Seat of War in North America, 1813.

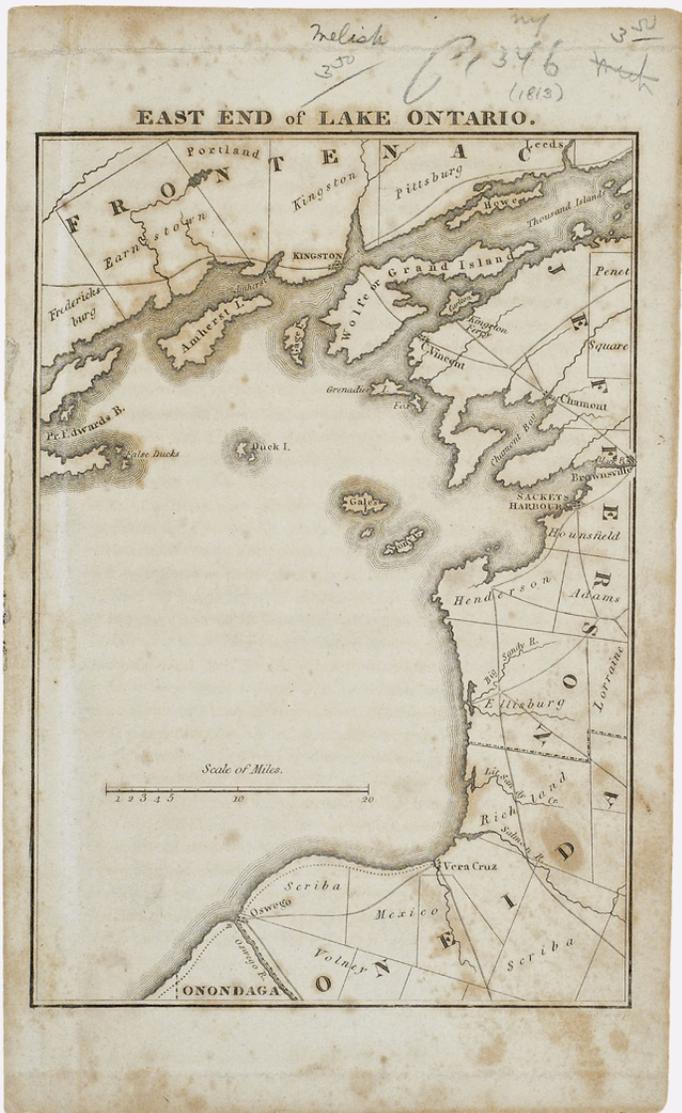


Figure 5: Melish's East End of Lake Ontario, 1813.

maps engraved by Tanner. It included innovative smaller maps which offered a more detailed view of areas which had already seen extensive fighting, including the East End of Lake Ontario (Ristow 1985, 181) (Figure 5).

An expanded edition of *The Military and Topographical Atlas of the United States*, published after the Peace of Ghent in 1815, featured twelve maps, including a map of the New Orleans area. Specifically associated with the War of 1812, these publishing projects distinguished Melish as one of Philadelphia's premier cartographic publishers. Seeing the profit to be made, other commercial publishers also entered the fray. Some of these Philadelphia competitors were at the same time collaborating with Melish on other publishing projects.



Figure 6: Amos Lay, A New Correct Map of the Seat of War in Lower Canada, 1813.

Land surveyor Amos Lay (1765–1851) collaborated with engraver Henry Schenck Tanner to publish *A New Correct Map of the Seat of War in Lower Canada* (1814). The map (Figure 6) locates several of the military actions fought during the 1813 Saint Lawrence campaign, including the site of the Battle of Chrysler's Farm (November 11, 1813).

A Correct Map of the Seat of War (1812), drawn by Samuel Lewis (1753?–1822) and engraved by Samuel Harrison (Figure 7), shows important detail in the Old Northwest, such as the "Road cut by Gen. Hull" at the beginning of the Detroit campaign of 1812 and also the "Indian Boundary Line," agreed upon at the Treaty of Greenville (1795).

British cartographers also seized upon similar opportunities and managed to reach even larger audiences: John Luffman (1756–1846) published *A Map of the American Lakes and Adjoining Country: The Present Seat of War between Great Britain & the United States* (1813), claiming the map was "done in part, from a sketch of the late Major General Sr. Isaac Brock." Having forged an alliance with Tecumseh and forced the surrender of an American army at Detroit, Brock died in October 1812 defending the Niagara frontier at the battle of Queenston Heights. Luffman also took care to locate the principal tribes of Tecumseh's Confederacy (Figure 8).

In contrast, *The Sketch of the March of the British Army under Gen'l Ross from the 19th to the 29th August 1814* published from a sketch by D. Evans, Lt 3d Dr's., by James Wild, the Elder (1790–1836) is a detailed campaign and battle map (Figure 9). It shows the positions of British and American troops, relevant roads and waterways, and an



Figure 7: A Correct Map of the Seat of War, drawn by Samuel Lewis, 1812.

inset, “Sketch of the engagement on the 24th of August 1814 between the British and American forces,” the Battle of Bladensburg, Maryland, also known as the Bladensburg Races. The American militia fled through the streets of Washington before the British entered the city unopposed

and set fire to many of the government buildings in what became known as the Burning of Washington.

John Melish’s Philadelphia publishing house went on to produce many other outstanding maps. Melish’s most widely known work, the *Map of the United States with the Contiguous British and Spanish Possessions* (1816), an iconic large format representation of the United States as a transcontinental power, was published right after the War of 1812. Its ambitious conceptualization shows the United States boldly stretching from coast to coast, foreshadowing a period of sustained national growth and expansion, fueled by visions of Manifest Destiny (Ristow 1962).

For more information about the War of 1812 materials and the Lilly Library’s other collections, please visit the Library website, www.indiana.edu/~liblilly.

The authors would like to thank Erika Dowell, Interim Head of Technical Services and Co-Director of the War of 1812 Project, Lilly Library, Indiana University; and Elizabeth Johnson, Emeritus Librarian, Lilly Library, Indiana University.



Figure 8: Luffman’s A Map of the American Lakes and Adjoining Country, 1813.

REFERENCES

Banta, R. E. Banta mss. collection. Lilly Library, Indiana University, Bloomington, IN.

Brückner, M. 2006. *The Geographic Revolution in Early America: Maps, Literacy, and National Identity*. Chapel Hill: University of North Carolina Press.

Langguth, A. J. 2007. *Union 1812: The Americans Who Fought the Second War of Independence*. New York: Simon & Schuster.

Lilly Library, Indiana University Bloomington. 2013. *War of 1812*. Accessed March 10. <http://collections.libraries.iub.edu/warof1812>.

Melish, J. 1813. *A Military and Topographical Atlas of the United States*. Philadelphia: Printed by G. Palmer.

Raisz, E. 1937. "Outline of the History of American Cartography." *Isis*. 26:373–391. doi:[10.1086/347187](https://doi.org/10.1086/347187).

Ristow, W. W. 1962. "John Melish and His Map of the United States." *Quarterly Journal of Current Acquisitions* 19:159–178.

———. 1985. *American Maps and Map Makers: Commercial Cartography in the Nineteenth Century*. Detroit: Wayne State University Press.

Schulten, S. 2001. *The Geographical Imagination in America, 1880–1950*. Chicago: University of Chicago Press.

Short, J. R. 2001. *Representing the Republic: Mapping the United States, 1600–1900*. London: Reaktion Books.

Skaggs, J. C. A. 2012. *War of 1812: Conflict for a Continent*. Cambridge: Cambridge Essential Histories.

Stephenson, R. W. 1989. *Civil War Maps: An Annotated List of Maps and Atlases in the Library of Congress, 2nd edition*. Washington: Library of Congress. <http://www.loc.gov/collections/civil-war-maps/special-presentation/history-of-mapping-the-civil-war>.

FEATURED LILLY LIBRARY MAPS

British Possessions in North America from the Latest Authorities. M. Carey, W. Robinson, engraver. Philadelphia: M. Carey, 1814.

Correct Map of the Seat of War. S. Lewis. Philadelphia: J. Conrad, 1812.

East End of Lake Ontario. J. Melish, H. S. Tanner, engraver. Philadelphia: G. Palmer for J. Melish, 1813.

Louisiana. M. Carey. Philadelphia: M. Carey & Son, 1818.

A Map of the American Lakes and Adjoining Country: The Present Seat of War Between Great Britain & the United States. J. Luffman. London: J. Luffman, 1813.

Map of the Seat of War in North America. J. Melish, H. S. Tanner, engraver. Philadelphia: J. Melish, 1813.

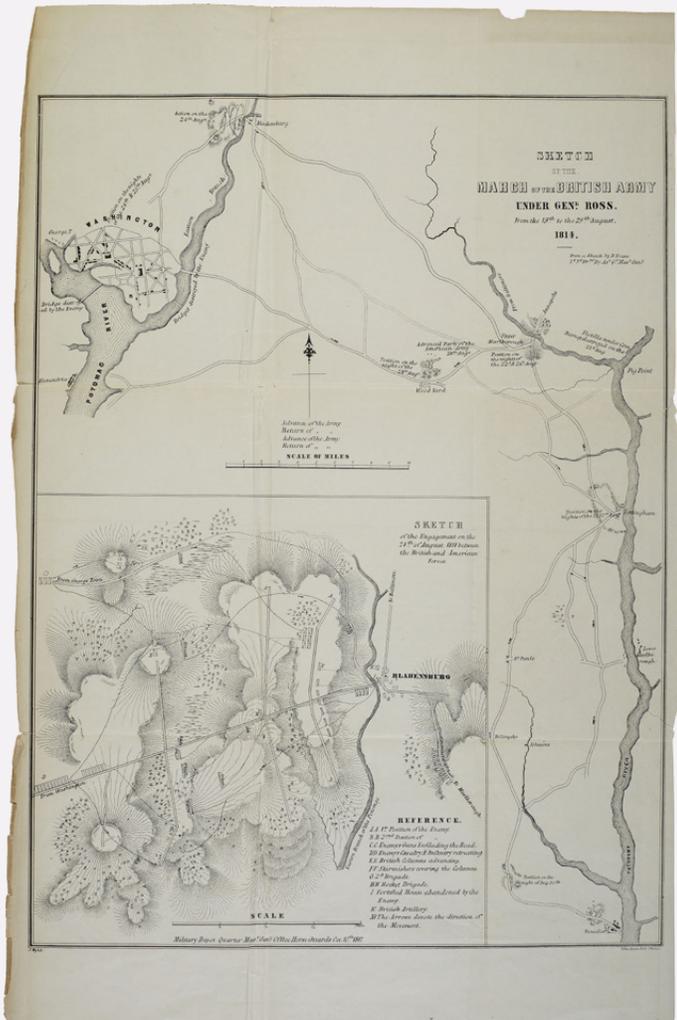


Figure 9: Sketch of the March of the British Army under Gen'l Ross, 1814.

*A New Correct Map of the Seat of War in Lower Canada:
Protracted from Hollands Large Map Compiled from
Actual Survey Made by Order of the Provincial Govt.
L. Amos and J. Webster, H.S. Tanner, engraver.
Philadelphia, 1813.*

*Sketch of the March of the British Army Under Gen'l Ross
from the 19th to the 29th August 1814. D. Evans, T.
Sinclair, lith. Philadelphia: T. Sinclair, 1814.*

Time Series Proportional Symbol Maps with Leaflet and jQuery

Richard G. Donohue
University of Kentucky
rgdonohue@uky.edu

Carl M. Sack
University of Wisconsin–Madison
cmsack@wisc.edu

Robert E. Roth
University of Wisconsin–Madison
reroth@wisc.edu

ASSUMED SKILLS AND LEARNING OUTCOMES

THE FOLLOWING TUTORIAL describes how to make a time series proportional symbol map using the Leaflet (leafletjs.com) and jQuery (jquery.com) code libraries. The tutorial is based on a laboratory assignment created in Spring of 2013 for an advanced class on Interactive Cartography and Geovisualization at the University of Wisconsin–Madison (www.geography.wisc.edu/courses/geog575). This is the first of two *On the Horizon* tutorials on the topic of web mapping, with the next tutorial covering multivariate choropleth mapping using the D3 library. Commented source code for the tutorial is available through a Creative Commons license at geography.wisc.edu/cartography/tutorials.

The tutorial assumes a basic understanding of the open web platform, particularly the HTML, CSS, and JavaScript standards. It also assumes that you are familiar with the manipulation of JavaScript objects as well as jQuery-style DOM element selection. Tutorials and reference documentation for HTML, CSS, and JavaScript are available at

such resources as developer.mozilla.org, www.lynda.com, www.codecademy.com, and www.w3schools.com. Further, it is assumed that you are familiar with in-browser development tools such as Chrome Developer Tools (developers.google.com/chrome-developer-tools) or Firebug (getfirebug.com). Finally, the tutorial assumes that you have access to a web server, either running remotely or as a local host; MAMP for Mac (www.mamp.info/en) and WAMP for Windows (www.wampserver.com/en) are useful for this.

After completing the tutorial, you will be able to:

- Work with the GeoJSON data format
- Use the Leaflet library to publish a time series proportional symbol map to the web
- Create interactivity using mouseover popup windows and range sliders

GETTING STARTED WITH LEAFLET

LEAFLET IS ONE OF MANY code libraries now available for publishing slippy maps to the web. Leaflet is a JavaScript library pioneered and maintained by Vladimir Agafonkin (agafonkin.com/en), and quickly is growing in popularity within the web development community because it is both lightweight (only 33kb of code at the time of this writing) and open source (meaning you can both view how it functions and extend it to fit your needs). Maps produced using Leaflet can load a variety of basemap tile services and can draw vector features atop these tiles using the SVG (Scalable Vector Graphics) standard. Leaflet is also

bundled with Mapbox.js (www.mapbox.com/mapbox.js), allowing for simple loading and manipulation of custom tilesets. Because of the small file size and support of touch-based interactions, Leaflet is considered among the best web mapping libraries when designing for mobile devices.

The Leaflet library is an open-source project on GitHub (github.com/Leaflet) and can be extended through numerous open plugins (leafletjs.com/plugins.html). The goal of this tutorial is to provide you with a broad introduction to using Leaflet for Web Cartography. The following tutorial



© by the author(s). This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

extends the reference and tutorials available at leafletjs.com/examples.html. There also are supportive communities for the Leaflet library on Twitter (<https://twitter.com/search?q=%23leaflet>) and Stack Overflow (stackoverflow.com/search?q=leaflet). Refer to these materials for additional background and guidance as you complete the tutorial.

com/search?q=leaflet). Refer to these materials for additional background and guidance as you complete the tutorial.

1. FINDING AND FORMATTING TIME SERIES INFORMATION

THE FIRST STEP is the assembly of appropriate *time series* information (i.e., geographic information that changes over time) to portray on your proportional symbol map. Because proportional symbol maps leverage the visual variable size, you only should map ordinal, or, preferably, numerical data using this thematic map type (i.e., do not collect categorical information).

Use your preferred spreadsheet (e.g., as a *.csv* file) or GIS (e.g., as a *.shp* file) software to prepare your dataset. Format the dataset with the unique map features (e.g., cities, regions) included as rows and the unique timestamps (generically describing either a single moment in time or a time interval) included as columns. Be sure to use logical header names (such as 2005, 2006, etc.), as these terms serve as attribute keys for referencing the time series information using JavaScript and will be used to create a temporal legend in the map itself. Because Leaflet natively understands the geographic coordinate system, you need to include a pair of columns for the latitude and longitude of the proportional symbol anchor (e.g., the city center, the centroid of the region). For this tutorial save the latitude value as *lat* the longitude value as *lon*. Finally, include an additional pair of columns at the start of your file for a unique *id* number and name field. **Figure 1** provides an example time series dataset for fifteen major cities in the United States.

Next, convert your dataset into the GeoJSON format. *JSON* stands for JavaScript Object Notation and has become a standard format for information loaded into and interpreted by a browser. *GeoJSON* is one geographic variant of JSON that structures each map feature as an array

	A	B	C	D	E	F	G	H	I	J	K
1	id	name	latitude	longitude	2005	2006	2007	2008	2009	2010	2011
2	1	Atlanta	33.7489	-84.3881	85	38	75	30	9	15	38
3	2	Chicago	41.85	-87.65	28	29	38	26	15	12	10
4	3	Dallas	32.7828	-96.8039	18	59	22	60	82	42	18
5	4	Denver	39.7392	-104.9842	35	45	31	26	14	9	15
6	5	Houston	29.7631	-95.3631	12	31	15	22	28	38	31
7	6	Kansas City	39.0997	-94.5783	25	50	25	25	25	25	100
8	7	Los Angeles	34.0522	-118.2428	88	46	56	15	12	25	46
9	8	Miami	25.7738	-80.1924	52	51	46	68	75	85	96
10	9	Minneapolis	44.98	-93.2636	7	12	18	11	9	9	4
11	10	New York	40.7142	-74.0064	23	18	16	24	26	28	30
12	11	Philadelphia	39.9522	-75.1642	32	28	29	25	22	15	8
13	12	Phoenix	33.4539	-112.0746	8	15	22	25	29	28	32
14	13	San Francisco	37.775	-122.4183	82	74	72	10	85	88	74
15	14	Seattle	47.6097	-122.3331	9	16	14	23	45	66	85
16	15	Washington	38.89	-77.03	33	45	68	96	102	82	74
17											

Figure 1: An Example Time Series Dataset.

of *nodes* (lat/long coordinate pairs) defining the complete outer boundary of the polygon.

There are multiple ways to convert your dataset to the GeoJSON format, including:

- GIS applications such as ArcGIS (www.esri.com/software/arcgis) or QGIS (www.qgis.org);
- Open utilities such as GDAL/OGR (www.gdal.org/ogr2ogr.html);
- Free web services, such as MapShaper (www.mapshaper.org), ShpEscape (www.shpescape.com), ToGeoJSON (togejson.com), and GeoJSON.io.

Leaflet also includes several methods to load formats other than GeoJSON, although these are not discussed in this tutorial. An example GeoJSON file (extension *.json*) for the **Figure 1** time series dataset is included in the online code repository.

2. PREPARING YOUR DIRECTORY STRUCTURE AND BOILERPLATE

WITH YOUR TIME SERIES information processed, it is now time to start building your map! Create a directory that includes folders named “data,” “css,” “img,” and “js”. Because you will be using AJAX requests, we strongly

recommend that you set up a development server on your machine and place this directory on the server, accessing it as a localhost. Using your preferred text editor, create three new files named *index.html* (root level), *style.css* (css folder),

```

1<!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <title>Leaflet Prop Symbol Map</title>
6
7      <!--[if IE]>
8          <script src="html5shiv.googlecode.com/svn/trunk/html5.js">
9              </script>
10     <![endif]-->
11
12     <!--stylesheet-->
13     <link rel="stylesheet" href="css/leaflet.css">
14     <link rel="stylesheet" href="css/style.css">
15 </head>
16 <body>
17     <!--scripts-->
18     <script src="js/jquery.js"></script>
19     <script src="js/leaflet.js"></script>
20     <script src="js/main.js"></script>
21 </body>
22 </html>

```

Example 1: Basic HTML5 Boilerplate, with References to Styles/Scripts (in: index.html).

and *main.js* (*js* folder). Copy your newly created GeoJSON file into the *data* folder.

Next, add the boilerplate text provided in **Example 1** into the *index.html* file. The boilerplate comprises the minimum markup of a valid HTML 5 document, with one conditional tag to handle older versions of Internet Explorer (**EX1: 7–10**¹). The boilerplate also includes references to the stylesheets (**EX1: 12–14**) and scripts (**EX1: 17–20**) that you will be using in your time series proportional symbol map. Before moving on, change the content of the `<title>` element (**EX1: 5**) to something logical for your map.

After configuring your directory, acquire the most recent, stable version of the Leaflet source code from leafletjs.com/download.html. Uncompress the downloaded *.zip* file and place the *leaflet.css* file into the *css* folder, and the Leaflet *images* folder, *leaflet.js*, and *leaflet-src.js* files into the *js* folder. While the html boilerplate links to minified *leaflet.js* file (**EX1: 19**), it is recommended that you reference

the un-minified, human-readable *leaflet-src.js* file when interpreting Leaflet functionality.

In addition to the Leaflet source code, you also need to acquire the source code for the *jQuery* library. jQuery (jquery.com) is a JavaScript plug-in that simplifies accessing and manipulating DOM elements for both representation and interaction. Additionally, jQuery handles many of the browser compatibility issues that otherwise require specific JavaScript solutions. If you are unfamiliar with jQuery, review the jQuery tutorial available at learn.jquery.com/about-jquery/how-jquery-works/. Download the jQuery source code from jquery.com/download/, uncompress the downloaded *.zip* file, and place the *jquery.js* file in your *js* folder.

Before moving onto the next step, check to see if your file structure and webpage files are properly configured. The primary method for debugging scripts is by printing a message to the *error console* using the `console.log()` method in JavaScript. To demonstrate its utility, and confirm that your webpage is properly configured, add a script to print to the console in the *main.js* file (**Example 2**).

1. This notation is used in the following tutorial for brevity; for example, "Example 1: Lines 7–10" will be displayed as "EX1: 7–10."

```
1 console.log("hello world!");
```

Example 2: Debugging Scripts with the Console (in: main.js).

Once added, open *index.html* in Firefox; at this point, it should be a blank webpage (**Figure 2**). Activate Firebug by clicking the Firebug icon; you may need to *Enable All Panels* in the Firebug dropdown option, if not already enabled. Once activated, click the console tab and reload the page.

3. LOADING A BASEMAP USING LEAFLET

YOU ARE NOW READY to load basemap tiles into your webpage using Leaflet. Leaflet allows you to load tiles from a variety of sources. An overview of public tile services is available on the UW–Madison GIS Collective blog: giscollective.org/tutorials/web-mapping/wmsthree/. For Leaflet to use a public tile service, you need to reference the URL using the following syntax:

```
{s}.acetate.geoiq.com/tiles/acetate/{z}/{x}/  
{y}.png
```

Every tile in a slippy map is a separate 256 x 256 pixel image—a *.png* file in the above example syntax. The {s} indicates possible server instances from which the map can draw tiles. For each loaded tile, the {z} indicates its zoom level, the {x} indicates its horizontal coordinate, and {y} indicates its vertical coordinate. Near all public tile services use this z/x/y directory format, which was pioneered by Google. The example syntax above loads the minimalist *Acetate* tile service (developer.geoiq.com/tools/acetate) from GeoIQ (now Esri); a minimalist tile design is recommended when adding thematic content atop the basemap tiles.



Figure 2: Debugging using the Error Console in Firebug.

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <title>Leaflet Prop Symbol Map</title>
6
7      <!--[if IE]>
8          <script src="html5shiv.googlecode.com/svn/trunk/html5.js">
9          </script>
10     <![endif]-->
11
12     <!--stylesheet-->
13     <link rel="stylesheet" href="css/leaflet.css">
14     <link rel="stylesheet" href="css/style.css">
15 </head>
16 <body>
17     <div id="wrapper">
18         <div id="map"></div>
19     </div><!-- end wrapper -->
20
21     <!--scripts-->
22     <script src="js/jquery.js"></script>
23     <script src="js/leaflet.js"></script>
24     <script src="js/main.js"></script>
25 </body>
26 </html>

```

Example 3: Adding a `<div>` Element for the Map (in: `index.html`).

To load a tileset into Leaflet, first add a `<div>` element to the `<body>` of `index.html`, giving it the `id` attribute `map` for referencing by stylesheets and scripts (EX3: 18). While not required, it is good practice to place this `<div>` within a second `<div>` element named `wrapper` containing any additional page elements you add to your design (EX3: 17–19).

Next, edit the `style.css` file to apply style rules for the pair of `<div>` elements, as well as to define the `<body>` element within the `index.html` document (Example 4). For the tutorial example, the wrapper `<div>` is given a width of 960px (EX4: 6), a conventional width in web design for non-mobile devices, and its left and right margin values are set to `auto` in order to center the wrapper `<div>` within the webpage (EX4: 7). It is necessary to set the height attribute of the map `<div>` in order for Leaflet to draw the map within this container. Note that the width of the map `<div>` will automatically fill 100% of its parent container, in this example the wrapper `<div>`. The height is set to

```

1  body {
2  font-family: sans-serif;
3  }
4
5  #wrapper {
6  width: 960px;
7  margin: 15px auto;
8  }
9
10 #map {
11 height: 450px;
12 margin: 15px auto;
13 }

```

Example 4: Styling the `<div>` Element Containing the Map (in: `style.css`).

```

1   $(document).ready(function() {
2
3       var cities;
4       var map = L.map('map', {
5           center: [37.8, -96],
6           zoom: 4,
7           minZoom: 4
8       });
9
10      L.tileLayer(
11          '{s}.acetate.geoiq.com/tiles/acetate/{z}/{x}/{y}.png', {
12              attribution: 'Acetate tileset from GeoIQ'
13          }).addTo(map);
14  });

```

Example 5: Loading a Basemap using Leaflet (in: main.js).

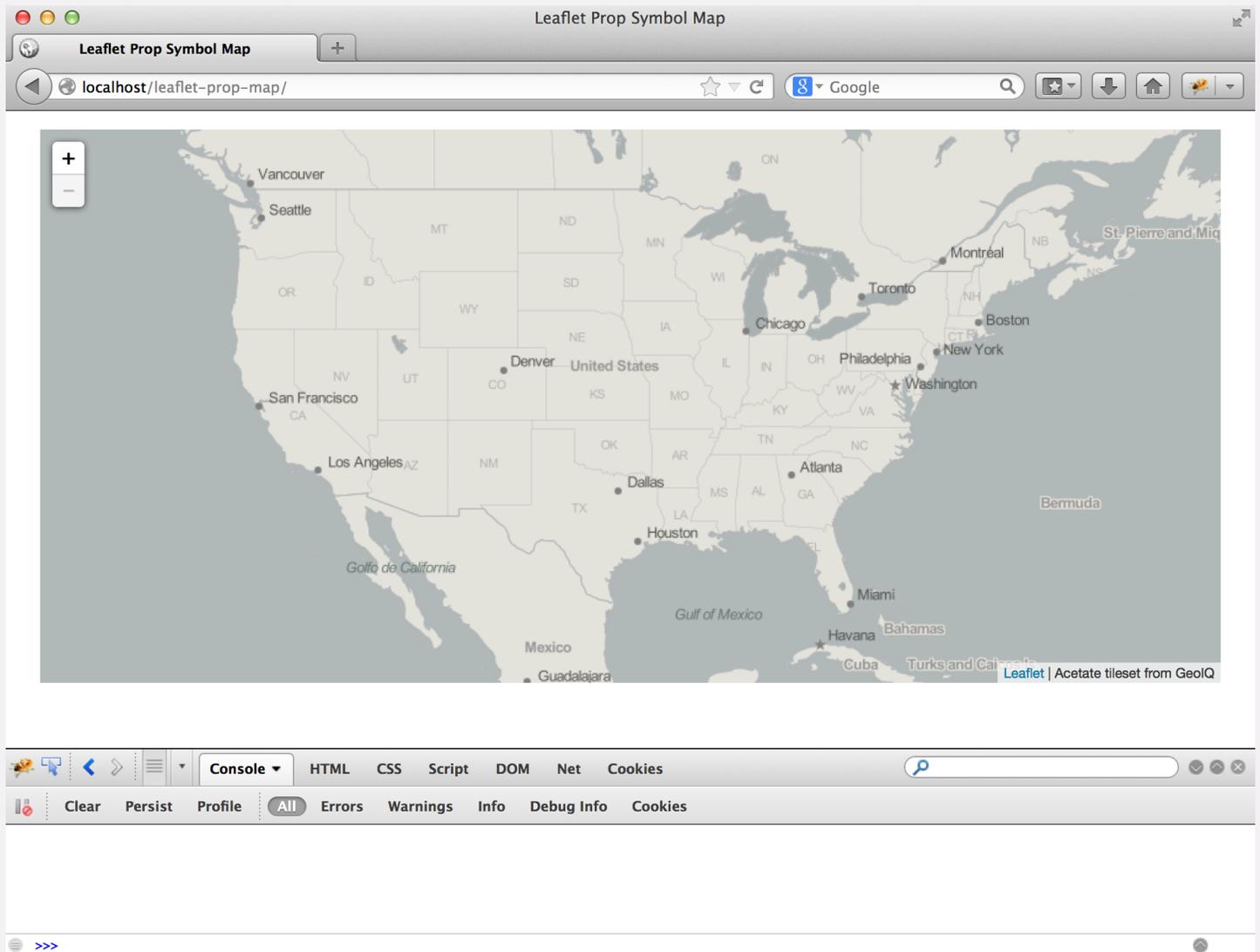


Figure 3: Loading the Acetate tileset.

450px (EX4: 11), and a margin again is used to keep the map away from other page elements as well as to center it within the page (EX4: 12). The values used for these styles are arbitrary, and should be adjusted given the layout of the map within your webpage.

With the page elements in place for the map, you now can add code to the *main.js* file for loading a tileset. The *main.js* file begins with the jQuery method `ready()`, which is used to ensure that the entire page has finished loading before the script begins executing (Example 5). The callback function() within the `ready()` method (EX5: 1) must be closed at the bottom of the *main.js* page (EX5: 14). All subsequent JavaScript code in the tutorial is written within the `ready()` callback function().

Declare two variables within the `ready()` callback function(): (1) `cities` and (2) `map` (EX5: 3–8). The unsigned `cities` variable will reference the proportional

symbols added atop the basemap, as explained later in the tutorial. The map object references the Leaflet map class (`L.Map`) itself, to which all mapped data and controls are added. The map object allows for configuration of basic map parameters, such as the map center (EX5: 5), the zoom scale on loading (EX5: 6), and constraints in zooming interaction (EX5: 7). Review the Leaflet documentation to learn about additional map parameters that can be set using the `L.map` class (leafletjs.com/reference.html#map-usage). The `cities` and `map` variables are declared with a global scope and therefore are accessible within all subsequent method definitions. After the map object is declared and defined, a `tileLayer` of your choosing can be added to the map (EX5: 10–13). Example 5 makes use of the aforementioned Acetate tileset.

Save your changes to the *main.js* file and refresh the *index.html* page in the browser. The map `<div>` element now should be populated with the Acetate tileset, including basic slippy map interactivity (Figure 3).

4. LOADING THE GEOJSON

ONCE YOU HAVE SUCCESSFULLY loaded a tileset into your map `<div>`, the next step is to load the time series dataset you prepared in the GeoJSON file into your webpage; the file is named *cityData.json* in Example 6. Once loaded, this information is used to draw and resize the proportional symbols atop the tile service.

Use the jQuery function `getJSON()` to load the GeoJSON file (EX6: 1); this code block should be placed within the `ready()` callback function(), after adding the `tileLayer`. The `getJSON()` method makes an AJAX request to a specified file (*cityData.json*). After the GeoJSON file is

loaded completely, the data contained in the file is accessible through the `done()` method chained to the `getJSON()` method (EX6: 2–4). In this case, the data object, which is itself a JSON object, is passed as an argument to the callback function(). Use `console.log()` to confirm that the GeoJSON file is loaded correctly (EX6: 3). Finally, an alert is sent if the GeoJSON fails to load (EX6: 5).

Save your changes to the *main.js* file and refresh the *index.html* page in the browser. The contents of the GeoJSON file now should be logged to the console (Figure 4).

```
1 $.getJSON("data/cityData.json")
2   .done(function(data) {
3     console.log(data);
4   })
5   .fail(function() { alert("There has been a problem loading the data.");});
```

Example 6: Loading the GeoJSON (in: *main.js*).

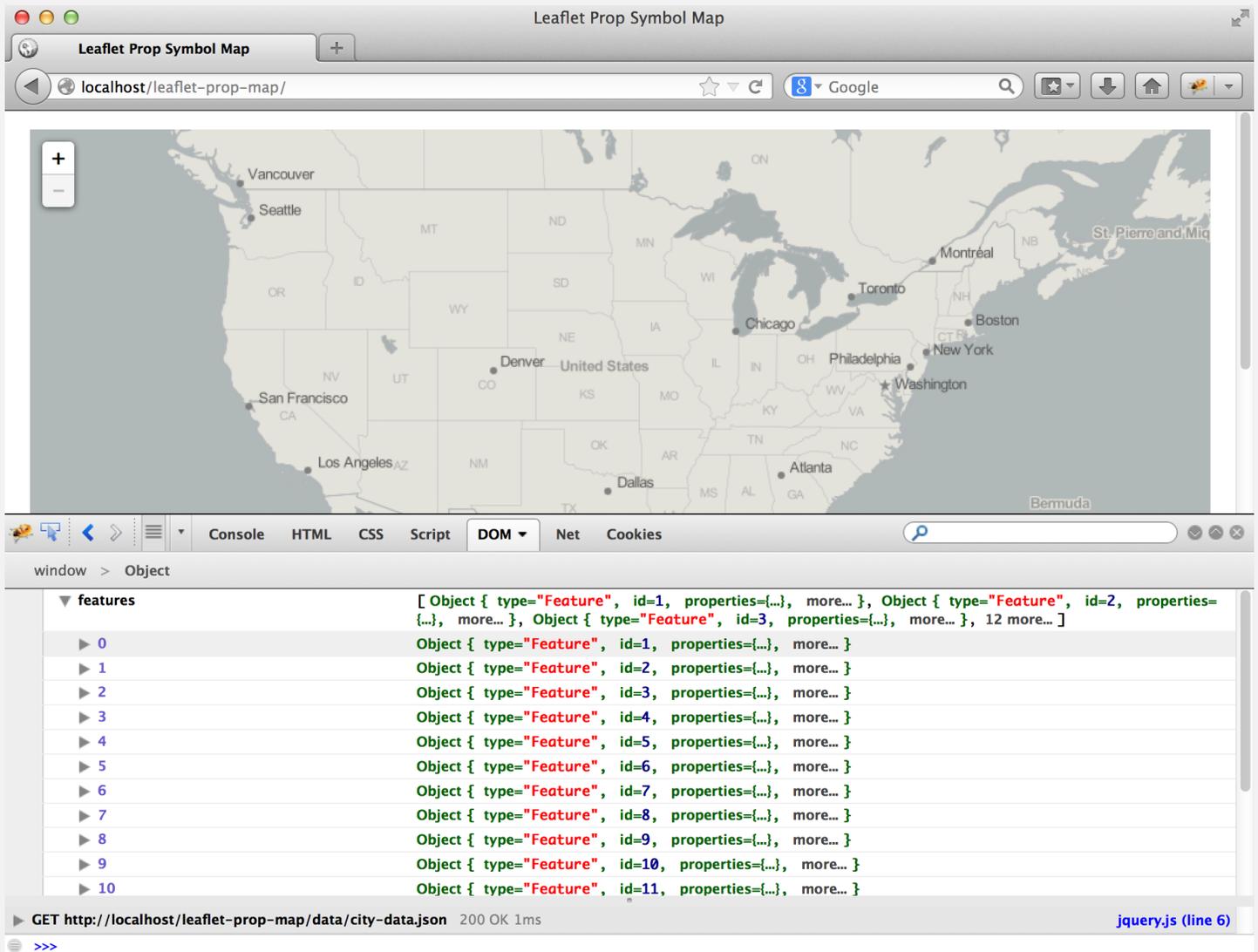


Figure 4: Printing the GeoJSON to the console.

5. PROCESSING THE GEOJSON

ONCE THE GEOJSON is loaded, you can use Leaflet to immediately draw the geographic linework as SVG markers atop the tileset (see Step 6 below). However, to improve the script's efficiency, first process the data to derive values that will be useful later on. In this tutorial, several pieces of information are derived dynamically from the GeoJSON so they need not to be hardcoded into the script. These data include the timestamp names (i.e., the name of each column) for use in a temporal legend and the minimum/maximum values across the time series for use in a map legend.

First remove the `console.log()` call within the `done()` function (EX6: 3) and replace it with a call to a new function named `processData()` (EX7: 3). The function takes

the data variable holding the loaded GeoJSON as a parameter. As described below, the `processData()` function returns the derived information as three key value pairs stored within a JavaScript object. Declare a new variable called `info` to store the returned values for future use in the temporal and map legends.

Next, define the `processData()` function (Example 8). The `processData()` function begins by defining three local variables used to store the derived information (EX8: 2–4): 1) `timestamps` (an array holding the time series headers from the GeoJSON), 2) `min` (a number holding the lowest value across the time series), and 3) `max` (a number holding the highest value across the time series).

```

1 $.getJSON("data/cityData.json")
2   .done(function(data) {
3       var info = processData(data);
4   })
5   .fail(function() { alert("There has been a problem loading the data.");});

```

Example 7: Calling the `processData()` function (in: `main.js`).

```

1 function processData(data) {
2     var timestamps = [];
3     var min = Infinity;
4     var max = -Infinity;
5
6     for (var feature in data.features) {
7
8         var properties = data.features[feature].properties;
9
10        for (var attribute in properties) {
11
12            if ( attribute != 'id' &&
13                attribute != 'name' &&
14                attribute != 'lat' &&
15                attribute != 'lon' ) {
16
17                if ( $.isArray(attribute,timestamps) === -1) {
18                    timestamps.push(attribute);
19                }
20
21                if (properties[attribute] < min) {
22                    min = properties[attribute];
23                }
24
25                if (properties[attribute] > max) {
26                    max = properties[attribute];
27                }
28            }
29        }
30    }
31
32    return {
33        timestamps : timestamps,
34        min : min,
35        max : max
36    }
37 }

```

Example 8: Processing the GeoJSON (in: `main.js`).

The `processData()` function then makes use of a nested looping structure to determine the values for these three local variables. First, a `for` loop is used to traverse each of the features in the `data` variable, treating each of the map features included in the GeoJSON one at a time (EX8: 6). Next, the properties associated with the given feature (i.e., the header names for all attributes in the GeoJSON) are stored in a local variable called `properties` (EX8: 8). A second `for` loop then is used to traverse through each attribute in the `properties` variable (EX8: 10). In other words, this nested looping structure accesses each map feature in the GeoJSON individually, and then accesses each of the attributes associated with a given feature individually before moving onto the next map feature.

Once a single attribute of a single map feature is isolated using the nested looping structure, the attribute is evaluated according to four `if` statements to determine if it influences the derived information (e.g., if it is a new timestamp name or the min/max value):

1. An `if` statement first is included to test if the current attribute is one of the included timestamps (e.g., 2005, 2006), or if it instead is the `id`, `name`, `lat`, or `long` column in the GeoJSON (EX8: 12–15). You will need to modify the set of conditions included in the `if` statement if you added additional columns to your GeoJSON, or gave the columns different header names.

2. If the attribute is one of the timestamps (i.e., if it conforms to the aforementioned conditions), then it is appended to the end of the `timestamps` array (EX8: 18). This `push()` call is encapsulated within an `if` statement that checks if the given timestamp name already has been added to the `timestamps` array (i.e., if 2005 already exists in the array) (EX8: 17–19); if it does not exist (`===`), a value of -1 is returned and the attribute name is appended to the `timestamp` array.
3. Next, an `if` statement is used to check if the value of the current attribute for the current feature is smaller than the current value assigned to the `min` variable (EX8: 21–23). If the value is smaller, then the `min` value is replaced with the attribute value of the current feature.
4. Finally, an `if` statement is used to check if the value of the current attribute for the current feature is larger than the current value assigned to the `max` variable (EX8: 25–27). The logic in this `if` statement is conceptually opposite to that used to update the `min` variable.

Once the nested looping structure works through all properties of all features, the `timestamp`, `min`, and `max` variables are returned (EX8: 32–36), concluding the `processData()` function.

6. DRAWING THE PROPORTIONAL SYMBOLS

WITH THE GEOJSON loaded and processed, it is now time to add the proportional symbols to the map. Leaflet supports the overlay of map symbols, or *markers*, using either pre-rendered iconic point symbols (e.g., in `.png` format) or dynamically drawn scalable vector graphics (SVG). Because SVG is a vector image format rendered in the browser, SVG markers can be easily resized with a change

in the time series data. This advantage makes SVG the preferred format for thematic web mapping generally.

To add markers to the map, first return to the `done()` function and note that there are now two local variables based on the GeoJSON: (1) the `data` object containing the GeoJSON and (2) the `info` array containing the three

```
1 $.getJSON("data/cityData.json")
2   .done(function(data) {
3       var info = processData(data);
4       createPropSymbols(info.timestamps, data);
5   })
6   .fail(function() { alert("There has been a problem loading the data.");});
```

Example 9: Calling the `createPropSymbols()` Function (in: `main.js`).

```

1 function createPropSymbols(timestamps, data) {
2
3     cities = L.geoJson(data).addTo(map);
4
5 }

```

Example 10: Adding Teardrop Markers to the Map (in: main.js).

derived variables returned by the `processData()` function (Example 9). Following the call to `processData()`, call a new functional named `createPropSymbols()`, passing `info.timestamps` and `data` as parameters (EX9: 4).

Next, define the `createPropSymbols()` function, adding it after the `processData()` definition (Example 10). Because of the popularity of the GeoJSON format, Leaflet offers the method `L.geoJson()` to create a new `GeoJson FeatureGroup` from the geographic information contained within a GeoJSON file. A `GeoJson FeatureGroup` is a specialized type of `FeatureGroup`, a Leaflet class that is used to group multiple map layers together, allowing for the group to be treated as one whole programmatically. When using Leaflet, a “layer” refers to a single map feature (e.g., a

point marker, a polygon), meaning that the `FeatureGroup` is closer to the concept of a “layer” or “geometry Collection” in GIS software. Refer to the Leaflet documentation for additional details about the `FeatureGroup` class: leaflet-js.com/reference.html#featuregroup. Create a `GeoJson FeatureGroup` and assign it to the previously declared `cities` variable using `L.geoJson()`, passing the data object as the parameter (EX10: 3). Then call the `addTo()` function, passing the global map variable as the parameter in order to place the markers onto the map.

Save your changes to the `main.js` file and refresh the `index.html` page in the browser. You now should see your map features added to the map as inverted teardrop markers, the Leaflet default for point features (Figure 5).

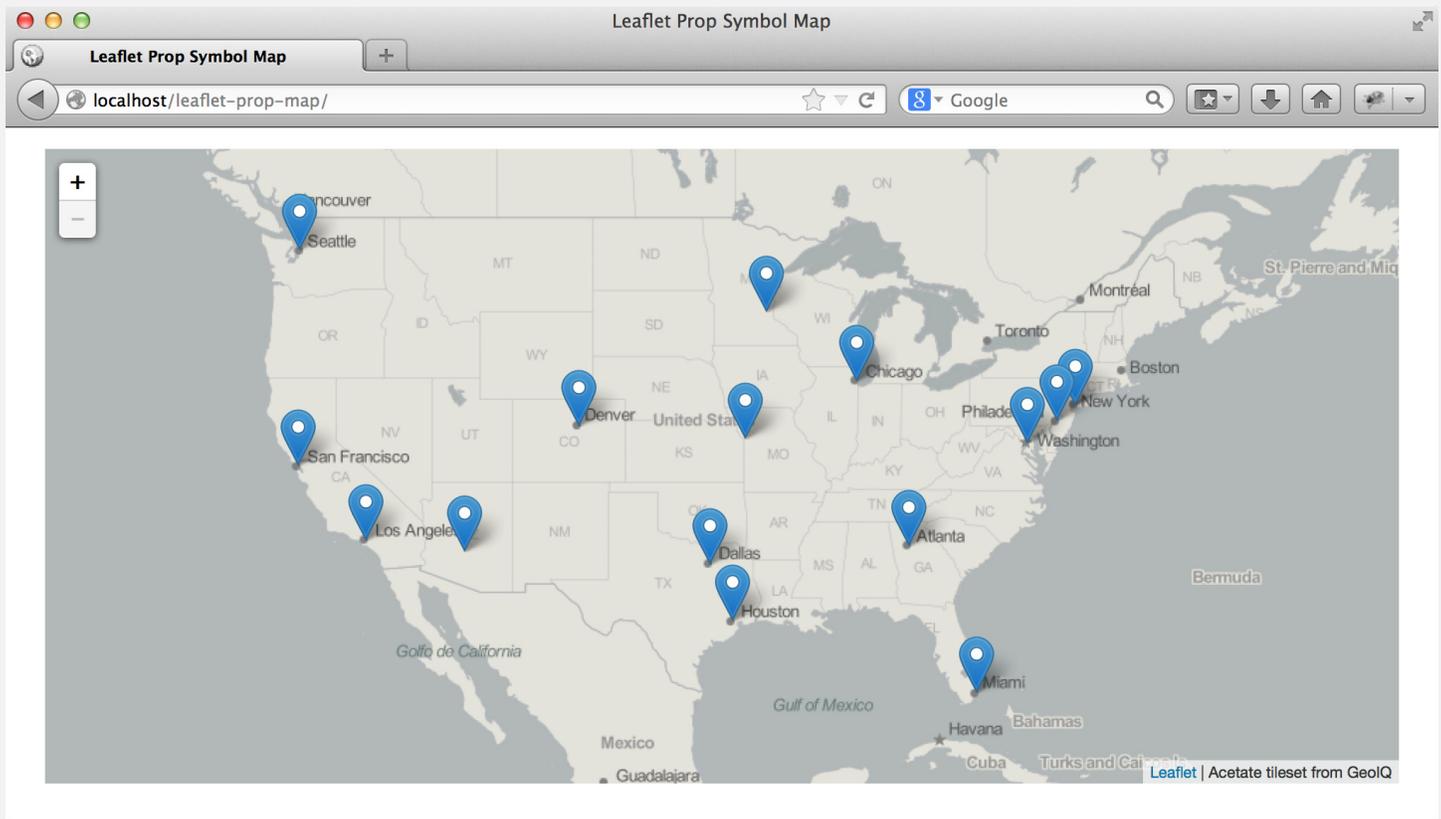


Figure 5: Adding markers to the map.

```

1  function createPropSymbols(timestamps, data) {
2
3      cities = L.geoJson(data, {
4
5          pointToLayer: function(feature, latlng) {
6
7              return L.circleMarker(latlng, {
8                  fillColor: "#708598",
9                  color: '#537898'
10                 weight: 1,
11                 fillOpacity: 0.6
12             }).on({
13
14                 mouseover: function(e) {
15                     this.openPopup();
16                     this.setStyle({color: 'yellow'});
17                 },
18                 mouseout: function(e) {
19                     this.closePopup();
20                     this.setStyle({color: '#537898'});
21                 }
22             });
23         });
24     }).addTo(map);
25 }
26 }

```

Example 11: Replacing teardrop markers with circle markers and adding event listeners for a popup window (in: main.js).

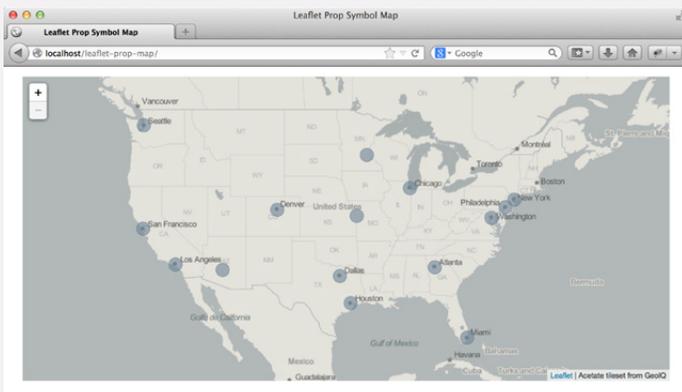


Figure 6: Drawing circles on the map.

A non-compact, teardrop symbol is not ideal for proportional symbol mapping. Make use of Leaflet’s `pointToLayer()` function to draw custom SVG markers for the proportional symbols, rather than using the teardrop images (Example 11). Using `pointToLayer()`, you are able to draw each proportional symbol as a Leaflet

`CircleMarker`, giving you control over the styling of the map symbol. The tutorial example manipulates the color (EX11: 8–9), stroke width (EX11: 10), and opacity (EX11: 11) of the proportional symbols. Additional details about the `CircleMarker` class, are available at: leafletjs.com/reference.html#circlemarker.

Again save your changes to the `main.js` file and refresh the `index.html` page in the browser. You now should see your map features added as partially transparent blue symbols, centered upon the lat/long location of the map feature (Figure 6).

Before moving on, append an additional method `on()` to the `pointToLayer()` callback function, which adds a pair of event listeners to open and close the popup window upon `mouseover` and `mouseout`, respectively (EX11: 12–23). Note that the popup window and content is not yet bound to these symbols, which we will do in the subsequent steps (see EX13: 14).

7. SCALING THE PROPORTIONAL SYMBOLS

AFTER DRAWING the SVG markers to the map, you now need to add the functionality to resize each marker according to a value in the time series. Such a function needs to be applied uniquely to each `CircleMarker` layer in the newly created `cities` `GeoJson` `FeatureGroup` layer, as each proportional symbol on your map has a different set of attributes values (i.e., a differently sized proportional symbol), and these values vary different over the time series.

To resize the proportional symbols, first return to the `createPropSymbols()` function and add a call to a new function named `updatePropSymbols()`. The call to this new function should come at the end of the

`createPropSymbols()` definition, after the `cities` `FeatureGroup` is added to the map (**EX12: 26**). The `updatePropSymbols()` function takes as a parameter the value stored in the first index position of the `timestamps` array (i.e., the first date in the time series).

Next, declare and define two new functions: 1) the aforementioned `updatePropSymbols()` function used to resize each proportional symbol individually and 2) a `calcPropRadius()` function providing the math to compute the appropriate size of a proportional symbol given its attribute value (**Example 13**). These pair of functions should be defined in `main.js` after the `createPropSymbols()` function.

```
1 function createPropSymbols(timestamps, data) {
2
3     cities = L.geoJson(data, {
4
5         pointToLayer: function(feature, latlng) {
6
7             return L.circleMarker(latlng, {
8
9                 fillColor: "#708598",
10                color: '#537898'
11                weight: 1,
12                fillOpacity: 0.6
13            }).on({
14
15                mouseover: function(e) {
16                    this.openPopup();
17                    this.setStyle({color: 'yellow'});
18                },
19                mouseout: function(e) {
20                    this.closePopup();
21                    this.setStyle({color: '#537898'});
22                }
23            });
24        });
25    }
26    }).addTo(map);
27
28    updatePropSymbols(timestamps[0]);
29
30 }
```

Example 12: Updating the proportional circles by timestamp (in: main.js).

```

1    function updatePropSymbols(timestamp) {
2
3        cities.eachLayer(function(layer) {
4
5            var props = layer.feature.properties;
6            var radius = calcPropRadius(props[timestamp]);
7            var popupContent = "<b>" + String(props[timestamp]) +
8                " units</b><br>" +
9                "<i>" + props.name +
10               "</i> in </i>" +
11               timestamp + "</i>";
12
13           layer.setRadius(radius);
14           layer.bindPopup(popupContent, { offset: new L.Point(0,-radius) });
15       });
16   }
17   function calcPropRadius(attributeValue) {
18
19       var scaleFactor = 16;
20       var area = attributeValue * scaleFactor;
21       return Math.sqrt(area/Math.PI)*2;
22   }

```

Example 13: Scaling the proportional circles (in: main.js).

The `updatePropSymbols()` function begins by calling the Leaflet `eachLayer()` method on the `cities` GeoJson `FeatureGroup` (EX13: 3). The `eachLayer()` is a simplified loop offered by Leaflet that applies the same logic (here a new function definition) to every layer included in a `FeatureGroup`. Within the new `function()` definition, three local variables are defined: 1) `props`, storing the complete set of attributes for the given proportional symbol (EX13: 5), 2) `radius`, storing the attribute value of the proportional symbol for the current `timestamp` (EX13: 6), and 3) `popupContent`, storing the markup text used to populate an information window popup (EX13: 7–11). The latter variable can be modified to include whatever information you wish to present to the user upon probing a proportional symbol.

Note that assignment of the `radius` variable makes use of the custom `calcPropRadius()` method (EX13: 17–22), which takes the attribute value of the proportional symbol and multiplies it against an arbitrary `scaleFactor` (here, the value of 16 is hardcoded given this particular sample dataset) to determine the area of proportional circle (EX13: 19–20). Experiment with the `scaleFactor` to

find a value that works well with your dataset; the larger the `scaleFactor`, the larger all proportional symbols will be. The `radius` then is calculated and returned to the `updatePropSymbols()` function, as the `CircleMarker` class scales a marker by a radius value rather than an area value. This geometry logic is included in a separate `calcPropRadius()` function, rather than embedded in the `updatePropSymbols()` function, so that it also can be used to resize the symbols included in the map legend.

The `updatePropSymbols()` function proceeds by calling two methods from the `CircleMarker` class on the currently treated layer: 1) `setRadius()`, which adjusts the size of the proportional symbol (EX13: 13) and 2) `bindPopup()`, which binds the aforementioned `popupContent` markup text to the proportional symbol (EX13: 14).

Save your changes to the `main.js` file and refresh the `index.html` page in the browser. You now should see your proportional symbols scaling according to the first timestamp (Figure 7). Your proportional symbols also should have popup functionality to retrieve details about the probed symbol.

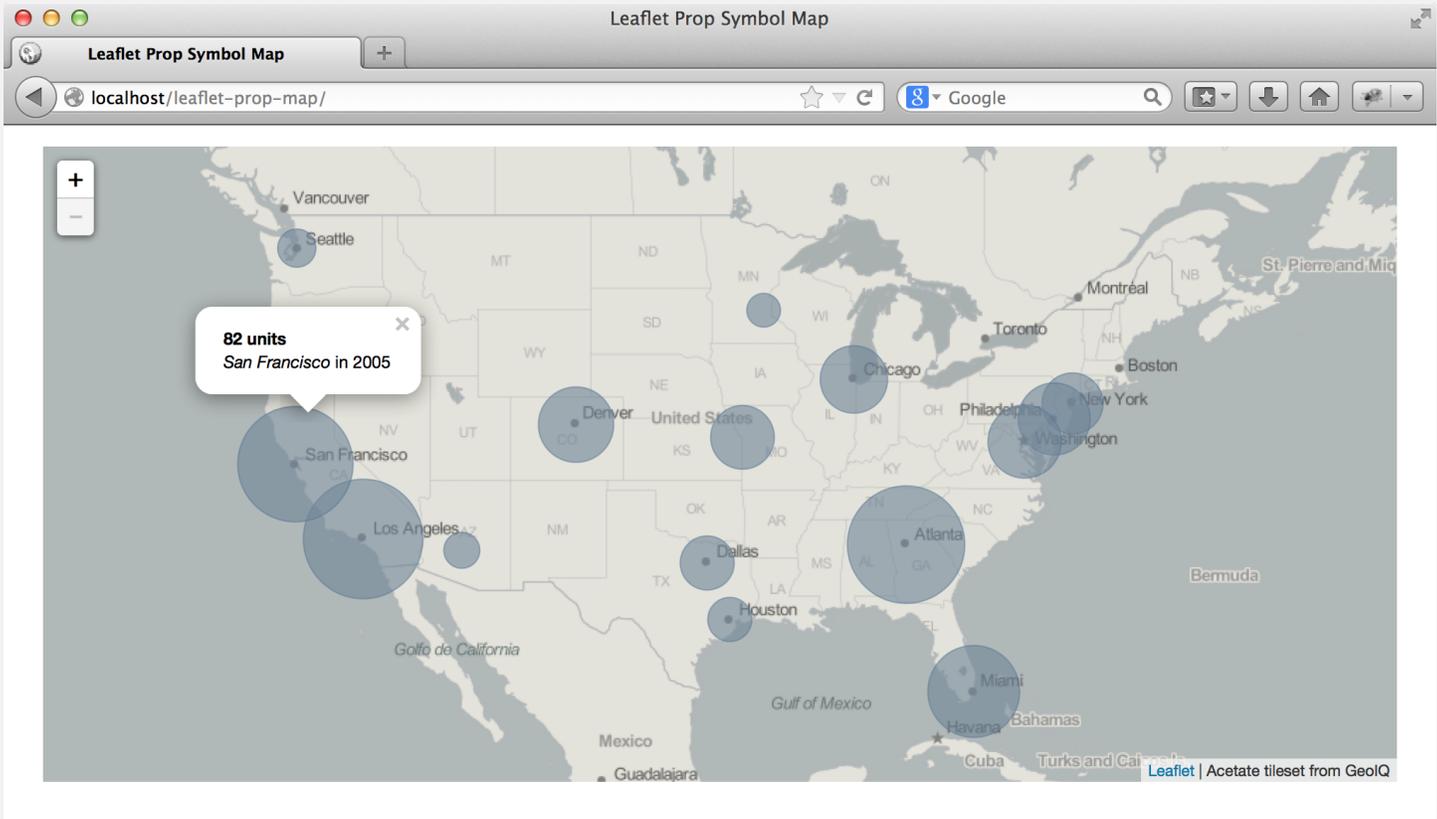


Figure 7: Scaling the proportional symbols and binding a popup window to symbols.

8. CREATING A MAP LEGEND

WHILE THE POPUP WINDOW provides a way to determine the specific value of each proportional symbol, it is conventional also to include a persistent map legend indicating the values of several example symbols. The following tutorial creates a map legend using HTML elements, in this case simple `<div>` elements styled with the CSS property `border-radius` value of `50%`, which rounds the corners of the `<div>` elements to make them appear as circles (EX16: 19). The `calcPropRadius()` method then is used to dynamically resize the legend symbols.

First return to the `done()` function and add a call to a new function named `createLegend()`. The call to this new

function should follow the existing call to the `createPropSymbols()` function (EX14: 5). The `createLegend()` function takes as parameters the minimum and maximum values across the time series, as identified through the `processData()` function.

Next, define the `createLegend()` function, placing the function definition after the `calcPropRadius()` definition (Example 15). The `createLegend()` function makes use of the Leaflet `L.control()` method, which adds a new UI element to the map, and the `L.DomUtil()` method for creating a new DOM entity. Read more about these methods in the API reference at leafletjs.com/reference.

```

1    $.getJSON("data/cityData.json")
2      .done(function(data) {
3        var info = processData(data);
4        createPropSymbols(info.timestamps, data);
5        createLegend(info.min, info.max);
6      })
7    .fail(function() { alert("There has been a problem loading the data.");});

```

Example 14: Calling the `createLegend()` Function (in: `main.js`).

[html#icontrol](#) and [leafletjs.com/reference.html#domutil](#).

The `createLegend()` function begins by assessing the `min` value passed as a parameter, changing it to a value of 10 if below 10 so that the small legend symbol remains visible in the webpage (EX 15: 3–5). You also can add logic here to place a ceiling on the `max` value, if desired. A local function named `roundNumber()` then is defined that rounds the input value to the nearest increment of 10 (EX15: 7–10); again, you can adjust this function to round to a different value (25, 100, etc.) depending on your time series dataset.

The `createLegend()` function proceeds by using `L.control()` to create a new control named `legend` that contains the map legend, setting the `position` style to `bottomright` on the map (EX15: 12). Leaflet's `onAdd()` event listener then is attached to the `legend` control so that the legend is configured only after added to the map. The `onAdd()` event listener defines a new `function()` that first declares seven local variables:

1. `legendContainer`, a wrapper `<div>` that holds the graphic and text elements in the legend (EX15: 16); note that the `DomUtil()` function is evoked to add the `<div>` as a page element;
2. `symbolsContainer`, a `<div>` that contains the example proportional symbols in the legend (EX15: 17); again, `DomUtil()` is used to add the `<div>` to the webpage;
3. `classes`, an array holding the values of the `min`, `max`, and a third value in the middle of the attribute range (EX15: 18); this solution produces a legend with three example proportional symbols, but can be modified to include a different number of example symbols;
4. `legendCircle`, an unassigned variable used repeatedly to restyle each of the three legend proportional symbols (EX15: 19);
5. `lastRadius`, a variable assigned an initial value of zero, which will be used to store the value of the previous symbol's radius while looping through the `classes` array (EX15: 20);
6. `currentRadius`, a unassigned variable used to store the current symbol's radius while looping through the `classes` array (EX15: 21);

7. `margin`, an unassigned variable used to store the relative pixel distance of each of the legend's proportional symbols from the left side of their parent container, in order to horizontally align them with each other (EX15: 22).

Before adding the logic needed to draw the legend, first disable the panning of the tiled basemap underneath the legend. To do this, pass the `mousedown` event into the callback `function()` and use the `stopPropagation()` method to prevent the click behavior from being applied to the legend's parent object, the Leaflet map (EX15: 24–26).

Once the `mousedown` event is disabled, the `createLegend()` function continues by selecting the newly created `legendContainer` element and adding an `h2` header element to it (EX15: 28). A `for` loop then is used to iterate through the three values within the `classes` array and to add new `<div>` elements to the `legendContainer` and provide widths that are proportionate to twice the values in the `classes` array (EX15: 30–47). The `for` loop first creates a new `<div>` element for the given legend symbol, storing it in the previously declared `legendCircle` variable so that CSS rules can be applied to the `<div>` (EX15: 32). The radius of the proportional symbol then is calculated (EX15: 34). With each iteration through the `for` loop, the value assigned to `margin` is calculated using the values of `currentRadius` and `lastRadius` (EX15: 34). Because each of the three `legendCircle` `<div>` elements is given a display property of `inline-block` (EX16: 22) within the external style sheet, they will normally flow alongside each other within the layout of the `legendContainer`. Deriving a negative value for the left margin of each allows them to be stacked on top of one another and vertically aligned, producing a nested display result. As the loop iterates from the smallest symbol to the largest, the negative left margin value is calculated using the current symbol's width, the previous symbol's width, as well as two additional pixel value units to account for the 1px border applied to the symbols within the external style sheet (EX16: 20). The newly created `legendCircle` element is then selected using `jQuery`, given a `width` attribute based on its current radius (multiplied by two in this case to fill the full width or diameter of the `div` element), a `height` attribute of the same value, and the calculated `margin` value to offset the symbol's `margin-left` property (EX15: 38–40). The `legendCircle` then is appended to the `symbolsContainer` and the `lastRadius` is set to the `currentRadius` value before the loop iterates again.

```

1  function createLegend(min, max) {
2
3      if (min < 10) {
4          min = 10;
5      }
6
7      function roundNumber(inNumber) {
8
9          return (Math.round(inNumber/10) * 10);
10     }
11
12     var legend = L.control( { position: 'bottomright' } );
13
14     legend.onAdd = function(map) {
15
16         var legendContainer = L.DomUtil.create("div", "legend");
17         var symbolsContainer = L.DomUtil.create("div", "symbolsContainer");
18         var classes = [roundNumber(min), roundNumber((max-min)/2), roundNumber(max)];
19         var legendCircle;
20         var lastRadius = 0;
21         var currentRadius;
22         var margin;
23
24         L.DomEvent.addListener(legendContainer, 'mousedown', function(e) {
25             L.DomEvent.stopPropagation(e);
26         });
27
28         $(legendContainer).append("<h2 id='legendTitle'># of somethings</h2>");
29
30         for (var i = 0; i <= classes.length-1; i++) {
31
32             legendCircle = L.DomUtil.create("div", "legendCircle");
33
34             currentRadius = calcPropRadius(classes[i]);
35
36             margin = -currentRadius - lastRadius - 2;
37
38             $(legendCircle).attr("style", "width: " + currentRadius*2 +
39                 "px; height: " + currentRadius*2 +
40                 "px; margin-left: " + margin + "px" );
41             $(legendCircle).append("<span class='legendValue'>" + classes[i] + "</span>");
42
43             $(symbolsContainer).append(legendCircle);
44
45             lastRadius = currentRadius;
46
47         }
48

```

Example 15: Creating a map legend (in: main.js). Continued next page.

```

49     $(legendContainer).append(symbolsContainer);
50
51     return legendContainer;
52
53     };
54
55     legend.addTo(map);
56
57 } // end createLegend()

```

Example 15, continued.

After adding all three proportional symbols to the `symbolsContainer` `<div>` and concluding the for loop, the `symbolsContainer` `<div>` is appended to the `legendContainer` `<div>` (EX15: 49). Finally, the `legendContainer` `<div>` is returned to the `legend` variable through the `onAdd()` callback function() (EX15: 51). The

`createLegend()` function concludes by adding the legend control to the map (EX15: 55).

Save your changes to the `main.js` file and refresh the `index.html` page in the browser. You now should see a map legend in the bottom, right corner of the map (Figure 8).

```

1     .legend, .temporal-legend {
2         padding: 6px 10px;
3         font: 14px/16px Arial, Helvetica, sans-serif;
4         background: white;
5         background: rgba(255,255,255,0.8);
6         box-shadow: 0 0 15px rgba(0,0,0,0.2);
7         border-radius: 5px;
8     }
9     #legendTitle {
10        text-align: center;
11        margin-bottom: 15px;
12        font-variant: small-caps;
13    }
14    .symbolsContainer {
15        float: left;
16        margin-left: 50px;
17    }
18    .legendCircle {
19        border-radius:50%;
20        border: 1px solid #537898;
21        background: rgba(113, 133, 152, .6);
22        display: inline-block;
23    }
24    .legendValue {
25        position: absolute;
26        right: 8px;
27    }

```

Example 16: Style rules for creating a nested proportional symbol legend using `div` elements (in: `style.css`).

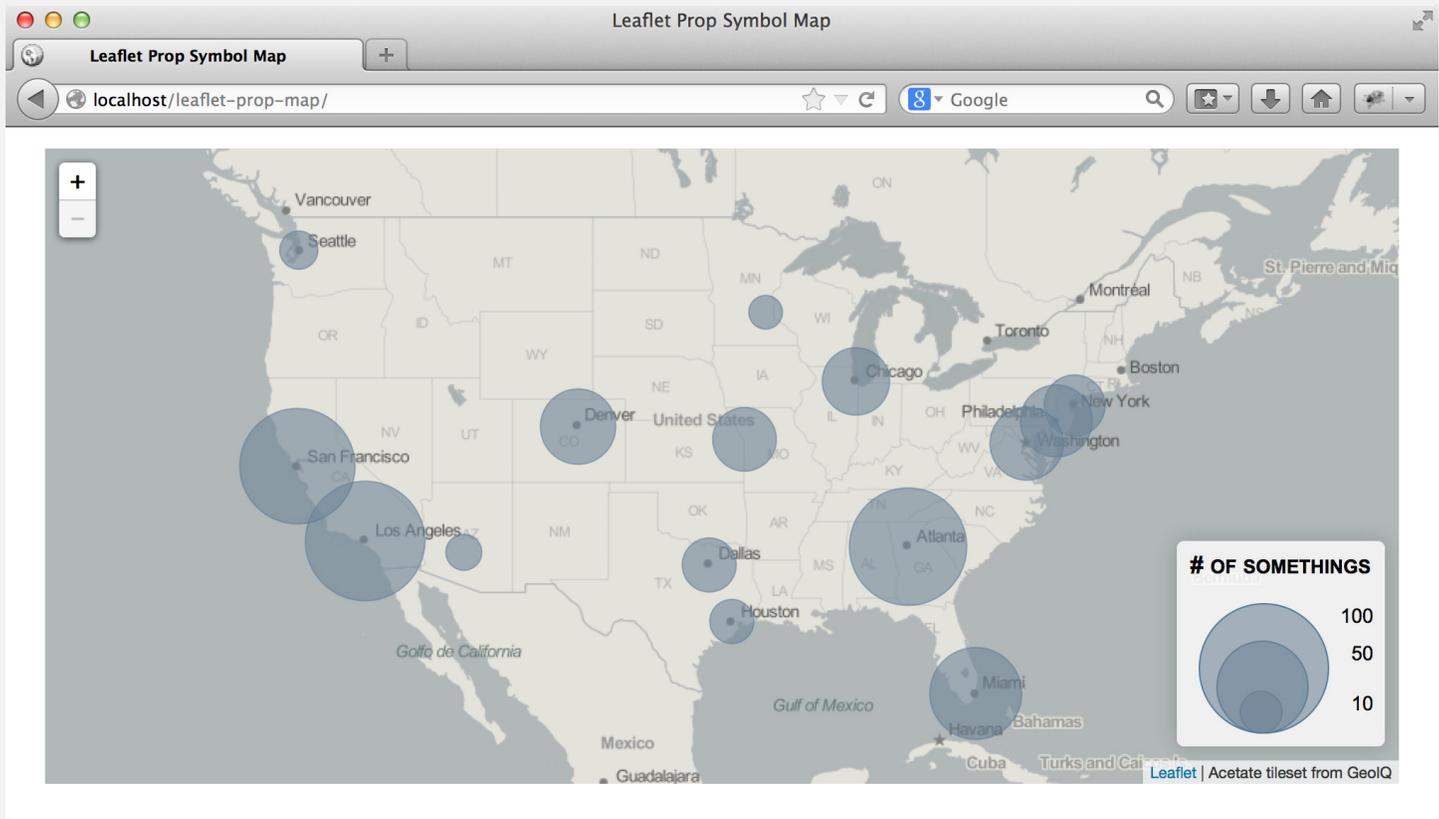


Figure 8: Adding a map legend.

9. ADDING A TEMPORAL SLIDER

THE NEXT STEP in completing the time series proportional symbol map is implementation of a temporal slider for displaying the time series. A *slider* is a UI widget that allows users to set the value of an ordinal or, more commonly, numerical variable; checkboxes (allowing compound selection of multiple values) or radio buttons (constraining selection to a single value in a set) are used for categorical variables. A temporal slider thus allows the user to change the current timestamp, updating the map to any point in the time series. A slider widget works best for depictions of linear time rather than cyclical time, following a timeline metaphor rather than a clock metaphor. Several options exist for implementing a slider widget within a web page (we will implement option 3 in this tutorial):

1. **jQueryUI** is a plugin library for jQuery that supports a range of common UI widgets. The jQueryUI plugin includes default graphics needed for the interface widgets as well as associated events and effects for implementing these widgets. Before getting started with jQueryUI, review the jQueryUI API Documentation and the jQueryUI demo pages at jqueryui.com/.

2. **noUiSlider** is a smaller jQuery plugin written specifically to create a range slider element, rather than the host of UI widgets supported by jQueryUI. The code for this plugin is available at: refreshless.com/nouislider/.
3. Finally, the HTML5 specification now includes a range type for the `<input>` element: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/Input>. The range type makes it easy to create a simple slider and apply basic styles to it. Note that the range type is a W3C recommendation and still is in the process of gaining support among web browsers. If support among older browsers is important, you may wish to use one of the first two plugins mentioned above. Their implementation will be very similar to the process described here.

To implement the temporal slider, return to the `done()` function and add a call to a new function named `createSliderUI()` (EX17: 6). Pass the `timestamps` array as the argument with this function call. This is the last

```

1 $.getJSON("data/cityData.json")
2   .done(function(data) {
3       var info = processData(data);
4       createPropSymbols(info.timestamps, data);
5       createLegend(info.min,info.max);
6       createSliderUI(info.timestamps);
7   })
8   .fail(function() { alert("There has been a problem loading the data.");});

```

Example 17: Calling the createSliderUI() function (in: main.js).

function invoked from within the done() callback function().

Next, define the createSliderUI() function, placing the function definition after the createLegend() definition (Example 18). The createSliderUI() function first uses the L.Control() method to add a new control named sliderControl to the map (EX18: 3). Note that there are now two Leaflet controls added to the map: one for the

map legend and one for the slider control, with the latter positioned in the bottom, left corner of the map.

Leaflet's onAdd() event listener then is attached to sliderControl to configure the temporal slider after it is added to the map (EX18: 5–24). The onAdd() event listener first adds a new <input> element to the DOM named slider using the L.DomUtil() function (EX18: 7); the <input> element is given the class name range-slider so that it can be styled. As with the map legend above, the

```

1 function createSliderUI(timestamps) {
2
3     var sliderControl = L.control({ position: 'bottomleft' });
4
5     sliderControl.onAdd = function(map) {
6
7         var slider = L.DomUtil.create("input", "range-slider");
8
9         L.DomEvent.addListener(slider, 'mousedown', function(e) {
10             L.DomEvent.stopPropagation(e);
11         });
12
13         $(slider)
14             .attr({'type':'range',
15                 'max': timestamps[timestamps.length-1],
16                 'min': timestamps[0],
17                 'step': 1})
18             .on('input change', function() {
19                 updatePropSymbols($(this).val().toString());
20             });
21         return slider;
22     }
23
24     sliderControl.addTo(map)
25 }

```

Example 18: Creating a temporal slider (in: main.js).

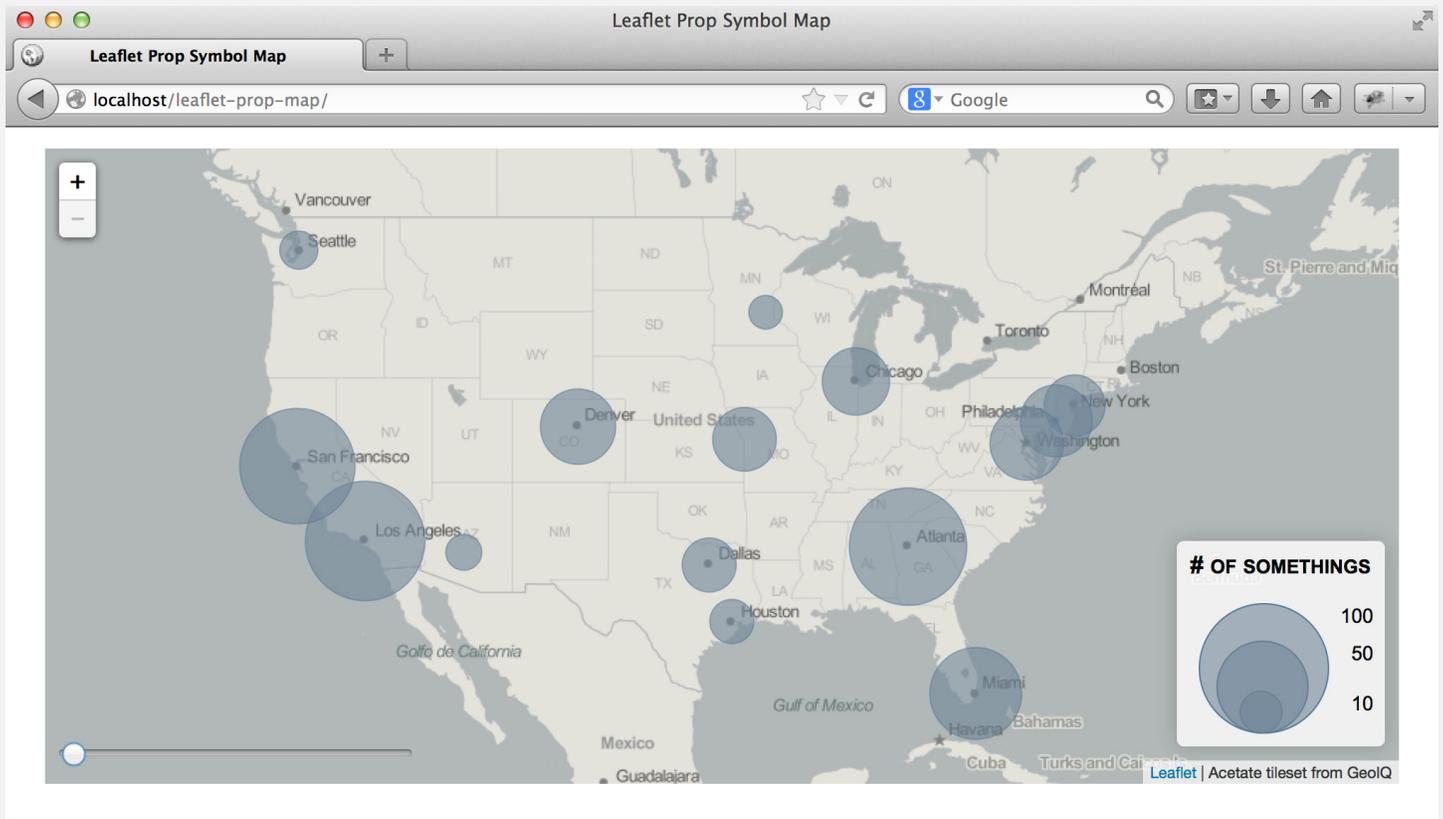


Figure 9: Adding a temporal slider.

`stopPropagation()` method is applied to the slider to prevent the click behavior from being propagated to the slider's parent object, the Leaflet map (EX18: 9–11).

Next, two methods are called on the newly created slider `<input>` element using jQuery. First, the `attr()` method is called to set four properties of the slider element:

1. the `type`, using the aforementioned range type (EX18: 14);
2. the maximum value of the slider, using the last value in the timestamps array (EX18:15)
3. the minimum value of the slider, using the first value in the timestamps array (EX18: 16);
4. the `step` interval, set to 1 to increment by one year for the tutorial example (EX18: 17).

The `on()` method then is called to listen for any change to the slider `<input>` element (EX18: 18–29). Traditionally this is behavior that needed to be written explicitly with the jQueryUI or noUiSlider JavaScript plugin, but is now supported within the browser itself. On any change to the slider (i.e., when the user interacts with it), the `updatePropSymbols()` function is called, passing the current value of the slider (i.e., the new timestamp value) to the `updatePropSymbols()` function (EX18: 19). Finally, the slider is returned to the `onAdd()` callback function() (EX18: 21). The `createSliderUI()` function concludes by calling the `addTo()` function, adding the `sliderControl` to the Leaflet map (EX18: 24).

Save your changes to the `main.js` file and refresh the `index.html` page in the browser. You now should see the slider widget in the bottom, left corner of the map (Figure 9). Take a second to play with the slider widget to ensure your complete time series dataset is loaded and mapped correctly.

10. CREATING A TEMPORAL LEGEND

FINALLY, THE TEMPORAL SLIDER requires a legend to alert the user to the current timestamp portrayed in the map. The temporal legend should update as the user interacts with the temporal slider. To create a temporal legend, make several modifications to the `createSliderUI()` function (**Example 19**):

1. First declare a fifth property named `value` for the slider `<input>` element (**EX19: 18**). This variable will store the name of the current timestamp (i.e., the header from your time series dataset). Assign the first timestamp as a default.
2. Next, add logic to the `on()` method to update this `value` property when the user changes the position of the slider `<input>` element (**EX19: 21**). Note that this logic actually changes the text of a page

element with the class name `temporal-legend`, which is an `<output>` element added to the DOM in the subsequently defined `createTemporalLegend()` function.

3. Finally, add a call to this new `createTemporalLegend()` function, passing the first value in the `timestamps` array as a parameter (**EX19: 27**).

Next, declare the `createTemporalLegend()` function at the bottom of the `main.js` file. This function is similar to the `createSliderUI()` function. First, a new control named `temporalLegend` is added to the bottom, left corner of the map using the `L.control()` method (**EX20: 3**). The `onAdd()` event listener then is called on the `temporalLegend` control, which creates an `<output>` element (a new HTML5 element used to represent the result of

```
1 function createSliderUI(timestamps) {
2
3     var sliderControl = L.control({ position: 'bottomleft' });
4
5     sliderControl.onAdd = function(map) {
6
7         var slider = L.DomUtil.create("input", "range-slider");
8
9         L.DomEvent.addListener(slider, 'mousedown', function(e) {
10             L.DomEvent.stopPropagation(e);
11         });
12
13         $(slider)
14             .attr({'type': 'range',
15                 'max': timestamps[timestamps.length-1],
16                 'min': timestamps[0],
17                 'step': 1,
18                 'value': String(timestamps[0])})
19             .on('input change', function() {
20                 updatePropSymbols($(this).val().toString());
21                 $(".temporal-legend").text(this.value);
22             });
23         return slider;
24     }
25
26     sliderControl.addTo(map)
27     createTemporalLegend(timestamps[0]);
28 }
```

Example 19: Updating the temporal slider (in: `main.js`).

```

1  function createTemporalLegend(startTimestamp) {
2
3      var temporalLegend = L.control({ position: 'bottomleft' });
4
5      temporalLegend.onAdd = function(map) {
6          var output = L.DomUtil.create("output", "temporal-legend");
7          $(output).text(startTimestamp)
8          return output;
9      }
10
11     temporalLegend.addTo(map);
12 }

```

Example 20: Creating a temporal legend (in: main.js).

a calculation or user action) with the class name temporal-legend in the DOM named output (EX20: 6). The output variable storing the element is then passed to the jQuery .text() method to place the first timestamp as its content (EX20: 7), then returned to the callback function() (EX20: 8). Finally, the temporalLegend control

is added to the Leaflet map using the addTo() function (EX20: 11).

For one last time, save your changes to the main.js file and refresh the index.html page in the browser. Congratulations, you have made a time series proportional symbol map using Leaflet and jQuery (Figure 10)!

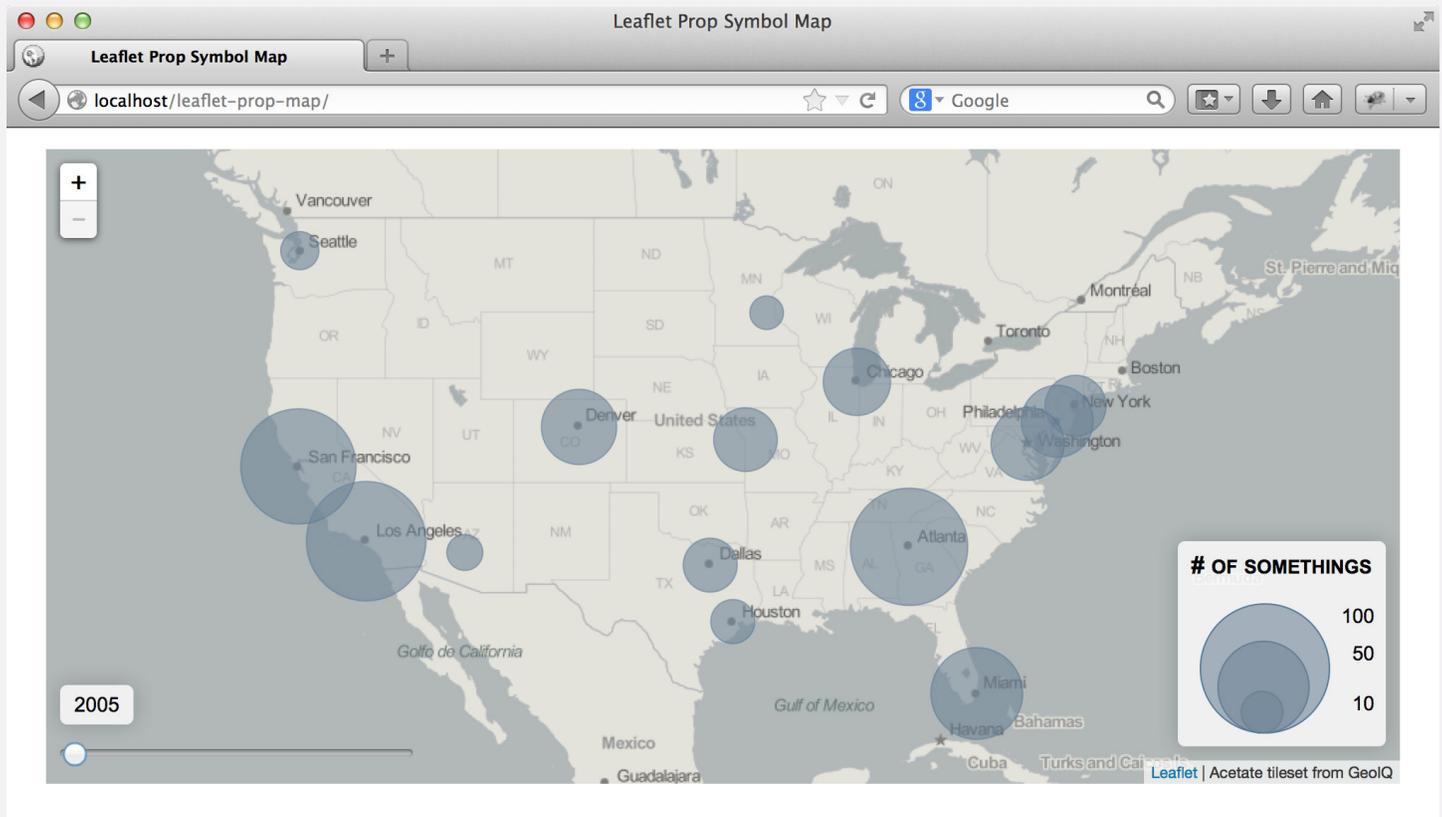


Figure 10: Adding a temporal legend.

THIS TUTORIAL PRESENTS the process and the tools to make a fairly impressive web map depicting time series information using proportional symbols. However, there are many ways you can improve your web map from this baseline. You may wish to enhance the existing map by contextualizing it within a webpage and providing pertinent supplemental content. Give the map a good title, cite your data sources, and consider how you can use additional text to help the map tell a meaningful story. How can the design of the webpage complement the objectives *and* aesthetics of the map?

You also may wish to extend the existing map and code, either through the representation of additional geographic information or by the support of additional user interaction. For example, consider how you could modify the script to add a second dataset of a different nominal type (and perhaps style with a different color)? How could you then use Leaflet's built-in methods (leafletjs.com/reference.html#map-stuff-methods) to provide the user with a

layer control to toggle these layers on and off? Another option would be to load in a second areal dataset and provide a choropleth map layer beneath the proportional symbols (see the Leaflet tutorial for creating an interactive choropleth: leafletjs.com/examples/choropleth.html).

Finally, you may wish to experiment with some of the additional features and functionality provided by the many Leaflet plugins (see leafletjs.com/plugins.html and <https://www.mapbox.com/mapbox.js/plugins>). If you have many point values to visualize, challenge yourself to implement the Leaflet heat map solution (www.mapbox.com/mapbox.js/example/v1.0.0/leaflet-heat) or the Markercluster plugin (github.com/Leaflet/Leaflet.markercluster). Reading and understanding other examples and solutions will greatly improve your ability to customize a map to create compelling graphic narratives. Most importantly, remember to have fun!

Census Mapping Mashup

Paul Hunt
University of Nebraska Omaha
phunt@unomaha.edu

By mandate, the United States Census Bureau compiles and distributes data on the American population. Open data initiatives have made it possible for users to access and analyze data with simple web-based tools. A new method for requesting data from the Census Bureau is described here, along with two different mapping mashups. Using the technology described in this article, a simple web mapping interface could unlock vast amounts of available data for user exploration.

INTRODUCTION

THE UNITED STATES CENSUS BUREAU collects and maintains a large, widely-used collection of data on general demographic, social, housing, and economic characteristics. Recently, they made their data available through an Application Programming Interface (API) (US Census Bureau 2014). Their API is a mechanism to access Census data through a set of web technologies referred to as AJAX (Asynchronous JavaScript and XML), which facilitate the continuous sending and receiving of data between client and server environments (Powell 2008). The main advantage of using the API is that the Census data are available for use without the need to store them on a local computer: they remain in the cloud.

JavaScript is the main programming language used with APIs; most are completely written in JavaScript.

Interpreted by the web browser, JavaScript is also the most widely used programming language for web development (Raasch 2013). As an interpreted computer language, there are no software packages that need to be installed, nor any special server-side setup. It is an efficient and simple solution for web-based application development and hosting.

The Esri JavaScript API (Esri 2014a) allows developers to use web services based on spatial data servers that implement their software. The Esri software packages and API are popular with government enterprise GIS systems. Data Driven Documents (D3) is another powerful set of JavaScript functions used to visualize large datasets. Both Esri and D3 provide JavaScript-based environments for creating mapping *mashups*, which are the pulling together of various online resources (Batty, et al. 2010).

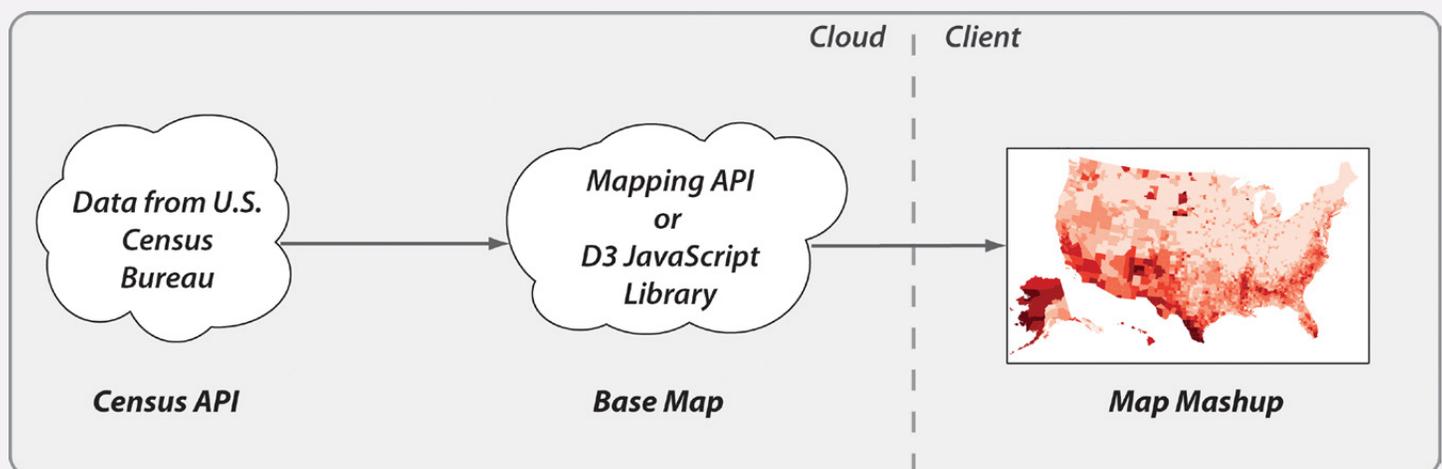


Figure 1: Representation of AJAX and JavaScript consuming web resources and a mashup of the attribute and spatial data in the client.

Two mashup methods are demonstrated here for mapping Census data obtained from the Census Bureau API. In the first, the Esri JavaScript API is used to obtain a base-map from the US Census Bureau TIGERweb spatial data

server (see Figure 1). In the second method, the base layer for D3 is provided in GeoJSON or TopoJSON format, which are compatible with JavaScript.

BACKGROUND

FOR MAPPING, most APIs are based on a technology called REST (Representational State Transfer) web services, which manipulate representations of web resources using a uniform set of stateless operations (Fielding 2000). One of the most popular map service architectures is the tiled web map service. This type of REST service involves a series of predefined static map tiles, produced for various scales, that populate the web map based on user interaction. There are additional REST mapping services, both raster and vector, that are the primary technology behind the plethora of Multi-Scale Panable maps available on the web (Peterson 2014). These mapping APIs provide the base layer of most map mashups.

There are alternatives for creating web map mashups. The D3 JavaScript Library can render a projected, SVG map element (base layer) that can be used to visualize spatial data (Cuesta 2013). D3 is designed for the creation of dynamic visualizations. In addition to mapping APIs, there are also APIs that revolve around accessing data. Data APIs focus on delivering specific queried data or streaming updated data. In a map mashup, these attribute (thematic) data sources, referred to as operational layers, are overlaid on top of the base layers.

CENSUS DATA API

AS A PRODUCT of the US Government, Census data have always been available free of charge. In the Internet era, there have been two main ways to access the hundreds of tables and thousands of spatial data elements derived from the Census. First, there are web-based query and extraction methods, the latest being the American Factfinder. Secondly, there is direct FTP access to the data that can then be processed locally within a relational database management system.

To promote access and innovation, the Census Bureau released an API in 2012 that would allow users to access data through properly formatted HTTP requests. This allows for AJAX methods to request and use data on-the-fly within a mashup environment. The Census API opens the door for alternative methods for developing mapping applications.

To make a data request, you must have a properly formatted HTTP string. This string begins with the Census API website, "api.census.gov/data." Next, the dataset you are querying is specified (i.e., Census 1990, 2000, 2010 or

ACS aggregate datasets). After this, you must provide your key, which is required to access the Census API and can be obtained by completing a short form at www.census.gov/developers/tos/key_request.html. Finally, the combination of variables and the spatial units are specified. Queries can be made directly in your web browser by pasting into your address bar the example requests listed in Example 1 (you will need to use your own key provided by the Census). A partial output of the first request is shown in Example 2: a two-dimensional array of data in the JSON (JavaScript Object Notation) format. The first row provides column names and subsequent rows contain the data values. The data in this array can then be mapped.

The Census API allows for up to 50 variables to be queried in a single request; a series of requests can unlock a vast amount of Census data for user mapping and analysis. In order to implement the Census API in a JavaScript mashup, the jQuery library is needed. JQuery is a free and widely adopted JavaScript library that has the built-in functions necessary for accomplishing common AJAX requests (Powell 2008).

1) URL for 2010 Census SF1 total population and name by for all states:

http://api.census.gov/data/2010/sf1?key=b48301d897146e8f8efd9bef3c6eb1fcb864cf&get=P0010001,NAME&for=state:*

2) URL for ACS 2010 5 Year data for Total Population for California and New York:

http://api.census.gov/data/2010/acs5?key=b48301d897146e8f8efd9bef3c6eb1fcb864cf&get=B02001_001E,NAME&for=state:06,36

3) URL for ACS 2011 5 Year data for Gross Rent as a % of Household Income, 10.0 to 14.9 percent for all counties in CA:

http://api.census.gov/data/2011/acs5?key=b48301d897146e8f8efd9bef3c6eb1fcb864cf&get=B25070_003E,NAME&for=county:*&in=state:06

4) URL for 2010 Census SF1 white population of 12 year olds in Alabama:

[http://api.census.gov/data/2010/sf1?key=\[user key\]&get=PCT012A015,PCT012A119&for=state:01](http://api.census.gov/data/2010/sf1?key=[user key]&get=PCT012A015,PCT012A119&for=state:01)

Example 1: Example Census API requests.

REFORMATTING

BEFORE THE DATA can be mapped, they must first be properly formatted. The two-dimensional array has certain limitations and is more usable if it is reformatted as key-and-value paired objects. In other words, we need to restructure the data from an array of individual elements into an array of record-like objects. The function shown in Example 3 produces the output shown in Example 4.

In this format, the data can be accessed more efficiently for database operations within a coding environment. As a two-dimensional array, the data would have to be referenced as a numeric [row], [column] of table elements. After reformatting, data items can be referenced by name

```
[ [ "P0010001" , "NAME" , "state" ],  
  [ "710231" , "Alaska" , "02" ],  
  [ "4779736" , "Alabama" , "01" ],  
  [ "2915918" , "Arkansas" , "05" ],  
  [ "6392017" , "Arizona" , "04" ],  
  [ "37253956" , "California" , "06" ], ... ]
```

Example 2: Results from the first request. This query has returned the total population (P0010001), state name (NAME), and Federal Information Processing Standards (FIPS) code (state). The FIPS code uniquely identifies the spatial unit and is used for joining the attribute data to the spatial components for mapping.

in a {key: value} pair and more easily mapped using JavaScript.

MAPPING

IN ORDER TO MAP the Census data, we need to acquire a basemap layer. After this, we dynamically join the reformatted Census data to their corresponding spatial counterparts as attributes. The map symbology (such as a choropleth) is then made based on those joined Census attributes.

MAPPING WITH THE ESRI JAVASCRIPT API

The Census' TIGERweb spatial data platform is based on the Esri ArcGIS Server software, which serves Open

Geospatial Consortium (OGC) compliant spatial data as web services. Since the data are OGC compliant, a mash-up could be done by using the OpenLayers API, an open source JavaScript API used for web mapping and consuming spatial data services. However, since the TIGERweb services are natively using Esri software, it is simpler to use Esri's freely-provided API.

To demonstrate the use of dual web services, the example here has two parts. First, the page is loaded with spatial

```

function getCensusData(){
  //jQuery AJAX function getting data from the Census API
  //and call a return function processing the Census JSON object in 'data' variable
  $.getJSON("http://api.census.gov/data/2010/sf1?key=[user key]&get=P0010001,P0050010,NAME&for=state:*",
  function(data){
    var keys = data[0]; //extract the first row of the returned 2d array that are the column headers
    var values = data; //copy the array
    values.splice(0,1); //delete the first row of headers in the copied array
    arrayCensus = []; //create a new array to store the formatted object outputs
    //nested loops combining the column header with appropriate values as {key:value} pair objects
    for(var i = 0 ; i < values.length; i++){
      var obj = {};
      for(var j = 0 ; j < values[i].length; j++){
        obj[keys[j]] = values[i][j];
      }
      arrayCensus.push(obj);
    }
  });
}

```

Example 3: Sample function that requests data using the Census API and then reformats the results into a usable array of objects that can be mapped.

```

[ { "P0010001" : "4779736" , "NAME" : "Alabama" , "state" : "01" },
  { "P0010001" : "710231" , "NAME" : "Alaska" , "state" : "02" },
  { "P0010001" : "6392017" , "NAME" : "Arizona" , "state" : "04" },
  { "P0010001" : "2915918" , "NAME" : "Arkansas" , "state" : "05" },
  { "P0010001" : "37253956" , "NAME" : "California" , "state" : "06" } , ... ]

```

Example 4: The reformatted Census API request, now an array of objects and ready to be used as a mashup with either the Esri API or D3.

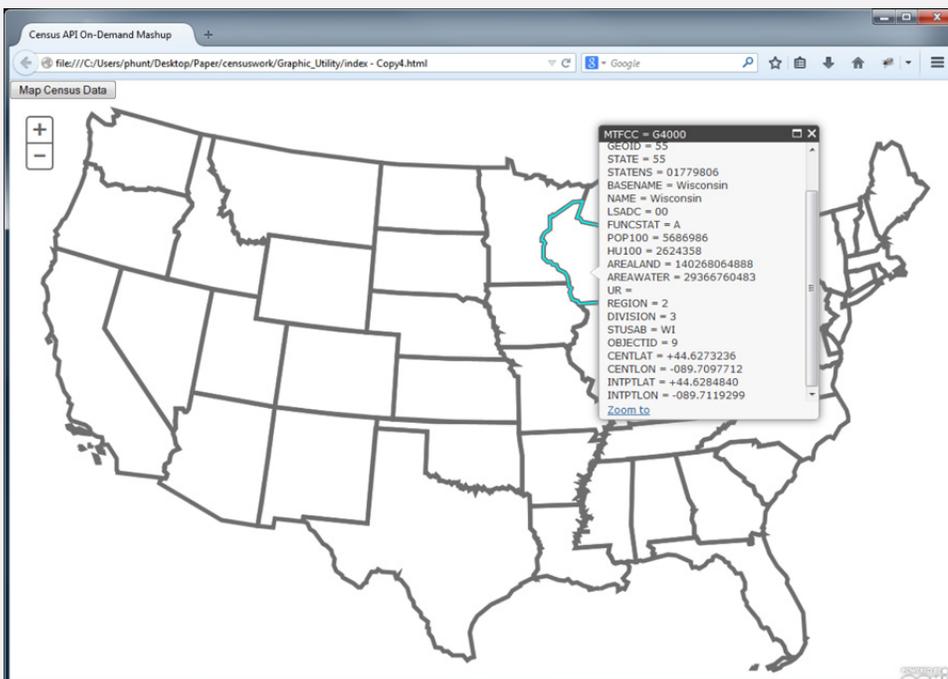


Figure 2: Initial loading of web page with the TIGERweb services.

data—in this case, data for the continental 48 states—derived from the TIGERweb web mapping services that is being consumed locally. In Figure 2, a feature has been selected to show the default popup window included with the Esri API. Second, there is a button marked “Map Census Data” in the top left corner of the page. Clicking it invokes the Census Data API AJAX request; when this button is pressed, the Census data are requested, reformatted, spatially joined, and symbolized. The resultant map is shown in Figure 3.

Example 5 shows the combination of Census and Esri code needed. Notice in Figure 3 that there is an additional

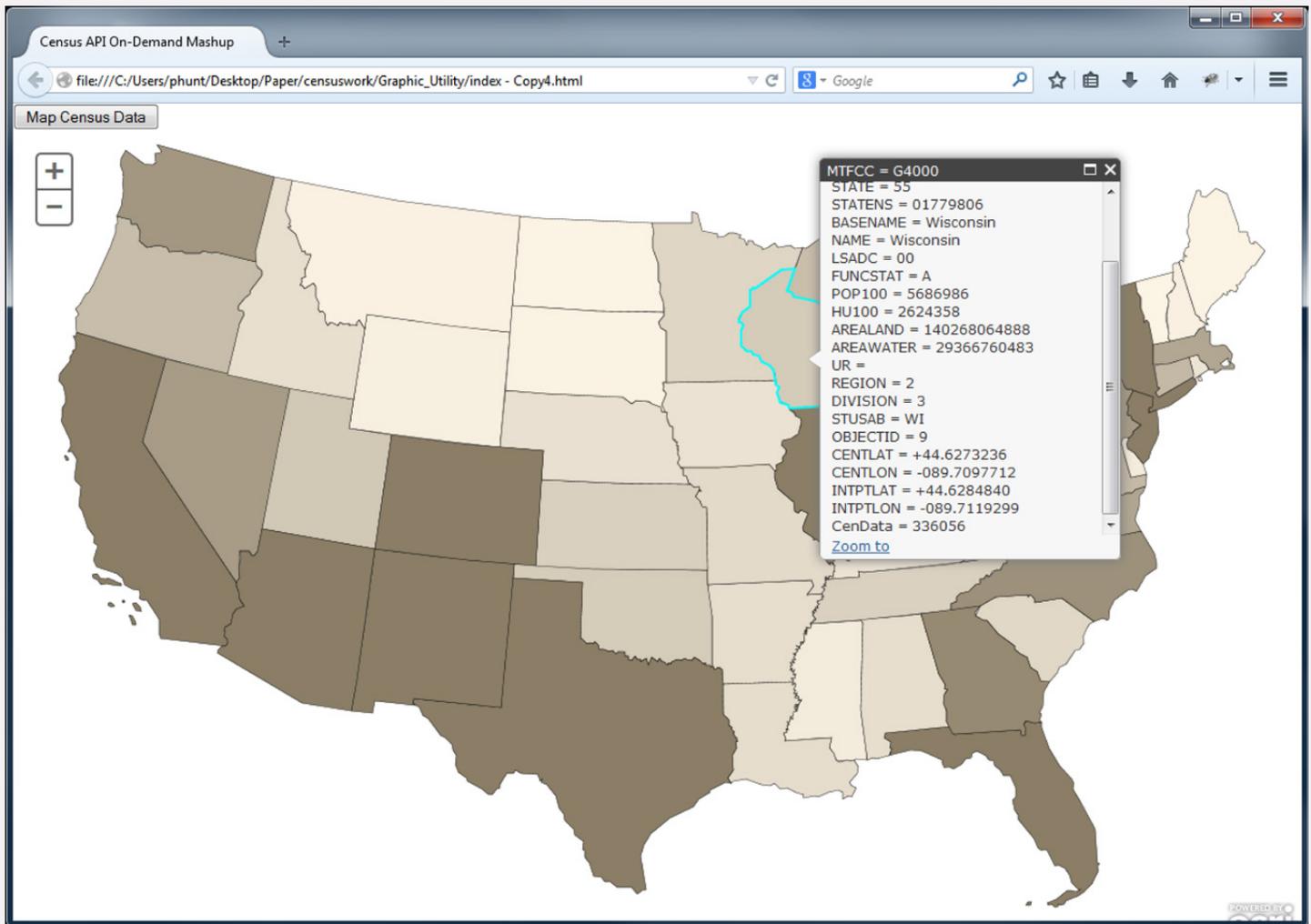


Figure 3: Map showing Total Hispanic Population (calculated CenData attribute) after Census data request, reformat, join, and symbolization.

```

<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <meta name="viewport" content="initial-scale=1, maximum-scale=1, user-scalable=no">
  <title> Census API On-Demand Mashup </title>
  <link rel="stylesheet" href="http://js.arcgis.com/3.8/js/esri/css/esri.css">
  //link to JQuery library
  <script src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.0/jquery.min.js"></script>
  <script src="getCensusData.js"></script>//link to Code 3 example
  <style> html, body, #map {height: 100%; width: 100%; margin: 0; padding: 0;} </style>
  <script src="http://js.arcgis.com/3.8/"></script>//link to ESRI API
  <script>
    require ([
      "dojo/parser", "dojo/dom-construct", "dojo/json", "dojo/_base/array", "dojo/_base/connect", "dojo/number",
      "esri/map", "esri/layers/FeatureLayer", "esri/geometry/Extent", "esri/InfoTemplate",
      "esri/renderers/SimpleRenderer", "dojo/_base/Color", "esri/symbols/SimpleFillSymbol",
      "esri/symbols/SimpleLineSymbol", "dojo/domReady!"
    ], function (
      parser, domConstruct, JSON, arr, conn, number, Map, FeatureLayer, Extent, InfoTemplate, SimpleRenderer,
      Color, SimpleFillSymbol, SimpleLineSymbol
    )
  
```

Example 5: A mashup using both the Census and Esri JavaScript APIs to map census data. Continued next page.

```

) {var bounds=new Extent ({//Set spatial extent of map and coordinate system
  "xmin" : -2473966, "ymin" : -2231235, "xmax" : 2421565, "ymax" : 1922548 ,
  "spatialReference":{"wkid" : 102004}
});
var map=new Map("map", {extent : bounds, center : [-96, 41]});
//Define States Layer from the Census TIGERweb rest Services
var urlSTE=
  "http://tigerweb.geo.census.gov/arcgis/rest/services/Census2010/tigerWMS_Census2010/MapServer/88";
var templateSTE=new esri.InfoTemplate ("${*}");
var CensusSTE=new FeatureLayer(urlSTE, {
  mode : FeatureLayer.MODE_ONDEMAND, outFields:["*"], infoTemplate : templateSTE ,
});
map.addLayer (CensusSTE); //add layers to map
$('#submit').click (function(){
  getCensusData ();//get and format census data with Example 3 example
  alert ("Getting census data...");
  DemMin=new Number; DemMax=new Number;
  //loop through each census object and set min and max value
  arr.forEach(arrayCensus, function(C){
    if(C.P0050010 < DemMin){DemMin=C.P0050010 ;}
    if(C.P0050010 > DemMax){DemMax=C.P0050010 ;}
    //for each census object,loop through spatial object and join data, add attribute called CenData
    //CenData represents the Census Variable P0050010 (Total Hispanic Population)
    arr.forEach(CensusSTE.graphics, function(G){
      if(G.attributes.GEOID==C.state){
        G.attributes.CenData=C.P0050010 ;
        return false ;
      }
    });
  });
  var renderer=new SimpleRenderer
  (new SimpleFillSymbol().setOutline(new SimpleLineSymbol().setWidth (0.5)));
  renderer.setColorInfo ({//set classification values and symbology color information
    field : "CenData", minDataValue : DemMin, maxDataValue : DemMax,
    colors : [new Color ([253, 245, 230]), new Color ([139, 126, 102])]
  });
  CensusSTE.setRenderer(renderer);
  CensusSTE.redraw ();
});
});
</script>
</head>
<body> <div id="map"><button id="submit"> Map Census Data </button> </div> </body>
</html>

```

Example 5, continued.

attribute at the bottom named CenData in the popup window. The feature symbology is a gradient between two colors assigned to the minimum and maximum CenData attribute values (outlined in the code).

MAPPING WITH THE D3 JAVASCRIPT LIBRARY

The D3/Census Bureau mashup is essentially the same as above, with two exceptions: the D3 JavaScript library requires data defined in either the GeoJSON or TopoJSON formats, and D3 has its own AJAX functionality—the jQuery library is not needed. Example 6 outlines the

process to render the map in Figure 4. To demonstrate dynamic field calculation with Census data, percent values are calculated on-the-fly.

D3 has more robust visualization capabilities because of its unique data handling functions. For example, the quantize classification utilized in Example 6 removes outliers in the data range that could skew the color values of the map symbology. Furthermore, we are using ColorBrewer (www.colorbrewer.org)’s 9-class “Reds” scheme, which is integrated into D3.

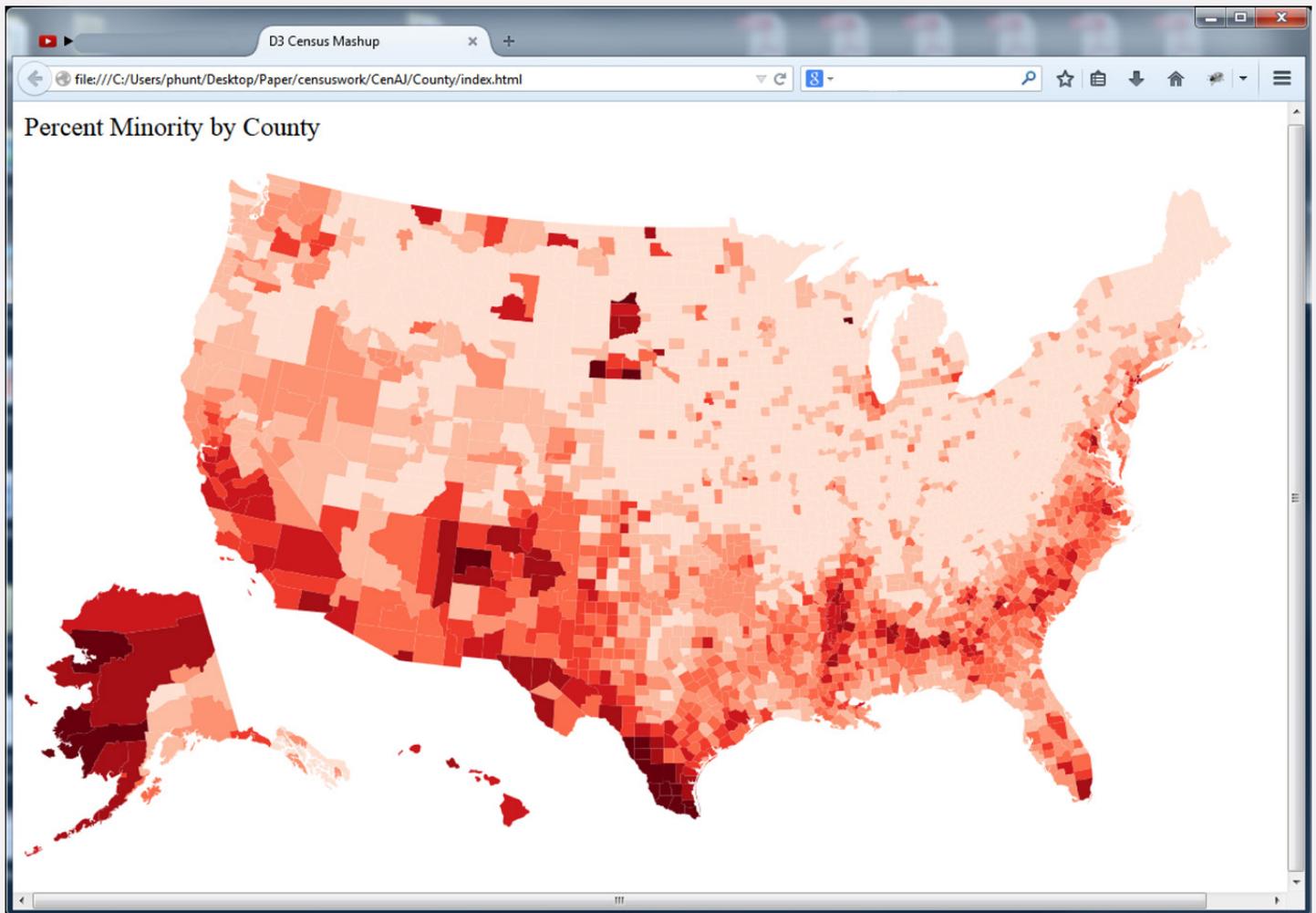


Figure 4: A mashup map produced with the Census Bureau API and the D3 JavaScript library displaying percent-minority population.

```

<html lang = "en" >
  <head>
    <title> D3 Census Mashup </title>
    <script type = "text/javascript" src = "d3/d3.v3.js" ></script> //link to D3 library
    //link to FormatCounty(), similar to Example 3
    <script type = "text/javascript" src = "d3/Paul_minMaster.js" ></script>
  </head>
  <body> <big> Percent Minority by County
  <script type = "text/javascript">
    var w=1000; var h=700 ;
    var projection=d3.geo.albersUsa().translate([w/2, h/2]) //Define map projection
    var path=d3.geo.path().projection(projection);
    var color=d3.scale.quantize() //Define quantize scale to sort data values into classes of color
    //Colors from ColorBrewer.js, included in the D3 download
    .range(["#fff5f0", "#fee0d2", "#fcbba1", "#fc9272", "#fb6a4a",
    "#ef3b2c", "#cb181d", "#a50f15", "#67000d"]);
    var svg=d3.select("body").append("svg").attr("width", w).attr("height", h); //Create SVG element
    //ajax the Census data - P0010001=Population, P0050003=Non Hispanic White Population
    d3.json("http://api.census.gov/data/2010/sf1?key=[userkey]&get=P0010001,P0050003,NAME&for=county:* ",
    function(data) {
      //call function to format data into array of objects like in Example 3

```

Example 6: Code to combine Census API with the D3 JavaScript library. Continued next page.

```

var arrayCensus=FormatCounty(data);
//loop through census data; create and calculate new normalized variable PerMin(percent minority)
for(var i =0 ; i < arrayCensus.length ; i++){
  arrayCensus[i].PerMin=1 -(arrayCensus[i].P0050003 / arrayCensus[i].P0010001);
}
//loop through dataset and calculate input domain min/max value for Quantize color scale
var min=parseFloat(arrayCensus[0].PerMin); var max=parseFloat(arrayCensus[0].PerMin);
for(var i=0 ; i < arrayCensus.length ; i ++ ) {
  pop=parseFloat(arrayCensus[i]. PerMin);
  if(pop < min) min=pop ;} if(pop > max){max=pop ;}
}
color.domain([min , max]); //Set the range of values for Quantize classification
d3.json("us-counties.json", function(json) { //Load in GeoJSON data
  //Merge census data and GeoJSON; Loop through once for each data value
  for(var i=0 ; i < data.length ; i ++ ) {
    var dataCnty=arrayCensus[i].GEOID;
    var dataValue=parseFloat(arrayCensus[i].PerMin); //Grab data value, and convert from string to float
    //for each census data value loop through counties, find the corresponding county inside the GeoJSON
    for(var j=0; j < json.features.length ; j ++ ) {
      var jsonCnty=json.features[j].id;
      if(dataCnty == jsonCnty) {
        json.features[j].properties.value=dataValue ; //Copy the data value into the GeoJSON
        break ;
      } } }
    svg.selectAll("path") //Bind data and SVG, create one path per GeoJSON feature
    .data(json.features)
    .enter()
    .append("path")
    .attr("d" , path)
    .style("fill", function(d) {
      //Use defined 'color()' Quantize scale to symbolize the fill based on census data value
      var value=d.properties.value ;
      if(value){return color(value);}
      else{return "#ccc" ;}
    }); }); });
</script>
</body>
</html>

```

Example 6, continued.

SUMMARY

A VARIETY OF APIs and JavaScript can be used to access Census data for on-the-fly mapping. With some additional work, an interface that allows the user to specify Census attributes, color scheme, classification, and spatial units (i.e., state, county, tract, etc.) could be implemented. This would unlock the potential of the Census data for spatial analysis.

Furthermore, the capabilities of D3 could be incorporated into the Esri JavaScript API to enhance how the data is visualized. There are current examples of D3 being used with Esri API on the Esri developers website (Esri 2014b). Using the technology described in this article, a simple web mapping interface could permit users to more easily explore the vast amount of data available from the Census.

REFERENCES

Batty, Michael, Andrew Hudson-Smith, Richard Milton, and Andrew Crooks. 2010. "Map Mashups, Web 2.0 and the GIS Revolution." *Annals of GIS* 16(1): 1–13. doi:[10.1080/19475681003700831](https://doi.org/10.1080/19475681003700831).

Cuesta, Hector. 2013. *Practical Data Analysis*. Birmingham, UK: Packt Publishing.

Esri. 2014a. *ArcGIS API for JavaScript*. Accessed January 21. <https://developers.arcgis.com/javascript/jsapi/>.

- . 2014b. *SVG and CSS using D3*. Accessed April 29. https://developers.arcgis.com/javascript/jssamples/styling_svg_quantize.html.
- Fielding, Roy Thomas. 2000. "Architectural Styles and the Design of Network-based Software Architectures." PhD diss., University of California, Irvine.
- Peterson, Michael P. 2014. *Mapping in the Cloud*. New York: Guilford Publications.
- Powell, Thomas A. 2008. *Ajax: the complete reference*. New York: McGraw-Hill.
- Raasch, Jon. 2013. *JavaScript Programming Pushing the Limits: Advanced Application Development with JavaScript & HTML5*. Chichester, England: Wiley.
- US Census Bureau. 2014. *Access Data with the Census API*. Accessed January 21. <http://www.census.gov/developers/>.

Experimental Terrain Representation

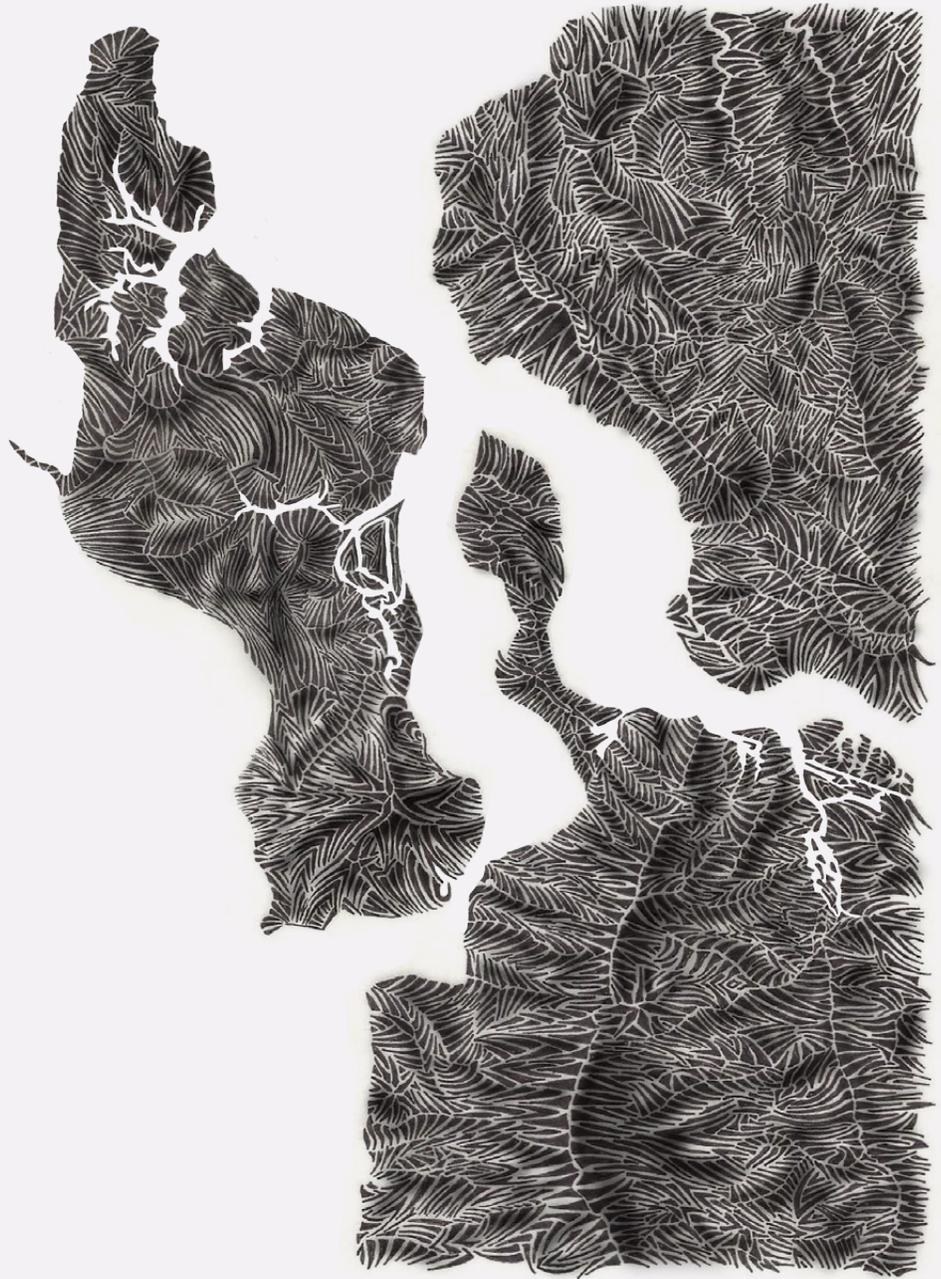
Kelly Abplanalp
 University of Wisconsin–Madison
 kabplanalp@wisc.edu

I AM ENTHRALLED by the way maps draw readers in and shape their opinions and thoughts towards the land. The way in which the cartographer depicts terrain plays a large role in how the subject is interpreted, and I wanted to explore showing mountains in a way that would instill a sense of beauty, curiosity, and a basic understanding of the land in the reader.

I began with pen, paper, and the ambition to replicate a standard hachuring technique. However, these attempts gradually gave rise to the distinctive pattern that serves as a base for my terrain maps. Elongated triangles and V shapes flowing with the land characterize this pattern. For example, a common arrangement is to place the short faces of the triangles against each other along a ridge top while their tails run down into the valley. The end result of the pattern work is intriguing to look at, but it is impossible to gain a sense of depth from such a uniformly black and white image. I added shading using Photoshop to create the illusion of elevation.

The overall result is a simple hillshade embellished by the ink

Long Island, Washington

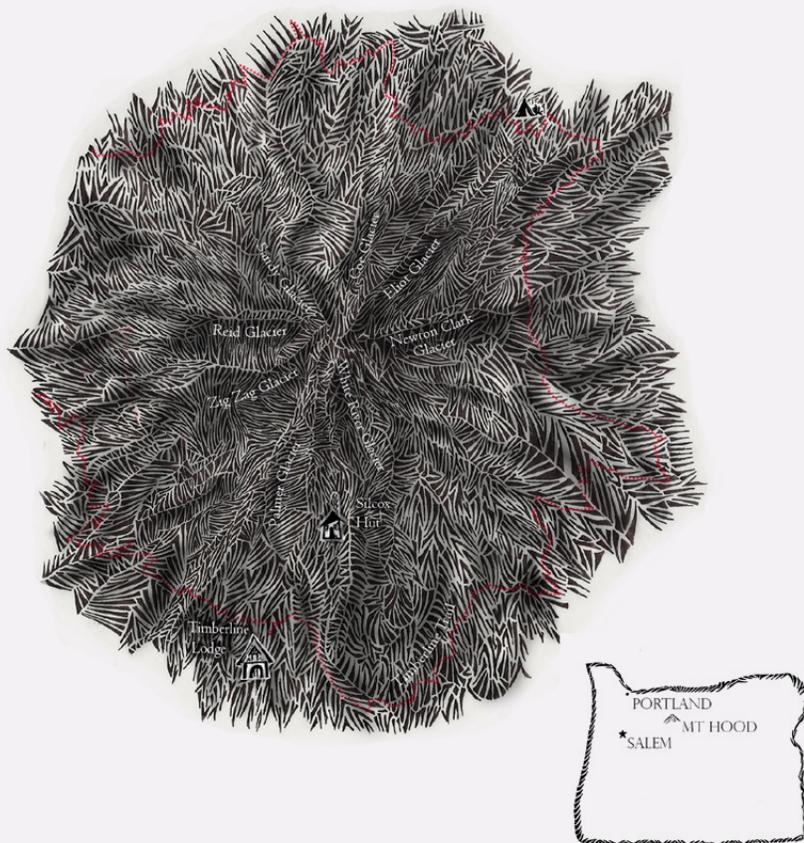


Long Island off the coast of Washington state.



© by the author(s). This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

MT. HOOD, OREGON



This map of Mt. Hood incorporates labels and hand drawn symbols into the terrain depiction.

pattern. I show the land in an artistic fashion, with minimal information, for the sake of the readers' experience. I like the style to stand as a whole without labels disrupting the flow of the design. Although I have experimented with labels and additional

features, I prefer an unlabeled landscape to let the land speak for itself. Mountains and their connections to people are powerful, and I hope that my maps allow people to explore familiar terrain in a new way.

Kelly Abplanalp is a cartography/GIS student at the University of Wisconsin-Madison. She can be contacted at kabplanalp@wisc.edu.



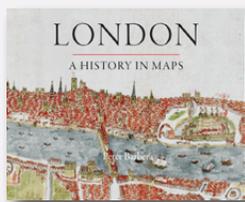
To begin the process, I place tracing paper over a topographic map and use the topo lines to guide my drawing.



A close up of Silcox Hut, a small lodge built by the Works Progress Administration in 1939.

Visual Fields focuses on the appreciation of cartographic aesthetics and design, featuring examples of inspirational, beautiful, and intriguing work. Suggestions of works that will help enhance the appreciation and understanding of the cartographic arts are welcomed, and should be directed to the section editor, Laura McCormick: laura@xnproductions.com.

LONDON: A HISTORY IN MAPS



By Peter Barber, with notes on the engravers by Laurence Worms, edited by Roger Cline and Ann Saunders.

The London Topographical Society in association with the British Library, 2012.

380 pages, 97 color maps, 48 views, 62 plans, 9 paintings, and 10 photos, 43 miscellaneous (images, text, architectural drawings, cartoons, etc.). \$45.00, cloth.

ISBN 978-0-7123-5879-8

Review by: Patti Day, University of Wisconsin–Milwaukee

London: A History in Maps is the result of an exhibition about the U.K. capital city that was held at the British Library from November 2006 to March 2007. Organized by Peter Barber, Head of Maps at the British Library, the exhibition coincided with the publication of Peter Whitfield's *London: A Life in Maps*, for which, unfortunately, there was neither a catalog nor a compiled list of captions. Acknowledging this gap, the Council of the London Topographical Society undertook to publish every illustrated item, in part or whole, along with captions, resulting in this beautiful publication.

The book illustrates and explains the metamorphosis of London from small town to one of the world's greatest cities. It presents not only official plans and maps but also those produced by individuals and groups as everyday documents for public use. As noted in the introduction, the volume endeavors to provide a balanced and inclusive view, one that documents the concerns, assumptions, and prejudices of the map makers and their times, while providing a unique perspective on the geography of London and the lives of its inhabitants. As a result, the maps and panoramas "are far more than topographical records...occasionally revealing attitudes which were too crass to be put into writing" (viii).

As the title indicates, this volume is not simply a historical London atlas, but rather a documentation of the history of the city. As such, the book incorporates not only traditional maps and plans, but also novel images of items such as the city's seals, cartoons, views, paintings, architectural

drawings, photographs, letters, and the products of GIS analysis.

The volume is divided into eight sections, which proceed roughly chronologically from 50 CE to contemporary (post-2012) plans for the future, although there is considerable overlap between time periods. Sub-sections are based around significant events that punctuated the development of the city. The sections are

1. "The Walled City 50–1066;"
2. "London Reborn," covering the reconstruction of the city after the great fire of 1066;
3. "Sweet Salutarie Air: London Countryside," describing the growth of the city and its interaction with the surrounding countryside;
4. "Out of Sight: The East End and Docklands," areas which before the 1650s were "ignored on printed maps of London" (140);
5. "The Age of Improvement," which includes the Regency period;
6. "The Mean Streets of Victorian London," during which period "the city more than doubled in size and population" (218);
7. "Metroland," which documents how the area around London changed prior to Green Belt legislation in 1938; and finally
8. "Maps in Modern London," essentially after World War II.

Generally, each section concludes with vignettes about the lives of Londoners during the relevant time period. Also included are separate entries about the engravers and publishers of the maps, together with interesting ancillary observations and pertinent information. The call numbers of the relevant British Library holdings are both helpful



© by the author(s). This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

to researchers and a tantalizing indication of the British Library's cartographic riches.

The entire volume exceeds the author's expressed aspirations to present an honest and inclusive representation of London. I particularly appreciated the author's perspectives on the history of mapping, which includes analysis and documentation of the highly selective nature of maps and mapping throughout history. The analysis of the motives of cartographers and publishers in particular is outstanding, being something few other books of this type include and a worthwhile addition to any publication discussing the history of cartography.

The book incorporates numerous references to contemporaneous social conditions in the city, which helps to situate the maps and other items culturally. The inclusion of materials other than maps and plans—letters, photographs, text, architectural drawings, cartoons, and even a poem—facilitates a deeper understanding of London's changing socioeconomic conditions. The layout of two pages per item (generally) allows the author to provide a comprehensive account of each item and also permits inset views of that which is described in the text. The descriptions of the items and the accounts of social conditions are outstanding, providing much more information than the items themselves possibly could. Barber not only documents the history of the city cartographically, but also describes the evolution of land ownership and outlines changes in the processes used in mapping through the ages. He gently yet effectively illustrates the historical human cost of metropolitan life for those not of the wealthy classes, while also drawing attention to the role of the wealthy in influencing the cartography of the times. Included are “must-read” accounts of episodes such as the 1790s fashion for the “optical tricks” of telescopic views, a synopsis of the origins of the Ordnance Survey, a discussion of John Snow's cholera map, and the story of Harry Beck's iconic map of the Tube, London's underground train system.

The entire volume is exceptionally well written and is a delight to read, including, for example, the elegant description of “ancient towns and villages now absorbed into Greater London...[that] were shown like stars surrounding the central sun of London” (104). The history of land ownership and the changes in settlement patterns are well documented and make for compelling reading. For example, during the post-Reformation period, the former monastic estates were distributed into the hands of secular families,

who eventually subdivided and sold them, facilitating the development of the gentry's villas and parks. These estates survived into the nineteenth century, when most were sold and developed for urban housing, with the original landowners' family names perpetuated in street nomenclature.

Section Four vividly describes London's East End and the Docklands, which were not shown on printed maps prior to the 1650s, when maps depicted nothing beyond the Tower of London. Even though the East End played a vital role in generating the profits that allowed the West End to flourish and prosper, it wasn't until the development of the new docks in the 1790s that mapping of the area came to be considered essential. The inclusion of plans that document the tenements to be demolished illustrates the human costs borne by “the poorest classes” and the steep price they paid with the coming of the docks (162). These details vividly outline the social divides of the city between the wealthy and the poor, a concept mirrored in Section Six, which includes the first use of thematic maps to identify disease, squalor and poverty. One view from 1844 inspires the statement that “One can almost feel the dirt, dust and pollution” (230).

The advent of railways and the car changed London and its environs radically, a transformation that is well documented by the selected maps and plans. Section Seven depicts vividly the changes wrought by an increasing population, new means of transport, and new building techniques, which led to increasing social expectations between 1850 and 1945. “Maps had to change radically in appearance to be of real use in this changed environment” (282). There are wonderful descriptions of Phyllis Pearsall's first A to Z map and the Ordnance Survey's one-inch to the mile Popular Edition series. The section concludes with the destruction of the two World Wars, and the inclusion of part of an elementary school student's poignant essay about a 1915 Zeppelin raid provides another striking example of how seemingly inanimate historical events are tied inexorably to intimate human impacts. A stark reminder is provided by WWII German maps showing the principal bombing targets along the Thames, particularly the Docklands. Something of which I was previously unaware is that immediately after the WWII air raids, surveyors were sent out to map the damage to individual buildings and to identify areas for post-war reconstruction. These maps furnished me with a greatly heightened understanding of the degree of destruction wrought upon London, and increased my appreciation for the personal stories of

my parents-in-law about nights on rooftops watching for rockets, bombs and fires.

Section Eight discusses the impacts of Geographic Information Systems, whose “low price and relative ease of production have also had the effect of democratizing map making, rendering it easier for the disadvantaged and for minorities to present viewpoints which often question the assumptions of the majority or highlight problems which might otherwise be glossed over” (344). Following post-WWII reconstruction, materials illustrate the turbulent post-1960s era of job losses, homelessness, and the provision of social services. The housing crisis during the economic downturn of the 1980s shows a city in social decline at the same time that tourism and the financial sector are “taking off.” “Swinging London” of the 1960s is depicted by a delightfully psychedelic Carnaby Street panorama reminiscent of Beatles’ album covers (356). Plans for Docklands redevelopment and others showing the upward growth of the city use the latest techniques to illustrate how the old and new can be combined in the present and future.

This work effectively accomplishes the author’s efforts to provide a truthful view of the history of this historic city. The general index and the separate index of people are both thorough and valuable for identifying not only cartographers and publishers but also people depicted in views, and for locating individual buildings, such as hospitals, and linking such diverse items as street names, sports (cricket), individual docks, and museums. There also is a helpful select reading list on historic maps of London, divided into bibliographies, monographs, and websites.

I did find that showing the majority of maps on only one page, with insets on the opposite page, sometimes resulted in me missing details of maps that might have warranted a two-page spread. There are a few examples where the original map was simply too large to condense onto a single page, and these were handled extremely well without any part of the map being “lost” to the binding. Such care typifies the overall quality of the work, and one wishes that all such volumes showed such attention to detail.

Having visited London but never having lived there, I sometimes needed a modern street map to help me place and appreciate the early maps. Fortunately, my spouse grew up in suburban London and assisted me in reconciling the historical and contemporary maps, but for those lacking a detailed geographical knowledge of the area, a simple modern street map of the central part of the city somewhere in the volume would have been helpful.

Overall, I highly recommend this book to all public and academic libraries and to anyone interested in the history of London. The price, the vast amount of information included, the writing and the range of items included makes this an extremely valuable aid to anyone seeking to understand this great city. We should all be grateful to Peter Barber, Laurence Worms, Roger Cline, Anne Saunders, the British Library, and Council of the London Topographical Society for this outstanding contribution. I sincerely wish I had seen the exhibit itself, but this volume certainly does it credit.

THE WORLD AT THEIR FINGERTIPS: EIGHTEENTH-CENTURY BRITISH TWO-SHEET DOUBLE-HEMISPHERE WORLD MAPS



By Geoff Armitage and Ashley Baynton-Williams.

British Library, 2012.

262 pages, 155 color plates.
\$65.00, cloth.

ISBN 978-0-7123-5877-4

Review by: Ian Muehlenhaus, James Madison University

The World at Their Fingertips is a book with a two-pronged mission: first, to definitively establish British two-sheet, double-hemisphere world maps as a distinct cartographic

genre, and second, to create as complete a catalog of such maps as possible, while concurrently exploring their history. Using the extensive collection of these maps housed at the British Library, the authors looked at “virtually all examples” of these maps that still exist in Britain in an attempt to establish a double-hemisphere world map genealogy of sorts. The book is very focused on achieving its goals, and it certainly achieves what the authors set out to accomplish.

Overall, the book is best summarized by its title; it is what it portends to be. Early on, the authors argue that

double-hemisphere world maps are important within the history of cartography. Such maps rose to prominence due to two interesting developments. First, the British were experiencing a prolonged period of prosperity following the reestablishment of the monarchy in 1660. Second, the public not only had more money to spend, but they were beginning to clamor for more geographical knowledge about the larger world around them. Before this time, maps were largely only affordable for the wealthy. Essentially, an increase in wealth throughout Britain broadly, coupled with a new interest in the world, helped create a market for mass-produced global maps and atlases. Businesses sprouted up to place the world at Britons' fingertips.

The structure of the book's content is unique. There are nine chapters, including the introduction, with three additional appendices. The introduction chapter succinctly contextualizes double-hemisphere maps within the history of cartography as a whole, before segueing into the first two chapters, which review the origins of the double-hemisphere world map in seventeenth-century France and the nuances of the British map trade in the following century. The second chapter in particular is very robust, and provides a thorough summary of the history of British map production and sales between 1650 and 1820. Chapter 3 comes off as a bit of a non sequitur. It presents the "high points" of mapping during the eighteenth century, beginning with an overview of James Cook's discoveries and concluding with a two-paragraph entry on Sir Isaac Newton. This is followed by a chapter on the "low points" of cartography during the same time period, concluding with an enjoyable discussion of a painting with a small double-hemispheric map in the background. Chapter 5 discusses topics related to double-hemisphere maps such as print runs, projections, prime meridians, and more relevant asides. The sixth chapter presents a map taxonomy of sorts, breaking down double-hemisphere world maps into different types based on a variety of factors, the primary of which is who drafted the earliest version. It is a fascinating look at how shamelessly people stole other people's maps during this era. Chapter 7 provides an incredibly detailed catalog of double-hemisphere world maps broken down by 24 mapmakers. This results in well over 24 maps being shown, it must be noted, as the authors include numerous map versions from the same cartographers. The history of each mapmaker is briefly reviewed here as well. This 113-page chapter acts as a map catalog. It is then followed with a two-page concluding chapter.

This book delivers what it says it will: an incredibly detailed, as holistic as possible, index and analysis of British double-hemispheric maps. There can be no critique of how well it fulfills its goals. If this is what you are looking for in a book, look no further! The map reproductions are large, full-color, and semi-gloss. The book feels as though it has more map illustrations than text, which is a good thing. To facilitate the subject matter, the book was published with the binding on the pages' short-end, allowing the double-hemispheric maps to fill the page layout. The production quality is very good. All of the maps are clearly reproduced. The writing is excellent.

On the other hand, like any book, this one suffers from several shortcomings. By attempting to achieve two goals—review the history of *and* catalog these maps—some strange things happen to the structure of the book. For example, the chapters vary drastically in style, composition, and length, making the book difficult to read through from start to finish. Some of the chapters, particularly the first ones, read like journal articles on the history of cartography. Others are structured more like concise, academic encyclopedia entries. For example, Chapter 4 ("Low points in world mapping...") does not have an introductory paragraph. It just begins with the heading "California" and dives into a description about how cartographers really missed the boat on the fact that California was not an island. One concern about both Chapter 3 and 4 is that at times there is little connection made to double-hemispheric maps. The average chapter length is approximately ten pages. Chapter 2, on the other hand, is 113 pages long. The inconsistencies in chapter length make for an awkward reading experience. After a brief introduction, Chapter 7 essentially starts listing the creators of double-hemispheric maps and providing image after image of them. This is the catalog part of the book. It does not feel like a chapter; it feels like a file catalog. This "chapter" might have worked better as a separate section of the book following the other chapters. In sum, this book is probably best enjoyed piece-by-piece, chapter-by-chapter.

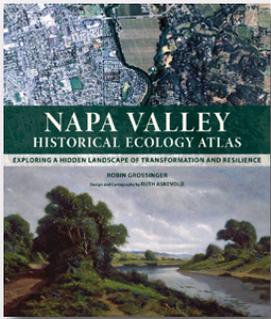
Though it is not the most readable book, this book stands as a substantial contribution when it comes to cataloging the history of cartography. It is the ultimate data source on double-hemispheric maps! It could also act as an excellent visual resource for those studying or teaching about the history of cartography during the magnificent and important periods of Dutch and British world hegemonies. Beyond academia, I can see this volume working as a coffee-table

book, as well. The authors need to be commended on their thoroughness, beautiful attention to detail, and the passion they display toward the subject matter.

On the other hand, this book is certain to have a limited market. For armchair map lovers, this book's focus on double-hemispheric maps is overkill. The images all begin to look the same. In fact, some of the images do show the same maps, just as different print versions with

slightly varied coloration. From an academic standpoint, aside from the first several chapters' historical reviews, there is a dearth of new knowledge to be gleaned. The book is absolutely packed full of excellent data and information about double-hemispheric maps, but it never quite packages this data into takeaway nuggets of knowledge. And to be fair, the book never says it will do this. However, acting primarily as a detailed repository does, unfortunately, limit its marketability both to the public and academics.

NAPA VALLEY HISTORICAL ECOLOGY ATLAS: EXPLORING A HIDDEN LANDSCAPE OF TRANSFORMATION AND RESILIENCE



By Robin Grossinger, design and cartography by Ruth Askeveld.

University of California Press, 2012.

223 pages, maps, illustrations.
\$39.95, hardcover.

ISBN 978-0-520-26910-1

Review by: Lisa Sutton

The *Napa Valley Historical Ecology Atlas* is a beautiful book, richly illustrated with maps and historical photographs of the Napa Valley. Divided into eight chapters, the first looks at the Napa Valley through time, while the next six chapters each take on an ecological component of the Valley: Oak Savannas, Wildflower Fields, Creeks, Valley Wetlands, the Napa River, and Tidal Marshlands. The book concludes with chapters on Landscape Transformation and Resilience, and Landscape Tours, offering a forward-looking view of where the Valley is going from here, and how to see and enjoy this place.

The book begins with an introduction to historical ecology and some of the inherent challenges in exploring past ecologies and landscapes. It then moves on to discuss how the Napa Valley is situated within a geographical, geological, and climatic context. Several maps and diagrams are provided to illustrate the geology, drainage, and natural history of the region. A map reconstructing the habitats and environments that would have been found in the Napa Valley in the early 1800s is provided, as well as an interesting explanation of how the authors used the source data that they had to create this map. A timeline of the data sources used in the book, ranging from Mexican land-grant sketches and diaries from the early 1800s to modern

aerial photography and soil surveys, provides a nice overview of the range of data that informed this synthesis.

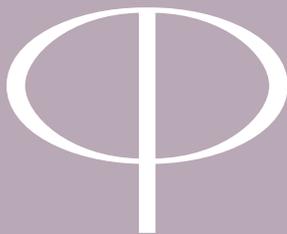
Each of the sections on an individual aspect of the ecology of the Valley discusses its particular characteristics and spatial patterns, as well as its cultural significance. A discussion of the loss of oak savannas and historical evidence for where they would have been found concludes with a look at where oaks have persisted and a possible plan for “re-oaking” the Valley to include these trees within the modern landscape. The chapter on creeks examines the unique conditions of water in the Valley and human impacts over time on its many creeks, as well as the roles they continue to play today. The chapter on Valley Wetlands explores the characteristics of the several varieties of wetlands found in the Napa Valley, then looks at changes to the overall wetland landscape, including a nice map comparing the historical and contemporary extents of wetlands. The Napa River, as the central component of the Napa Valley, merits the longest chapter. Parts of the historical river are explored, as well as ways in which the river has changed or been changed, and the ecology of the river in the past and present. The tidal marshlands make up the last main section of the Valley, with a discussion of their characteristics, the dredging and reclamation they have undergone, and recent restoration efforts as well as the effects of sea level rise.

The chapter entitled “Landscape Transformation and Resilience” includes a set of maps that compare the habitat types and land uses in the early 1800s and in 2010, and which show the changes that have taken place in Napa Valley. It also discusses how the historical landscape perspective can allow people to better understand the context of the land and make more conscious choices about how

to plan land use. Finally, the section on landscape tours divides the Valley into four sections and provides a guide to characteristic locations within each, complete with maps (modern aerial photography overlaid with ecological types and modern streets) and driving directions.

The landscape tours allow the reader to actually step into the landscapes being described (assuming one lives near or is making a visit to the Napa Valley), and provide a tangible way of engaging with the stories that have been told about these areas. As an ecologist, I appreciated the list of common and scientific names of the plant and animal species found in the Napa Valley that is included at the end of the book. An extensive bibliography also provides plenty of supplemental resources to readers who wish to explore any aspect of the book more deeply.

This is a well-researched and well-documented book, which provides a loving portrait of an area the author clearly knows well. The combination of an abundance of historical illustrations and diagrams to illustrate ecological and geographical processes works well to create a visually appealing and accessible book. The author's focus on ways that knowledge of the historical landscape patterns can aid in restoration efforts makes this a hopeful and forward-looking story, rather than simply a story of lost landscapes. Even those who are not familiar with the Napa Valley will find themselves drawn in to the fascinating story told here. Both the casual reader and the more highly trained ecologist or geographer will find this book interesting and engaging.



Cartographic Perspectives (CP) publishes original articles demonstrating creative and rigorous research in cartography and geographic visualization under open-source licensing. Papers undergo double-blind peer review; those accepted for publication must meet the highest standards of scholarship, address important research problems and issues, and appeal to a diverse audience.

Articles should be submitted online, in OpenOffice, Microsoft Word, or RTF file format. Each manuscript is reviewed by the editor, one or more members of the editorial board, and at least one external reviewer. By uploading to the CP website, authors agree to not submit the manuscript elsewhere until the CP editor has reached a decision. Any submitted manuscript must not duplicate substantial portions of previously published material.

GENERAL GUIDELINES

Content should be submitted online via the *Cartographic Perspectives* website, cartographicperspectives.org.

OPINION/RESPONSE PIECES: CP welcomes topical responses to previously published articles. The length of such pieces may vary; however, we suggest 2,000 words or less as an informal guide.

ILLUSTRATIONS: Maps, graphs, and photos should convey ideas efficiently and tastefully. Graphics should be legible, clean, and clearly referenced by call-outs in the text. Sound principles of design should be employed in the construction of graphic materials, and the results should be visually interesting and attractive.

- All graphics must be in digital form, either digitally generated or scanned. Preferred formats are .tif, .ai, .eps, .jpg, or press-ready .pdf.
- Images must not be embedded in the manuscript, but should instead be included as separate files.
- Color images should be submitted in CMYK mode where possible.
- Images in raster format must, at minimum, be 600px wide; at least 1000px is strongly preferred. Images will be printed at 300 ppi.
- Where possible, graphics should have a transparent, rather than a white, background.
- Digital art files should be cropped to remove non-printing borders (such as unnecessary white space around an image).
- The editor reserves the right to make minor adjustments to illustrations.
- Authors are responsible for ensuring that they have permission to use all illustrations.
- Image orientation should be the same as intended for print.

- For vector files, fonts should be embedded or converted to outlines.
- Type sizes below 6 point should be avoided.
- Captions should not be part of the illustration. Instead, please supply captions within the text of the article.

For questions on specific guidelines for graphics, please contact Assistant Editor Daniel Huffman for more information: daniel.p.huffman@gmail.com.

PERMISSIONS: If a manuscript incorporates a substantial amount of previously published material, the author is obliged to obtain written permission from the holder of the copyright and to bear all costs for the right to use copyrighted materials.

LICENSE: Articles submitted to CP will be distributed under the Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International license. For a description of the terms of this license, please see: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

PEER-REVIEWED ARTICLES

TITLE: The title serves as the author's invitation to a diverse audience. It should be chosen wisely. The title section should include the full name(s) of the author(s) and academic or other professional affiliation(s).

ABSTRACT: An abstract of 250 words or less should summarize the purpose, methods, and major findings of the paper.

KEYWORDS: Five to ten keywords should be listed at the end of the abstract.

REFERENCES: References should be cited parenthetically in the text, following the author-date system found in *The Chicago Manual of Style*, 16th ed. (chicagomanualofstyle.org). When making a direct quote, include the page number. Examples: (Doe 2001) and (Doe 2001, 38).

Books: Invert the first author's name (last name, first initial or name, and middle initial). Middle initials should be given wherever known. For books with multiple authors, authors' names are listed in the order in which they appear on the title page, with the last author's name preceded by a comma and *and*. Note: With more than ten authors, invert first author's name and follow it with a comma and the words *et al.* without italics in the reference list.

The general format is: Name of author(s). Year. *Title in Italics*. City of Publication: Publisher Name.

MacEachren, A. M. 1995. *How Maps Work*. New York: Guilford Press.

Robinson, A. H., J. L. Morrison, P. C. Muehrcke, A. J. Kimerling, and S. C. Guphill. 1995. *Elements of Cartography, 6th Edition*. New York: John Wiley & Sons.

Articles in Periodicals: Author's or authors' names as in *Books*, above. Year. "Title of Article." *Title of Periodical*, volume number, page numbers [follow punctuation and spacing shown in the following example].

Peterson, M. 2008. "Choropleth Google Maps." *Cartographic Perspectives* 60:80–83.

Articles in edited volumes: Name of author(s). Year. "Title of Article." In *Title of Edited Volume*, edited by [Editor's or Editors' names, not inverted], page numbers. City of Publication: Publisher's Name.

Danzer, Gerald. 1990. "Bird's-Eye Views of Towns and Cities." In *From Sea Charts to Satellite Images: Interpreting North American History through Maps*, edited by David Buisseret, 143–163. Chicago: University of Chicago Press.

Websites: Websites may be generally referenced in running text ("On its website, the Evanston Public Library Board of Trustees states...") rather than with a URL listing. For more formal citations, use the following format: Name of author(s). Year. "Title of Document." *Title of Complete Work (if relevant)*. Access date. URL.

Cartography Associates. 2009. "David Rumsey Donates 150,000 Maps to Stanford University." *David Rumsey Map Collection*. Accessed January 3, 2011. <http://www.davidrumsey.com/blog/2009/8/29/david-rumsey-donates-150-000-maps-to-stanford>.

Maps: Maps should be treated similarly to books, to the extent possible. Specific treatment may vary, however, and it is often preferable to list the map title first. Provide sufficient information to clearly identify the document.

A Plan of the City of New York and its Environs. P. Andrews, sold by A. Dury in Dukes Court, St. Martins Lane, surveyed by John Montessor, 1775.

E-mail correspondence: E-mail messages may be cited in running text ("In an e-mail message to the author on October 31, 2005, John Doe revealed...") instead of in a note or an in-text citation, and they are rarely listed in a bibliography or reference list.

Additional examples: For additional examples, please consult *The Chicago Manual of Style*, 16th ed. (chicagomanualofstyle.org).

REFERENCES LIST: The list of references should begin in a separate section, immediately after the text. Entitle the section "References" and list all references alphabetically by the author's last name, then chronologically. Provide full, unabbreviated titles of books and periodicals.

FOOTNOTES: Footnotes should be used sparingly: i.e., only when substantive enough to amplify arguments in the text. They should be addressed to a single point in the manuscript. Footnotes should be numbered sequentially in the text and will appear at the bottom of the page.

UNITS OF MEASURE: *Cartographic Perspectives* uses the International System of Units (metric). Other units should be noted in parentheses.

EQUATIONS: Equations should be numbered sequentially and parenthetically on the right-hand edge of the text. If special type styles are required, instructions should be provided in the margin adjoining the first case of usage. Authors should carefully distinguish between capital and lower-case letters, Latin and Greek characters, and letters and numerals.

TABLES: Tables should be discussed in the text and denoted by call-outs therein, but the meaning of a table should be clear without reading the text. Each table should have a descriptive title as well as informational column headings. Titles should accent the relationships or patterns presented in the table.

Cartographic Perspectives

The Journal of **naGIS**



Number 26, 2013

