Transactions in GIS

A User-centered Design for the Addition of Interactive Masking Capability within an existing Web GIS

John D. Morgan

Department of Earth and Environmental Sciences, University of West Florida

Abstract

This article presents a case study of how a user-centered design (UCD) approach was utilized during the addition of interactive masking capability to an existing web-based geographic information system (Web GIS). By analyzing and discussing specific aspects of the user-developer dialog within the context of a Web GIS software development life cycle, this article presents a case study for similar systems. The results of the UCD methodology is a discussion that presents a shift from an initial design to a new design that, based on user feedback, furthers the utility and usability of interactive masking within the Web GIS.

1 Introduction

Until recently, desktop GIS software – the development of which was dominated by a small group of commercial enterprises – was primarily used by select professions and specialty groups (Tsou 2011, Goodchild 2013). With the increasing availability of web mapping application programming interfaces (APIs) and free and open-source geospatial software (termed FOSS4G), those tasked with the job of developing and designing web-based geographic information system (Web GIS) applications are faced with the question of how to (re)develop new and existing capabilities. While Web GIS often reproduces features currently available within desktop GIS, usage requirements often demand it be both easier to use and more specific in its purpose.

Interaction as a subject of study within Web GIS builds upon established Window-Icon-Menu-Pointer modes of graphical interaction, such as changing the scale (zoom), location, and the ability to pan the map interface (You et al. 2007, Manson et al. 2012). Studies that have focused on the map interface as an area of interactivity, or cartographic interaction, are mostly concerned with the communication of information from the cartographer to the map user (Crampton 2002). However, a Web GIS often provides methods of interaction beyond the map such as layer toggle checkboxes that are peripherally located to the map itself.

Usability testing has been adopted for graphical interface design purposes (human-computer interaction, or HCI) ranging from web pages to portable electronic devices (Demšar 2007). Early on, Marchionini and Komlodi (1998) suggested minimizing mouse clicks and quickly bringing relevant information to the user. More recently Eicher et al. (2010), in their discussion of how usability tenets were applied in the design of reusable GIS software components, followed Wickens et al.'s (2004) principle of minimizing information access cost and

Address for correspondence: Department of Earth and Environmental Sciences, University of West Florida, 11000 University Parkway, Pensacola, FL 32514, USA. E-mail: jdmorgan@uwf.edu

Acknowledgments: Primary acknowledgements go to the Eastern Forest Environmental Threat Assessment Center (EFETAC) team at the USDA Southern Research Station of the Forest Service for their expertise and support. The author of this article would also like to thank his colleagues at NEMAC, where he worked as an Applied Research Software Designer from 2011-2015.

doi: 10.1111/tgis.12197

suggest minimizing mouse clicks by providing tools and commands on buttons rather than on menus, and by placing these on a floating palette that is always available.

As early as the 1990s, geospatial scholars also began addressing usability interface issues in desktop GIS (Kuhn and Frank 1991, Medyckyj-Scott and Hearnshaw 1993, Tsou and Buttenfield 1998). Building on usability work in other disciplines that Web GIS designers can apply, Wickens et al. (2004) proposed 13 principles of display design to cartographic interfaces towards making the Web GIS experience most efficient for users. However, as Roth (2014) points out, a science of cartographic interaction must begin with consideration of the overall experience of cartographic interaction and not immediately focus upon the implementation and use of cartographic interfaces (Beaudouin-Lafon 2004). Therefore, considerations of utility are equally important to usability concerns if we are not to end up with highly usable cartographic interfaces that are not useful for their intended purpose (Grudin and Markus 1997, Johannessen and Hornbaek 2014).

Forest change assessment as a discipline has greatly benefited from advances in geospatial technologies. These advances range from increasingly available remotely sensed satellite data, to more accessible spatial information delivery platforms such as Web GIS (Kelly and Tuxen 2003, Van Leeuwen 2008). The science of forest change assessment using GIS is as much about locating disturbances, as it is placing those events within a specific spatio-temporal context. In order to situate a forest change event into spatio-temporal context, scientist rely on ancillary layers and the ability to filter data by specific attributes. For instance, Spruce et al. (2011) presented an assessment process for detecting forest defoliation from Moderate Resolution Imaging Spectroradiometer (MODIS) data collected from 2000 to 2006. In their study they classified MODIS-based defoliation detection products into non-forest, non-defoliated forest and defoliated forest by using the 2001 National Land Cover Database. This approach helped them to analyze a regional gypsy moth defoliation within the context of specific land classification.

The Forest Change Assessment Viewer (FCAV) is a Web GIS application launched by the USDA Forest Service in 2010, providing interactive mapping capability for the purposes of forest disturbance detection. The FCAV is used primarily by a science team at the USDA Forest Service Southern Research Station's (SRS) Eastern Forest Environmental Threat Assessment Center, but is also available to the general public via the World Wide Web. The FCAV was developed and is maintained by UNC Asheville's National Environmental Modeling and Analysis Center (NEMAC) under contract to the USDA Forest Service. Since its inception, the data software architecture for the FCAV was based on a FOSS4G software stack. All of the application's vector data are stored and served from PostgreSQL with spatial functionality provided by PostGIS. The raster data (e.g. NDVI map layers) are stored in GeoTIFF files, a public domain metadata standard that allows geographical reference data to be embedded within a Tagged Image File format. MapServer, an application which was developed at the University of Minnesota (UMN), receives requests for Web Map Service (WMS) layers, the open standard protocol for serving geographically-referenced map images over the Internet. When MapServer receives a web client WMS request, it passes them along to the file and database server, which then responds with the geographic data of interest. This software stack runs on the CentOS operating system (OS), a distribution of Linux.

The interface software architecture operates within the client web browser interface. The interface software architecture for the FCAV was originally developed using Esri's ArcGIS API for Flex. The ArcGIS API for Flex requires the Flash plug-in from Adobe Systems Inc. This plug-in requirement constitutes a major limitation, as Flex has experienced limited support on mobile devices (Boulos et al. 2010). During the 2011 and 2012 vegetative growing seasons –

when FCAV usage peaks – the developers at NEMAC received many new functionality requests from the scientists at the USDA Forest Service, in particular an interactive masking capability. While addressing these new functionality requests and looking forward to increased demand for FCAV availability on more client platforms, a concerted effort was undertaken in 2013 to redevelop the interface software of the FCAV from its existing plug-in software architecture to that of a plug-in-free architecture that relies on a suite of hypertext markup language (or HTML5)–related technologies.

HTML5, or the fifth revision of HTML, represents a group of web browser technologies with many goals focused on increasing usability and platform independence, namely by reducing the necessity of plug-ins (Taivalsaari and Mikkonen 2011, Sarris 2012). OpenLayers, an open source mapping API written for JavaScript (the de facto standard scripting language for HTML5), was selected for the interface software architecture in the newest version of FCAV. This plug-in–free software platform has allowed for wider operation of the FCAV across multiple platforms, including standard consumer mobile devices. This wider adoption is central to one of the goals of the FCAV team which is wider user adoption.

Unique to the FCAV is data computed in collaboration with scientists at NASA's Stennis Space Center containing Normalized Difference Vegetation Index (NDVI) values. NDVI is a land cover inventory measure used to monitor change in land cover condition, namely "greenness," over short periods, such as a growing season (Loveland et al. 1991). By proxy, changes in NDVI values can be used to detect forest disturbance by events such as wildfires, wind storms, insects, diseases, and human-induced disturbances These satellite-derived raster NDVI maps are then used to provide ongoing and historical records of forest health, supporting a science of forest disturbance detection beyond the scope of this article (see Mills et al. 2013). To achieve the second notable purpose, the FCAV has the capability to save or share a given map state, including the map zoom level, active map layers, and opacity status. This share map functionality helps the Forest Service achieve a goal of efficient forest planning and management across jurisdictions (USDA Forest Service 2012).

In order to better detect forest health anomalies, in early 2013 the Forest Service team requested that NEMAC develop into the FCAV interface software the ability to mask the NDVI maps by other ancillary layers (e.g. land cover). The end result of the masking functionality is an operator-based primitive of filtering synonymous with Crampton's (2002) description of an extract/suppress action, where a user can act on a cartographic interface to show a subset of features or enhance cartographic representation. Within the broader field of digital cartography, this capability is also referred to as masking (Wade and Sommer 2006). To assist in their forest disturbance detection procedures, the science team at the Forest Service requested that the ability to mask NDVI values by ancillary layers (e.g. tree species classifications, such as conifer and deciduous) be added to the FCAV.

Shortly after the interactive mask capability request, NEMAC's FCAV development team discovered that MapServer did indeed have the ability to mask WMS layers "on the fly" at the server level. Therefore, the ability to mask one layer by another within the current data software architecture was feasible. Of note is that the server-side implementation of masking within MapServer results in a mask being treated as another instance of a WMS layer within the data software architecture. (For more information on how masking is implemented within the UMN MapServer, see the online documentation at http://mapserver.org/development/rfc/ms-rfc-79.html.) However, the challenge of how to best design masking within the existing Web GIS interface became more nuanced as a UCD approach was followed. This approach revealed both usability and utility challenges and resulted in adopted solutions.

2 Methodology

The design, prototyping, and development of interactive masking capability within the OpenLayers version of FCAV took place from mid-2013 to early 2014. During this time, four evaluators with expertise in NDVI-based forest disturbance detection via FCAV (a Web GIS) were provided with multiple prototype versions and solicited for feedback as to design usability and utility. These four evaluators were senior level scientists with domain expertise in forest change detection and were the primary users of FCAV at the time of this project.

2.1 Selecting a UCD Method

There are a number of usability evaluation methods available that can be used for identifying interface problems during the design process. These methods first began to take shape in the late 1980s, informed by Norman (1988) who began to look at how product design helps and/or hinders communication between object and user. Later work such as that of Polson et al. (1992) and Nielson (1999) had developed usability evaluation methods specific to how computer interfaces (HCI) could best be designed. Traditional usability testing then has become a method to test a product design and judge if it works as expected. Unique to this method is that it requires a representative sample of users who can attempt to complete a typical task while evaluators watch, listen and/or take notes. Another approach to usability inspection is called heuristic evaluation and was developed by Mack and Nielson (1994) as a way of identifying problems specifically associated with the design of user interfaces. During the iterative design process evaluators examine the interface and judge its compliance with recognized usability principles (known as the "heuristics"). And while heuristic evaluation does not require a sample population of users, it does require very clear metrics allowing for the measure of any deviation from the heuristics during testing. During the design of interactive mask capability we did not have a representative sample of users or a codified set of heuristics with which to inspect the usability of alternative designs.

A third approach to usability inspection is focused on how the completion of the task, at the user interface, is accomplished. The more general of the task-oriented methods is called task analysis and can be defined as the study of what an operator (or team of operators) is required to do in terms of actions and/or cognitive processes (Kirwan and Ainsworth 1992). Task analysis is very similar to another method called cognitive walkthrough because it is a usability inspection method for simulating a user's cognitive processes as he or she interacts with an interface in an effort to accomplish a specific task (Polson et al. 1992). However, during the cognitive walkthrough method of usability inspection, the focus of inspection is more on user goals and their interactions with the system (Kirwan and Ainsworth 1992, Jordan 1998).

Cognitive walkthrough, then, is focused on uncovering the complexity of the task and therefore requires that the designer provide evaluators with a copy of the interface, or a prototype (Richards and Egenhofer 1995, Love 2005). In this sense, the requirements of cognitive walkthrough are similar to those of the software design methodology called rapid prototyping (see Preece 2002), where a prototype can be a partial representation to demonstrate main features and test if the given HCI architecture is working. A secondary outcome of cognitive walkthrough is feedback as to the usefulness, or the way in which an interface could be used to create actionable knowledge (Bacastow and Bellafiore 2009).

During the design of interactive mask capability, the FCAV development team provided multiple prototypes to the evaluation team for their review. The evaluation team were experts on the

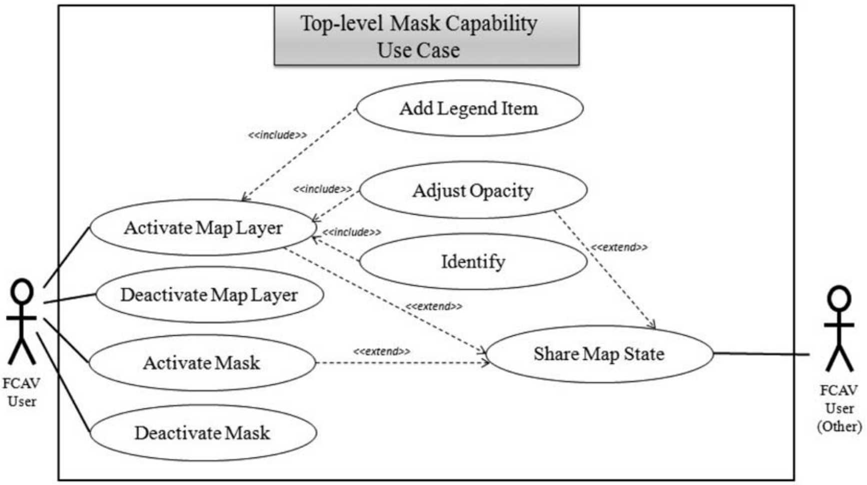


Figure 1 Top-level mask capability use case

existing task utilized to detect forest disturbance via the existing version of FCAV. For this reason, the evaluation team was able to consider the masking functionality within the context of forest disturbance detection. Also present throughout this phase, was an awareness of the primary goal for the FCAV of having the ability to share of disturbance information with other scientists and the public, across jurisdictions. Of relevance to this goal of wider adoption, was that any increased complexity of the interface needed to be balanced with usability and utility concerns. The approach taken by the FCAV development/evaluation team, during the design of interactive masking capability, fell somewhere between the task analysis and the cognitive walkthrough methods of usability inspection. It was not a formal cognitive walkthrough, but rather a hybrid approach combining elements from both task analysis and the cogitative walkthrough.

2.2 Identifying the Task

Through a dialog between the evaluator's developers, an understanding of the functional requirements of masking was discovered. Figure 1 utilizes a Unified Modeling Language (UML) use case diagram to illustrate one of the primary use cases for the FCAV, which is the ability to save or share a particular state of a map. After an instance of the FCAV loads, the user may perform a variety of tasks (e.g. zoom, pan, and load ancillary map layers). For the evaluators' purposes, these tasks are all performed in an effort to determine potential locations of forest disturbance. Once the location(s) of a potential forest disturbance is found, the map state of the FCAV is saved and passed along to the next evaluator for review.

By being able to filter (or mask) these layers, an FCAV user can more easily visually explore the relationship between NDVI and ancillary layers within the map interface. Beyond the initial task of activating a mask and sharing, subsequent functionality of the FCAV were considered in the evaluation. Figure 1 shows a top-level use case for the subsequent task related to interactive masking capability within the FCAV. This figure provides insight into the initiating tasks, subsequent tasks, and functionality expected when working with interactive mask capability within the Web GIS. For example, the action of activating a map layer is followed by the action of adding a corresponding legend, and is illustrated with the *«include»* relationship.

Also, you see by the *«extend»* relationship that the *Share Map State* use case allows one to "share" the map state with another FCAV user.

3 Results

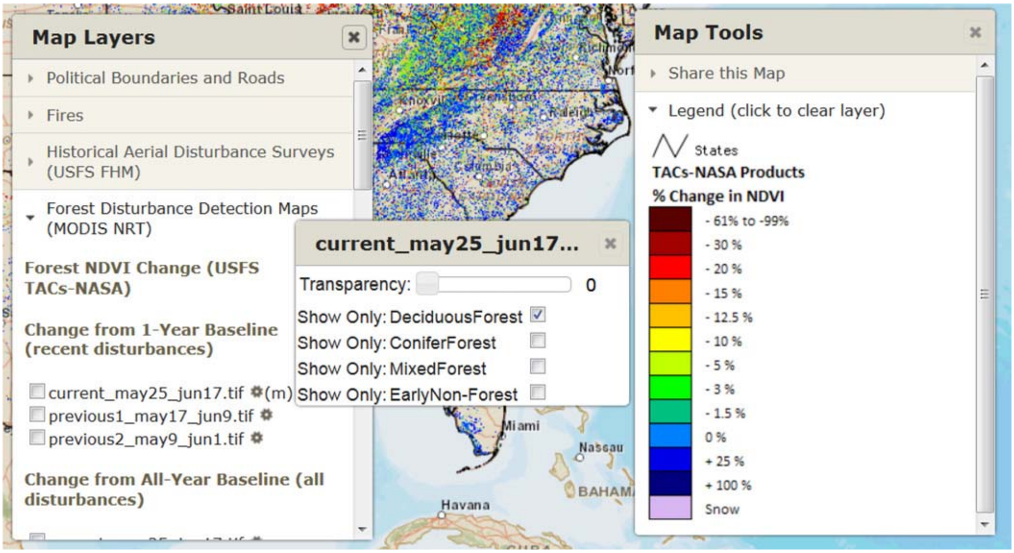
Over the course of the UCD process the evaluator's feedback to the development team resulted in an evolution of the interactive masking capability from a less usable to a more usable design. During each iteration of interface design the evaluators followed a series of tasks that allowed them to assess the usability of a particular design. As the four FCAV evaluators progressed through their task testing, they continued checking both the resulting state of the Web GIS and the subsequent expected functionality. For instance, once a map layer is activated (*Activate Map Layer* use case), the ability to get feature information about that layer is available (*Identify* use case). Feedback from evaluators to the FCAV development team occurred through electronic mail, phone and in-person meetings. Between the dates of March 2013 and February 2014, six in-person meetings took place between the FCAV development team and the evaluators. The feedback received by the FCAV development team from the evaluators during these meetings was integral in guiding the iterative design of the interactive masking functionality with the FCAV.

During the initial meetings the Forest Service team requested that interactive masking capability be added to the FCAV, but they did not specify, nor did they in any consensus realize, what this new functionality should look like within the Web GIS interface. In order to reveal user preferences with regard to the interface design of interactive masking capability within the FCAV, an iterative UCD approach was utilized (see Section 2.2). While many minor design changes were requested and discovered during the iterative task testing, there was one major shift in architecture that is highlighted in this article. This shift in architecture took place during the latter of the in-person meetings between the design team and the Forest Service team (the evaluation team).

The initial design(s) of interactive masking capability were built on a parent/child relationship that exists between the mask and its "parent" layer (more fully described below) at the server data software architecture. Since the requesting Forest Service team had not specified how to implement masking within the FCAV interface, a design that built on the object relationships at the data software architecture level took place. In this sense, a mask was treated as any other layer in the system, except that it could only be activated by its "parent" layer. This design, which we will call *layer-level masking*, is shown in Figure 2 as a screen shot of the early interactive masking capability design within the FCAV.

With layer-level masking, a given child mask is activated by clicking on its parent layer's property dialogue button. With this approach a *one-to-one* relationship existed between the layer and the mask. The parent layer's property dialogue button is illustrated as a standard gear symbol next to the layer name. The mask dialogue screen shown in Figure 2 is for the parent layer named *current_may25_june17.tif*. Further, a status symbol, *(m)*, was placed next to the parent layer, indicating that a mask was active for that given layer. Specifically, in Figure 3 the conifer mask is active for the *current_may25_june17.tif* layer. After working with the layer-level masking prototype for a few weeks, the evaluation team provided feedback that this method of working with masks was cumbersome to their process.

Based on further feedback from the evaluation team, the interactive masking prototype was then redesigned into what we will call *global masking functionality* (see Figure 4). With the global masking design, the mask functionality at the interface was detached from the layer-level controls and moved into a separate menu. Put simply, it was requested that once a mask



Map Layers ✖

- Political Boundaries and Roads
- Fires
- Historical Aerial Disturbance Surveys (USFS FHM)
- Forest Disturbance Detection Maps (MODIS NRT)

Forest NDVI Change (USFS TACs-NASA)

Change from 1-Year Baseline (recent disturbances)

□ current_may25_jun17.tif ⚙(m)
□ previous1_may17_jun9.tif ⚙
□ previous2_may9_jun1.tif ⚙

Change from All-Year Baseline (all disturbances)

current_may25_jun17... ✖

Transparency: 0

Show Only: DeciduousForest ☑
Show Only: ConiferForest □
Show Only: MixedForest □
Show Only: EarlyNon-Forest □

Map Tools ✖

- Share this Map
- Legend (click to clear layer)

States

TACs-NASA Products

% Change in NDVI

- 61% to -99%
- 30 %
- 20 %
- 15 %
- 12.5 %
- 10 %
- 5 %
- 3 %
- 1.5 %

0 %
+ 25 %
+ 100 %
Snow

Figure 2 Layer-level masking prototype

was activated (or deactivated), it should be active (or inactive) for all NDVI layers. With this approach, a *one-to-many* relationship existed between the layer and the mask, respectively.

With global masking, the masks are activated completely separately from the map layers. In Figure 4, you can see that the conifer, deciduous, and mixed forest masks are active. Note also that the layer titled *current_apr5_may8*.tif is active. After working with the global masking design for a few weeks, the evaluation team provided feedback that the preferred design for interactive masking in the FCAV interface was achieved.

A class diagram is simply a static model that shows the classes and their relationships as they remain constant in the application over time (Willard 2007). Object oriented modeling

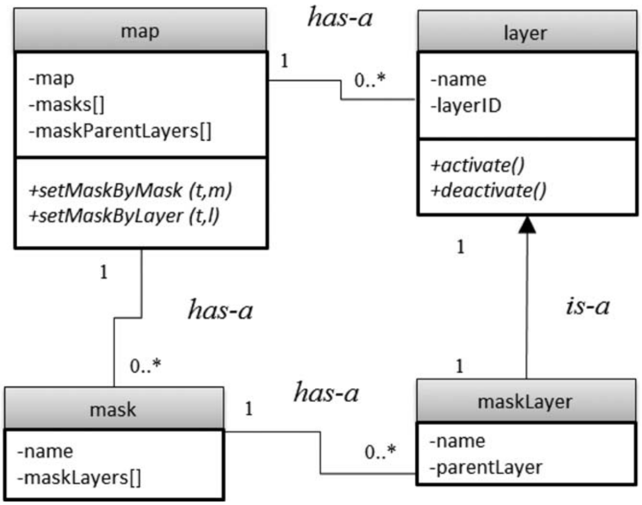
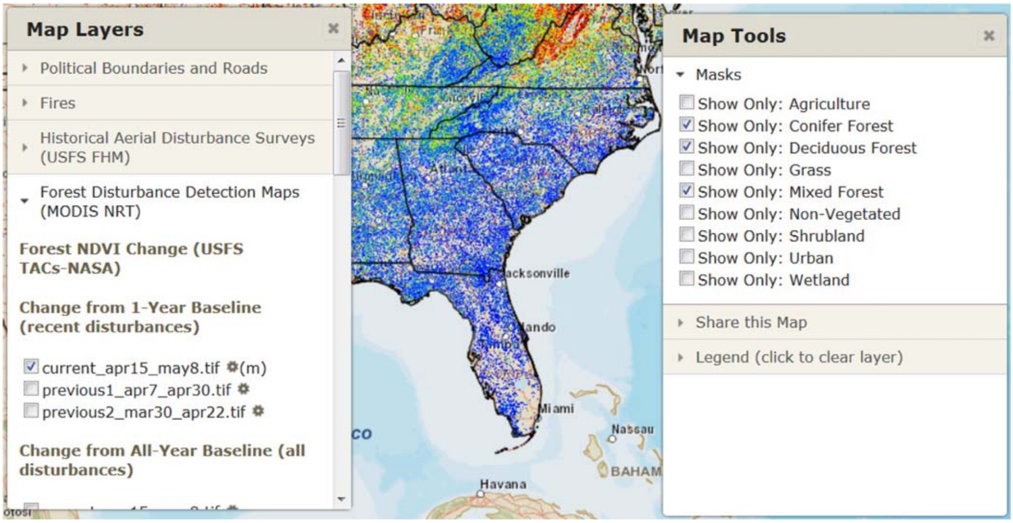


Figure 3 Global masking class diagram



Map Layers

- Political Boundaries and Roads
- Fires
- Historical Aerial Disturbance Surveys (USFS FHM)
- Forest Disturbance Detection Maps (MODIS NRT)

Forest NDVI Change (USFS TACs-NASA)

Change from 1-Year Baseline (recent disturbances)

☑ current_apr15_may8.tif (m)
□ previous1_apr7_apr30.tif
□ previous2_mar30_apr22.tif

Change from All-Year Baseline (all disturbances)

Map Tools

- Masks

□ Show Only: Agriculture
☑ Show Only: Conifer Forest
☑ Show Only: Deciduous Forest
□ Show Only: Grass
☑ Show Only: Mixed Forest
□ Show Only: Non-Vegetated
□ Show Only: Shrubland
□ Show Only: Urban
□ Show Only: Wetland

- Share this Map
- Legend (click to clear layer)

Figure 4 Global masking prototype

with UML allows us to formalize the metaphor approach. An *object* is the run-time instantiation of a class (Egenhofer and Frank 1992) and can serve as a conceptual bridge between the *data* and *interface* software architectures (Gordillo et al. 1999). As illustrated in UML class diagram of Figure 5 the software architecture in support of layer-level masking treated the mask and its resultant layer as a single object, *maskLayer*.

The class diagram above (illustrated in Figure 3) shows the object architecture to support the global masking functionality design. Illustrated in this class diagram is that a *maskLayer* is a subclass of a *layer*. From this conceptualization, we see that an association exists where the

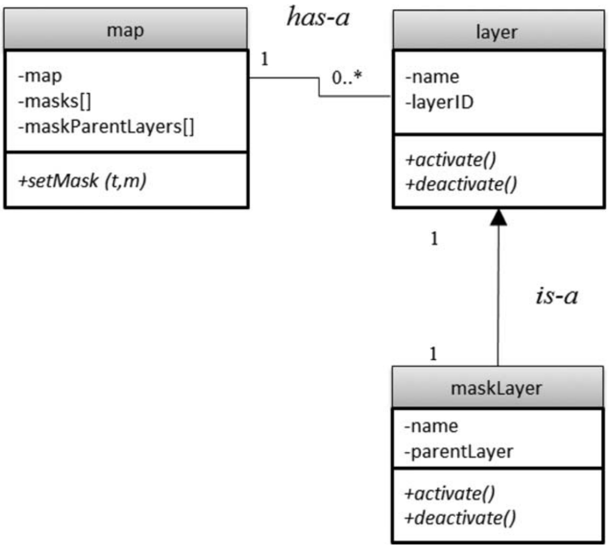


Figure 5 Layer-level masking class diagram

subclass, *maskLayer* in this case, inherits from the superclass or the *layer*. Here again we can infer that a parent/child relationship exists between the *maskLayer* and its "parent" *layer*. It is seemingly intuitive, then, from the user perspective that when one activates a mask, they are essentially seeing the subset of a "parent" layer. However, not necessarily intuitive is that the *mask* should be treated as an object unto itself and that this object should keep track of the *maskLayers[]* (an array data structure). The feedback received from the evaluators during the UCD process was instrumental in the final global masking functionality design.

With the knowledge obtained during the iterative design process the final design and development of interactive masking capability within the FCAV was able to progress. The software redesign of masking functionality from layer-level to global was greatly aided by the conceptualization of user interaction as metaphor (Kuhn and Frank 1991, Madsen 1994). The software design to support this feature led to the creation of two unique methods within the *map* class: *setMaskByMask()* and *setMaskByLayer()*. These methods each receive a toggle (*t*) and a mask (*m*) or layer (*l*) parameter, respectively. By implementing these two methods at the *mask* object we could then create, for the benefit of the map user and software design, the illusion that masks are independent of their parent layers, supporting the class diagram presented in Figure 5.

4 Conclusions and Future Work

Landscape architects at university campuses are able to discern where students and staff walk in order to better place paved walkways to avoid wear and tear in existing green spaces. However, this type of design by perceived use is not always feasible when designing systems with complex HCIs such as those increasingly available within Web GIS. A particular challenge exists when the HCI has nested functional expectations where interdependent events must occur in certain sequences. This HCI complexity implies that even very similar goals are often completed using very different processes, thus limiting the ability to generalize any findings (Brown et al. 2013). Also, it is important not to reduce design to the least number of clicks, but rather the least information cost while retaining cognition that achieves the desired results (termed utility).

This article presents a case study of how a UCD approach was utilized during the addition of interactive masking capability to an existing Web GIS. Hopefully, this article illustrates that UCD can be useful in improving both usability and utility of cartographic interface design within Web GIS. The design of these interfaces should be easy to use, while still complex enough to achieve utility, in this case forest change assessment. Because the evaluators used in this article were also the primary users of the FCAV, the design was mostly driving by considerations of utility with an eye towards usability. Future work in this area would benefit by considering evaluations of how new users (with forest change domain knowledge) perceive the final design of the interactive masking capability. This work should include empirical usability tests where new users of the FCAV are presented with the pre- and post-masking interfaces and asked to conduct forest change assessment activities.

References

Bacastow, T. S., & Bellafiore, D. J. (2009). Redefining geospatial intelligence. *American Intelligence Journal*, 27, 38–40.

Bacastow, T. S. (2014). Elements of a use case diagram. Retrieved from http://www.e-education.psu.edu/geog468/18-p4.html

Boulos, M. N. K., Warren, J., Gong, J., & Yue, P. (2010). Web GIS in practice VIII: HTML5 and the canvas element for interactive online mapping. *International Journal of Health Geographics*, 9, 14.

Brown, M., Sharples, S., Harding, J., Parker, C. J., Bearman, N., Maguire, M., . . . and Jackson, M. (2013). Usability of Geographic Information: Current challenges and future directions. *Applied Ergonomics*, 44, 855–865.

Crampton, J.W. (2002). Interactivity types in geographic visualization *Cartography and Geographic Information Science*, 29, 85–98.

Dennis, A., Wixom, B.H. & Tegarden, D. (2005). *Systems analysis and design with UML Version 2.0*. New York, NY: John Wiley and Sons.

Egenhofer, M.J., & Frank, A. (1992). Object-oriented modeling for GIS. *Journal of the Urban and Regional Information Systems Association*, 4, 3–19.

Eicher, C., Bedel, M., Neuffer, D., Kappeler, S., Neun, M., & Schoening, R. (2010). Smiling Cartographers: Making GIS More Usable for Cartography. *Cartographic Journal*, 47, 173–179.

Gordillo, S., Balaguer, F., Mostaccio, C., & Neves, F. D. (1999). Developing GIS applications with objects: A design patterns approach. *Geoinformatica*, 3, 7–32.

Goodchild, M.F. (2013). Cartographic futures on a digital earth. *Cartographic Perspectives*, 36, 3–11.

Jordan, P. W. (1998). *An introduction to usability*. Boca Raton, FL: CRC Press.

Kirwan, B. & Ainsworth, L.K. (1992). *A guide to task analysis*. London, UK: Taylor and Francis.

Kuhn, W., and Frank, A. U. (1991). A formalization of metaphors and image-schemas in user interfaces. In *Cognitive and linguistic aspects of geographic space* (pp. 419–434). Springer Netherlands

Layman, I., Diep, M., Nagappan, M., Singer, J., DeLine, R., & Venolia, G. (2013). Debugging revisited: Toward understanding the debugging needs of contemporary software developers. Paper presented at the International ACM/IEEE Symposium on Empirical Software Engineering and Measurement, Baltimore, Maryland, 383–392.

Love, S. (2005). *Understanding mobile human-computer interaction*. Oxford, UK: Butterworth-Heinemann.

Loveland, T., Merchant, J., Brown, J., & Ohlen, D. (1991). Development of a land-cover characteristics database for the conterminous U. S. *Photogrammetric Engineering and Remote Sensing*, 57, 1453–1463.

Madsen, K. H. (1994). A guide to metaphorical design. *Communications of the ACM*, 37, 57–62.

Marchionini, G., & Komlodi, A. (1998). Design of interfaces for information seeking. *Annual Review of Information Science and Technology*, 33, 89–130.

Mack, R.L., & Nielsen, J. (Eds.). (1994). *Usability inspection methods*. New York, NY: John Wiley and Sons.

Mills, R.T., Kumar, J., Hoffman, F.M., Hargrove, W. W., Spruce, J.P., & Norman, S.P. (2013). Identification and visualization of dominant patterns and anomalies in remotely sensed vegetation phenology using a parallel tool for principal components analysis. *Procedia Computer Science*, 18, 2396–2405.

Nielsen, J. (1999). *Designing web usability: The practice of simplicity*. London, UK: New Riders Publishing.

Norman, D.A. (1988). *The psychology of everyday things*. New York, NY: Basic Books.

Polson, P.G., Lewis, C., Rieman, J., & Wharton, C. (1992). Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *International Journal of Man-machine Studies*, 36, 741–773.

Preece, J., Rogers, Y., & Sharpe, H. (2002). *Interaction design: Beyond human-computer interaction*. New York, NY: John Wiley and Sons.

Richards, J., & Egenhofer, M. (1995). A comparison of two direct-manipulation GIS user interfaces for map overlay. *Geographical Systems*, 2, 267–290.

Roth, R.E. (2014). Interactive maps: What we know and what we need to know. *Journal of Spatial Information Science*, 6, 59–115.

Sarris, S. (2012). *HTML5 unleashed*. Harlow, UK: Pearson Education.

Spruce, J. P., Sader, S., Ryan, R. E., Smoot, J., Kuper, P., Ross, K. and Hargrove, W. (2011). Assessment of MODIS NDVI time series data products for detecting forest defoliation by gypsy moth outbreaks. *Remote sensing of environment*, 115(2), 427–437

Shapiro, F.R. (1987). Etymology of the computer bug: History and folklore. *American Speech*, 62, 376–387.

Taivalsaari, A., & Mikkonen, T. (2011). The web as an application platform: The saga continues. Paper presented at the Thirty-seventh EUROMICRO Conference on Software Engineering and Advanced Applications, Oulu, Finland, 170-174.

Tsou, M.H. (2011). Revisiting web cartography in the United States: The rise of user-centered design. *Cartography and Geographic Information Science*, 38, 250–257.

Wade, T. & Sommer, S. (2006). *A To Z: An illustrated dictionary of geographic information systems*. Redlands, California CA: Esri Press.

Wickens, C.D., Gordon, S.E., & Liu, Y. (2004). *An introduction to human factors engineering* (2nd Ed.). Upper Saddle Creek, NJ: Prentice Hall.

Willard, B. (2007). UML for systems engineering. *Computer Standards and Interfaces*, 29, 69–81.