Chapter 12
Web Map Design for a Multipublishing Environment Based on Open APIs

Pyry Kettunen, L. Tiina Sarjakoski, Salu Ylirisku, and Tapani Sarjakoski

Abstract The aim of the study described in the paper is to carry out research on the utilization of web-based multipublishing for the purpose of outdoor activities. The idea behind a multipublishing service is that the service is able to deliver different kinds of outdoor maps through a number of channels and at varying scales from a single data core. The paper focuses on the technical solutions, design principles and usability testing of a web map that was created to serve as one of the publishing channels. The other channels of the implemented multipublishing service are map applications for a mobile phone and a multi-touch screen, and printed graphic maps. The environment is based upon a web server that provides raster and vector geodata for the channels through open standard web services. Both the server and the web map client application are built on free and open source geospatial software, to which modifications were made to achieve the design goals of the multipublishing environment. The web map user interface (UI) aims at providing a complete tool to interact with the maps being published and it shares similar design with the other publishing channels. We applied the "minimalist" and "direct manipulation" design paradigms for UI design to limit the user's cognitive overload.

12.1 Introduction

Network-delivered interactive and dynamic digital maps with high cartographic quality and user-friendly interfaces have been a dream of cartographers for years (Moellering 1984; MacEachren 1994; MacEachren and Kraak 1997; Cartwright 1997; Slocum et al., 2001). It has proven difficult, though, to design and implement maps that satisfy all of these aspects. The problem has been divided into a number

P. Kettunen (✉)
Department of Geoinformatics and Cartography, Finnish Geodetic Institute, P.O. Box 15, 02431 Masala, Finland
e-mail: Pyry.Kettunen@fgi.fi

M.P. Peterson (ed.), *Online Maps with APIs and WebServices*, Lecture Notes in Geoinformation and Cartography,
DOI 10.1007/978-3-642-27485-5_12,

of sub-problems which range from web technology to usability issues. Typically, the development of the sub-problems has followed the progress of the dedicated research fields. For instance, advances in web technology have had a major influence on solving the problems of the delivery of digital maps, and usability research has guided the development of the map UIs (Haklay and Zafiri 2008; Roth and Harrower 2008).

Digital cartography utilizes and integrates advanced information technologies, but it also has some specific problems to solve, most prominently that of cartographic generalization (Sarjakoski 2007). A stable integration of the technologies has become a reality only during the first decade of the twenty-first century and digital web maps are still often considered to have insufficient performance and poor usability (Nivala et al., 2008; Coltekin et al., 2009). The holistic design of web maps is necessary in order to provide a satisfactory user experience and to fulfill the expectations and needs of the users, while taking into account the usability, cartographic quality and an efficient computing architecture. Such a design is of noteworthy complexity and can significantly benefit from open application programming interfaces (APIs) that reduce the need for implementing the underlying intricate processes.

12.1.1 Background of the Study

The research described in the paper is part of two projects: "Multipublishing in supporting outdoor leisure activities" (MenoMaps), and MenoMaps II (Map services for outdoor leisure activities supported by social networks). The aim of the MenoMaps (2008–2010) project has been to carry out research on utilizing web-based multipublishing for the purpose of outdoor activities, and to develop a user interface (UI) that is easy to use, useful and challenging and which provides a pleasurable use experience (Sarjakoski et al., 2009; Flink 2009). The main outcome of the first project was a map-based multipublishing service prototype.

The idea of a multipublishing service is that the service is able to deliver different kinds of outdoor maps through a number of channels and at varying scales from a single data core. The data core utilizes integrated data sources, such as high-resolution digital terrain models based on airborne laser scanning. Figure 12.1 shows the implemented channels of the MenoMaps service: the map applications for an iPhone (Kovanen et al., 2009) and a multi-touch screen (Sarjakoski et al., 2010), printed graphic maps (Oksanen et al., 2011) and a web map, the channel which is the focus of this paper. The map-based multipublishing service prototype will be further developed in the follow-up project, MenoMaps II (2010–2013) and it will be exhibited in the Nuuksio Nature Centre in southern Finland in the beginning of 2013.

12.1.2 Structure of the Paper

In the following section, we review previous research in the field. The viewpoints for the research works come from three perspectives: the technological point of

Fig. 12.1 The channels of the MenoMaps multipublishing service: the map applications for a mobile phone (iPhone) and a multi-touch screen, printed graphic maps and a web map (Flink et al., 2011). The data for all the channels is derived from the same data core

view, the design point of view and the usability point of view. After Sect. 12.2, we present a case study, in which a multipublishing service prototype for outdoor activities was implemented based on geospatial Free and Open Source Software (FOSS), for example OpenLayers application programming interface (API). Finally, we summarize the experiences and success of the prototype in terms of architecture, design and usability.

12.2 Previous Studies

The following points of view have been considered relating to studies on web-based map applications: technological perspectives, UI design and usability.

12.2.1 Technological Solutions for Web Maps

Web-based APIs have recently become common platforms for building web maps. A web map API is a collection of programming utilities which can be used for the implementation of a web map without the need to program the elementary functionalities from scratch. Web map portals such as Google Maps, Yahoo Maps and Bing Maps provide APIs for using their maps on web pages. On the other hand, stand-alone, data-independent client APIs such as OpenLayers API and ArcGIS Web Mapping APIs are also available. Current APIs allow users to add a default map application to a web page with very basic programming skills, but they also provide a developer with a number of tools for customized implementation. The APIs are generally based upon Asynchronous JavaScript and XML (Ajax) web technologies that run on ordinary web browsers without additional plugins.

The Open Web Service (OWS) specifications of Open Geospatial Consortium (OGC) define a set of network interface standards for transferring geospatial data between clients and servers. Web maps typically use the Web Map Service (WMS) for requesting and receiving georeferenced raster data that are static map images, and the Web Feature Service (WFS) for the retrieval and remote manipulation of vector geodata (e.g., González et al., 2009; Henrie 2009). The OWSs allow developers to collect multiple independent data sources from distinct servers into a single web map application as well as distribute geodata from a single server to multiple independent clients. In this way, the OWSs enable the creation of federated and heterogeneous GIS server architectures that can grow into wide data infrastructures (Webster 1988). Versatile and flexible web-based system architectures are possible, like the multi-channel publishing of maps for diverse devices (Lehto et al., 2001).

The OWSs have been actively implemented in Free and Open Source Software (FOSS) GIS applications to fit into open GIS architectures. An open architecture means that both the system components and the binding architecture are accessible to a developer on the server and client side (Dunfey et al., 2006). The developer can fully control an open system since he can read the source code to understand how the system operates and is also capable of modifying the code when needed. FOSS software provides, often easily, the much needed interoperability in GIS when several data formats and coordinate systems must be used in concert. Dunfey (2006) achieved promising results in building a proof-of-concept open architecture for a SVG map viewer on top of FOSS components, confirming their flexibility and functionality. Henrie (2009) reports similar results in building a web map for historic town plans based upon FOSS GIS software. He points out that it was easy to implement with these tools, and that the open architecture allowed good performance.

The rendering of vector data into bitmap graphics, which can be done either on the client or server side, is, computationally, one of the most demanding parts of web map architecture. The server-side rendering usually outperforms the client-side rendering because of the superior computing power of the server. However, client-side rendering is necessary when the vector attributes should be passed to the client application. In both cases, the styling rules for rendering vectors are needed to specify the appearance of the output.

Styled Layer Descriptor (SLD) is an open OGC standard for defining vector styles. An XML-derived SLD enables graphically rich, dynamic display styles for georeferenced vector features such as arbitrarily dashed and round-ended lines. The styles can be piled up to form complex styles and filters can be applied to restrict the effect of a style on specific zoom levels or targeted features. Many FOSS GIS applications apply SLD either on the side of the client or the server (Zipf 2005).

12.2.2 Web Map UI Design and Usability

MacEachren and Kraak (1997) discuss exploratory cartographic visualization (ECV) within the context of the use space cube of maps (MacEachren 1994). They state that map display systems need to be designed following the intended use of every single map, whether that use is exploration, analysis, synthesis or the presentation of geographic data. Several web map UIs have been built in order to study the UI design for ECV. One of the earliest such UIs was in a software system called Descartes, which provided versatile dynamic manipulation tools for exploring spatial data using maps (Andrienko and Andrienko 1999). The UI of Descartes applied multiple linked views (MLVs) which are central to the graphical UI (GUI) design: the contents of the multiple graphical windows in the UI are actively tied together so that changing the view in one window updates the other windows accordingly (Roberts 2005).

The MLVs fit well for creating UIs for ECV because they provide parallel visualizations of the same data and, thus, prepare the way for new, interesting cognitive linkages (Roberts 2008). Recently, the study of the ECV has continued through web map UIs like London Profiler (Gibin et al., 2008) and Nunaliit Cybercartographic Atlas Framework (Pulsifer et al., 2008). Both of these interactive, dynamic digital maps have been designed with an amateur in mind, taking care to avoid the undesired complexity of usage which is common in the professional GISs. In addition to combine the data, these systems also aim at displaying cartographically polished geographical data that can be easily explored by a content-oriented user.

The UI design must consider usability issues in order to bring the user experience of the interactive and dynamic digital maps to a pleasing level (Nivala 2007). Our UI research applies design methods called "minimalism" and "direct manipulation". Minimalist design means that every component included within a UI must have a good reason to be included and spare components are left out (Carroll 1985). Direct manipulation design places the functionalities of a UI on the objects so that the objects can be manipulated just where they are (Hutchins et al., 1985). For a successful final product, a usability evaluation of the already implemented software is also necessary in order to detect the remaining problems in usage situations (Nivala et al., 2008).

Usability evaluation methods distribute on a continuum between controlled experimentation using quantitative measures such as eye-movement tracking, and usability testing using qualitative measures such as focus groups (Roth and Harrower 2008). Comparative usability studies have been made in order to create design guidelines for interactive and dynamic digital maps. Nivala et al., (2008) conducted a usability evaluation of web mapping sites using user tests and expert evaluations. They found that web mapping sites should be improved and harmonized in order to better guide the user. Coltekin et al., (2009) compared two separately designed web map UIs using eye-tracking combined with performance metrics, system usability scale (SUS) and participant interviews. The findings revealed that the users found the design with the map in the focus and with floating menus more attractive than traditional map UI design with a separate, hierarchical layer-based menu. Users also felt that the former maps performed better.

12.3 Case Study: MenoMaps Web Map

A multipublishing service prototype was designed and built as part of the MenoMaps project for outdoor leisure activities in Nuuksio National Park in the southern Finland. The MenoMaps web map was implemented as one of the channels for personal exploration and annotation of the maps within the multi-publishing environment. The intended use of the web map is for exploring the national park and planning activities, for example before a hike, as well as memorizing and sharing the activities after a hike. In this section, we present the MenoMaps web map from the viewpoints of the architecture, UI design and usability evaluation.

12.3.1 System Architecture

12.3.1.1 Overall Architecture of the Multipublishing Environment

The multipublishing service prototype of the MenoMaps project was implemented according to the open architecture model so that both the architecture and the components can be accessed (Dunfey et al., 2006). The FOSS software was used for the implementation because we wanted to test the capabilities of such a solution. We wanted to create a stand-alone system that would be flexible for a developer. We also wanted to use our own data independently from the web map portals such as Google Maps and Bing Maps. The architecture (Fig. 12.2) consists of three functional levels that inter-operate on the same or distinct computers:

1. The user interface level displays the maps on the client side through the following channels: the web map on a web browser, a mobile application on an iPhone (Kovanen et al., 2009), a multi-touch screen map application in a public space, (Sarjakoski et al., 2010) user-made map printouts and professional printed maps (Oksanen et al., 2011) all utilize the same data core for the map contents.
2. The data access level supplies the geodata requested by the clients and stored on the servers. The data access level also performs data transformations when

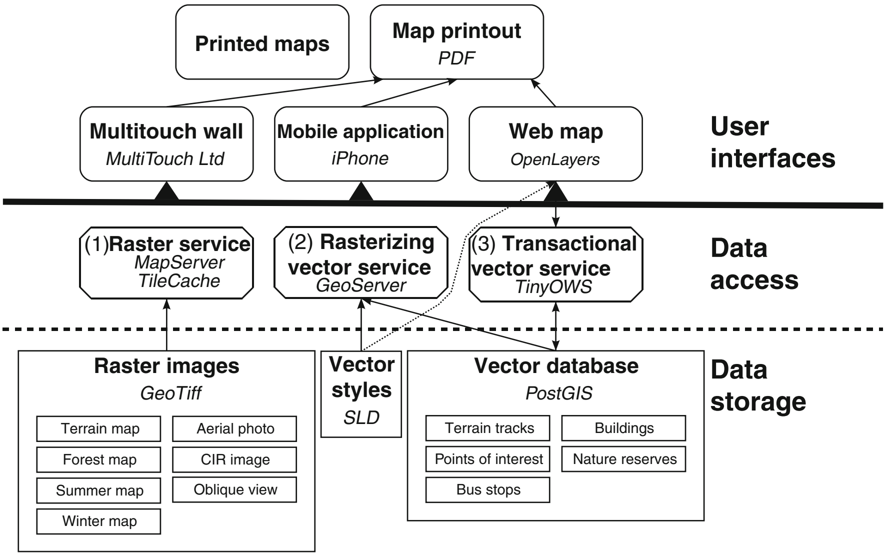

Fig. 12.2 Overall architecture of the MenoMaps multipublishing service prototype. The utilized technologies are *italicized*. The *arrows* denote the data flow

needed. The MapServer is used for supplying the raster data together with the TileCache caching service. The GeoServer and TinyOWS supply the vector data as rasterized images and geospatial vectors, respectively.

3. The data storage level contains and delivers the source data on the server. The raster data is stored as static, georeferenced and tiled GeoTiff image files on the hard disks. The vector data is stored in an open multi-relational PostgreSQL database that is spatially enabled as a PostGIS database.

12.3.1.2 Web Map Architecture

The web map client application was implemented using Asynchronous JavaScript and XML (Ajax) web technologies. HyperText Markup Language (HTML) was used for creating the general web site layout, the appearance of which was defined using Cascading Style Sheets (CSS). The web map interactivity is produced with JavaScript programming language, which enables the dynamic modification of HTML through the Document Object Model (DOM). Web map functionality is realized using OpenLayers API, which is a free and open source web map UI programming library in JavaScript that takes care of the OWS data requests, data rendering and map interaction.

The web map client displays the geodata that is requested from our three data access services (Fig. 12.2). The WMS interfaces of the services are used for retrieving raster data (1, 2) and a WFS for vector data (3):

1. The raster service provides the web map with static map images that are used as background layers in the map view. The TileCache delivers tiled Portable Network Graphic (PNG) files directly from a disk cache when the requested tiles are available. Otherwise, the TileCache requests the tiles from the neighbouring MapServer, which generates PNG tiles from the GeoTiff data store.
2. The rasterizing vector service provides the web map with static PNG images that are rendered as overlaid layers in the map view. The vector data is rasterized by the GeoServer using styling rules from the SLD files. The rasterization is needed for some vector layers because processing large amounts of vectors is inefficient on the client-side. Another reason for using server-side rasterization is that the GeoServer is capable of styling features that OpenLayers cannot render, for example, multiple overlaid styles and grouping of point symbols (Fig. 12.3).
3. The transactional vector service provides the web map with vectors that are rendered as overlaid layers in the map view and carries their data attributes along to be utilized on the client. The vectors are provided in the Geographic Markup Language (GML) format, which is transformed into SVG by the OpenLayers API and rendered into the map view by a SVG renderer of a web browser. In the case of Internet Explorer, which does not support SVG, OpenLayers uses Vector Markup Language (VML). Vector appearance is drawn on the client by the OpenLayers API according to styles defined in SLD files, which are separately collected from the server.

In addition to querying and displaying raster and vector geodata, the MenoMaps web map allows users to modify certain vector database tables through the transactional operations of the WFS. The INSERT, UPDATE and DELETE operations of the WFS are handled by the TinyOWS service and passed to the PostGIS database as Structured Query Language (SQL) queries.

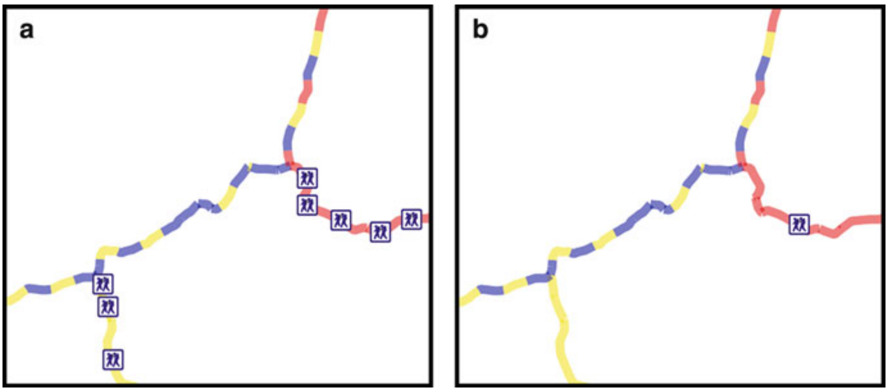

Fig. 12.3 Multiple styles are used for displaying three overlapping routes (**a**) without and (**b**) with grouping of point symbols. Piled line styles are used on single vectors

12.3.2 Web Map User Interface Design

The web map UI is directed at casual users interested in outdoor life who possess basic web browsing skills. Elderly users are considered as a special user group because they are an important subgroup of outdoor people with special needs. The aim of the UI design was to create an easy-to-use map UI which supports map exploration and which can easily be learned. We viewed the "minimalist" and "direct manipulation" design paradigms, as well as "expand-in-context" design pattern, as appropriate for achieving these goals. The web map UI is comprised of three main usage environments: a home page for configuring the UI, a map view for interacting with the map and a tool menu for controlling the map functionalities.

12.3.2.1 Home Page

The home page is initially opened when the user arrives at the MenoMaps web map page. The page lets the user easily configure the display parameters for the map usage (Fig. 12.4). The user chooses the parameters from a floating window on top of

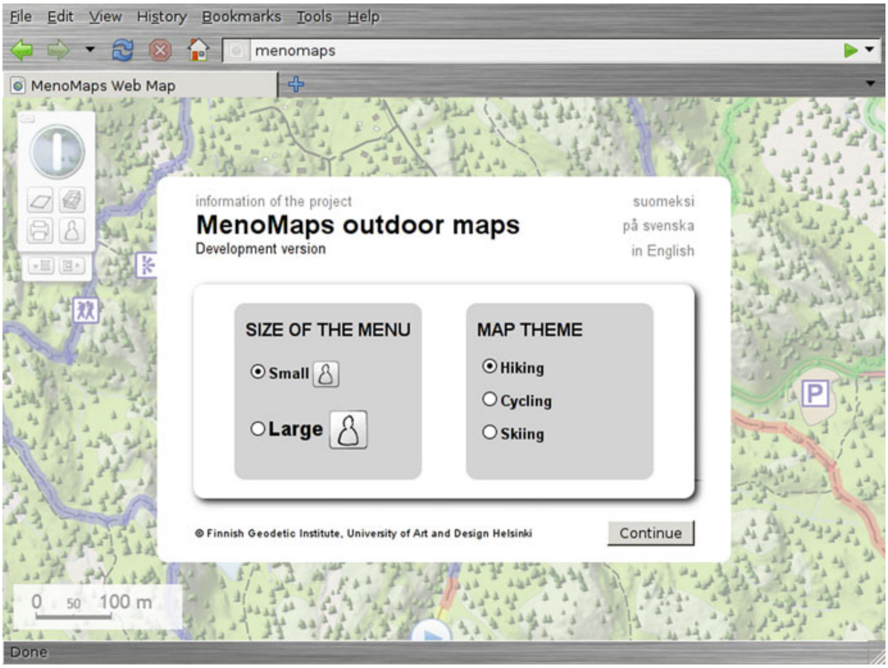

Fig. 12.4 The home page is initially opened when the user arrives at the MenoMaps web map page. The default choices of display parameters are selected (Hiking theme)

the faded map view so that the effects of the changes can be simultaneously observed on the map below. Figure 12.4 shows the default choices.

The main choices of the home page are “size of the menu” and “map theme”. The “size of the menu” lets the user change the sizes of the icons and fonts in the UI, a selection that is particularly targeted at elderly users who may be far-sighted or have difficulties in pointing accurately with a mouse. The “map theme” provides a simple means to change the contents of the map view according to the three popular outdoor activities of hiking, cycling and skiing. When the user selects the map theme, this activates pre-defined selections of background map and overlaid map layers (Figs. 12.5, 12.6, 12.7). The contents of the overlaid layers are adapted according to the season and the interests in each outdoor activity. For example, swimming places are not shown in the skiing theme.

The user can select the language of the UI at the web map’s home page, a selection that applies to textual contents in the UI. An information page about the project is available. This page opens via a separate tab or window in the browser. The home page can be closed and the map view entered by using the “continue” button or by clicking on the faded map view.

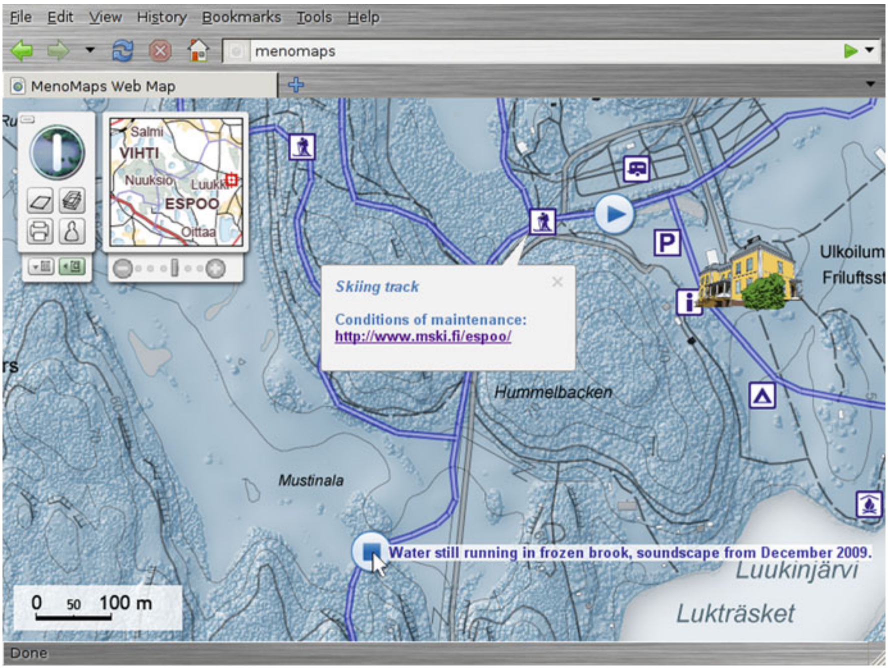

Fig. 12.5 The map view is based upon “minimalist design”. The primary functionalities are embedded within the map (Skiing theme)

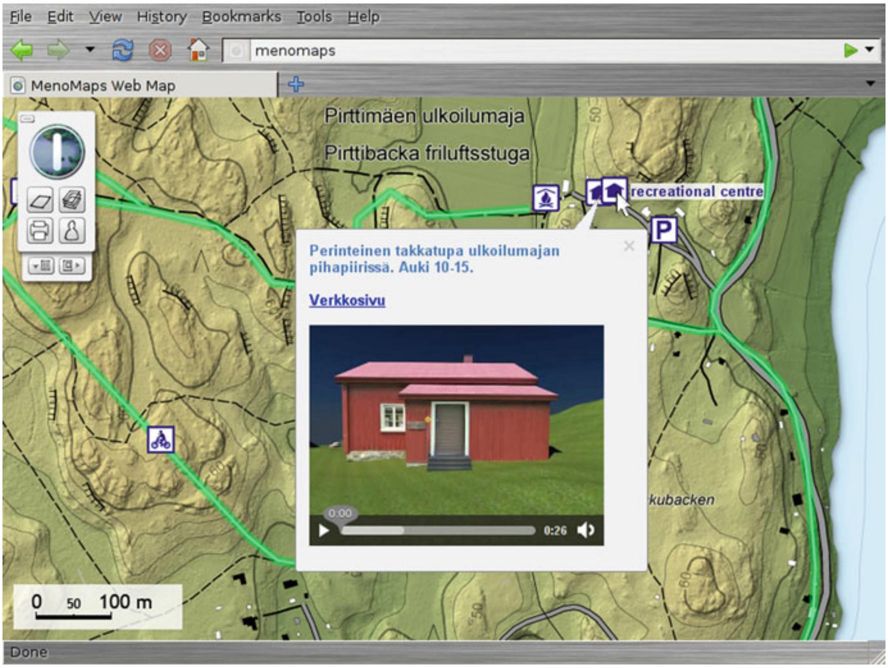

Fig. 12.6 Multimedia objects such as video clips are shown in the pop-ups (Cycling theme)

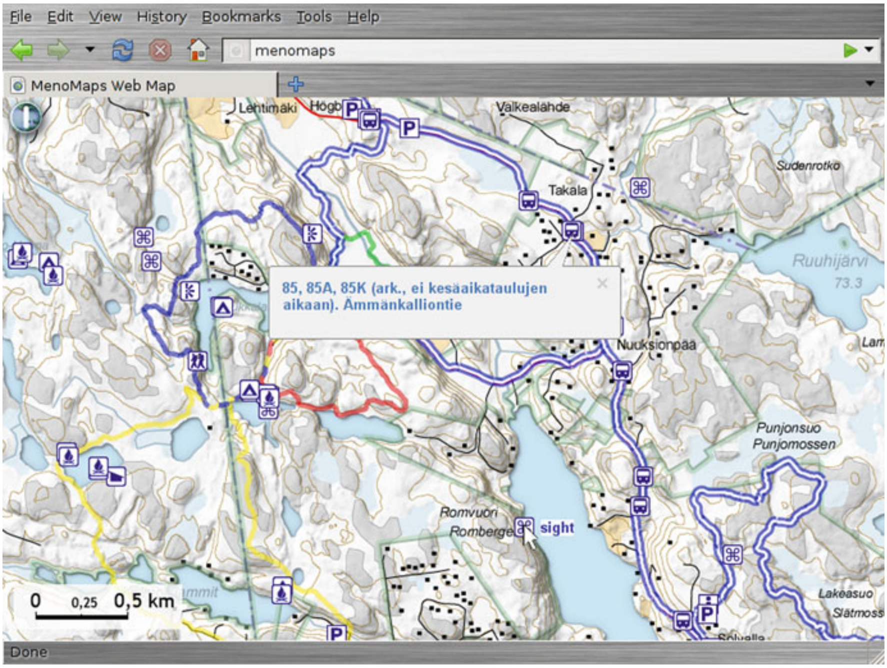

Fig. 12.7 The user can minimize the tool menu. Symbol sizes adapt to the scale level (Selected layers)

12.3.2.2 Map View

The map view is the main usage environment of the MenoMaps web map. Map viewing and map interaction take place within this environment. The map view is based upon the principle of minimalist design. It focuses on the map itself and minimizes the role of menus. The map area is maximized and primary functionalities are embedded within the map following the direct manipulation design paradigm: the user can zoom in and out with the mouse wheel, pan by dragging, get feature tooltips using the mouse-over and open feature pop-ups by clicking on the mouse. The tooltips offer short descriptions of the features and the pop-ups display feature-specific information that can contain text, images and videos (Figs. 12.5, 12.6, 12.7). The map is designed so that it can be used with the mouse, but keyboard controls are also provided for zooming and panning purposes.

The tool menu lies in the upper left corner of the map view. The menu can be minimized to the size of a small icon so that as large a map as possible can be displayed (Fig. 12.7). A transparent scale bar is located in the lower left corner of the map view and it changes dynamically in accordance with the zoom level.

12.3.2.3 Tool Menu

The abstract or complex functionalities of the web map are embedded in an icon-based tool menu in the upper left corner of the map view. The icon menu is built in such a way that it only shows a minimal amount of buttons at the beginning and, as the menu expands, more sophisticated tools can be opened when needed.

The tool menu is made up of three modules that can be opened and closed independently (Fig. 12.8). Initially, the main menu and the overview map are visible and the annotation menu is closed. The usage of the menus is guided by tooltips that come up when the cursor stops for a moment on the button (Fig. 12.8a).

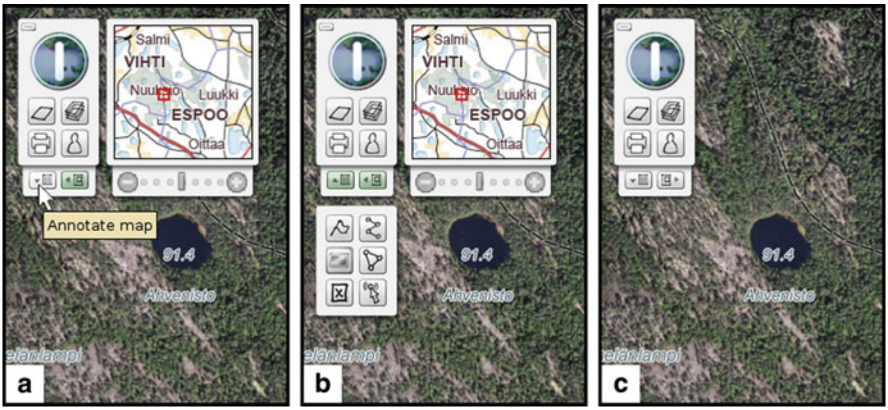

Fig. 12.8 Modularity of the tool menu: (**a**) initial state; (**b**) all menus opened; (**c**) main menu only

The clickable buttons are brightened on mouse-over so that the user can recognize their interactivity.

The overview menu consists of an overview map and zoom buttons. The overview map follows the position of the large map but can also be used for setting the position by clicking on the desired position or dragging the small pointer on the overview map. The zoom buttons are available as an additional means for changing the map scale. The zoom bar denotes the present zoom level among the available levels and can be clicked upon to set a desired view.

The annotation menu gives the user tools for making her/his own markings on the map (Fig. 12.8b). Buttons are available for adding a point of interest or an image, for drawing lines or areas and for editing and removing annotations. The buttons set the annotation mode on "click" so that the annotation can be directly made on the map. For example, the user may denote an interesting area by first clicking the "draw area" button in the right and center of the annotation menu. The map then switches to "draw area" mode and the user is able to draw a polygon on the map. The drawing is finished by double-clicking the location of the last node. The application adds the polygon to the "My content" layer.

The main menu consists of submenus for setting map layers and for communicating with the user database as well as for printing the map view (Fig. 12.9). The background layer is set in the "choose map" submenu and visible overlaid layers are chosen in the "show on map" submenu. The background map selection adapts the related overlaid layers accordingly; for example, ski tracks and winter-only bus stops are hidden when selecting the summer map. The "share" submenu provides a simplistic tool for storing and sharing self-made annotations using the server-side user database. An identifier is entered that is used when saving or loading vector features. The identifier can be used for sharing modifiable annotations with others. The printer button launches the printing functionality of the web browser and applies a printing layout for the present map view. The printing layout is fitted onto the paper, a general title is set on top of the page and the menus are excluded.

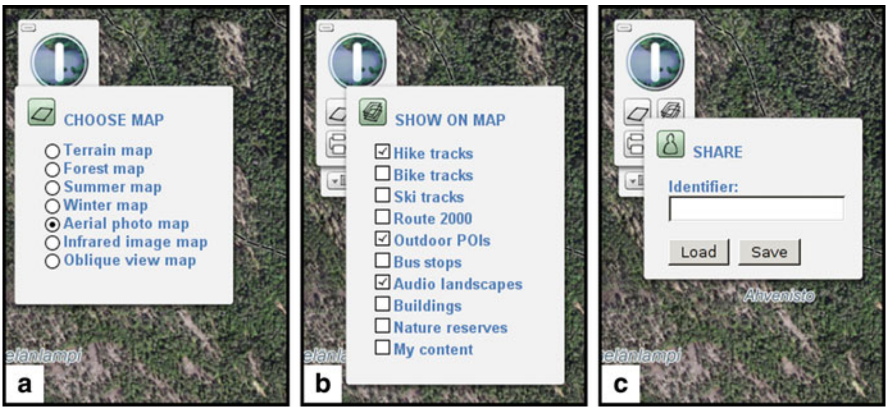

Fig. 12.9 Submenus of the main menu for choosing the visible layers and communicating with the database. The printing button launches the printing function of the browser

12.3.3 Web Map Usability Evaluation

Flink et al., (2011) studied the ease-of-use and usefulness of the MenoMaps multipublishing service by using the methods of thinking aloud and questionnaires. The testing situations were videotaped, and the sessions took about 1–1.5 h each. Six potential users recruited from the Finnish hiking association, Tunturilatu, took part in the usability evaluations. The tasks required the users to use various functions of the web map, such as finding additional information on the trails or drawing their own routes. After the thinking aloud tasks, the participants answered to a System Usability Scale (SUS) questionnaire concerning the tested maps. The SUS is a 0–100 scored Likert measure of the usability of a system as evaluated by users through ten subjective usability questions (Brooke 1996). In the present study, we made use of the preliminary results of the usability evaluations. The evaluations and results are more thoroughly discussed in (Flink et al., 2011).

The usability study guided us to instantly make two simple modifications on the web map functionalities that the participants reported to be problematic. For example, we set the overview map initially visible when entering into the map view the first time. We also added a tooltip to instruct that the drawing should be completed with a double-click, an operation that the participants found difficult to perform without guidance (Fig. 12.10).

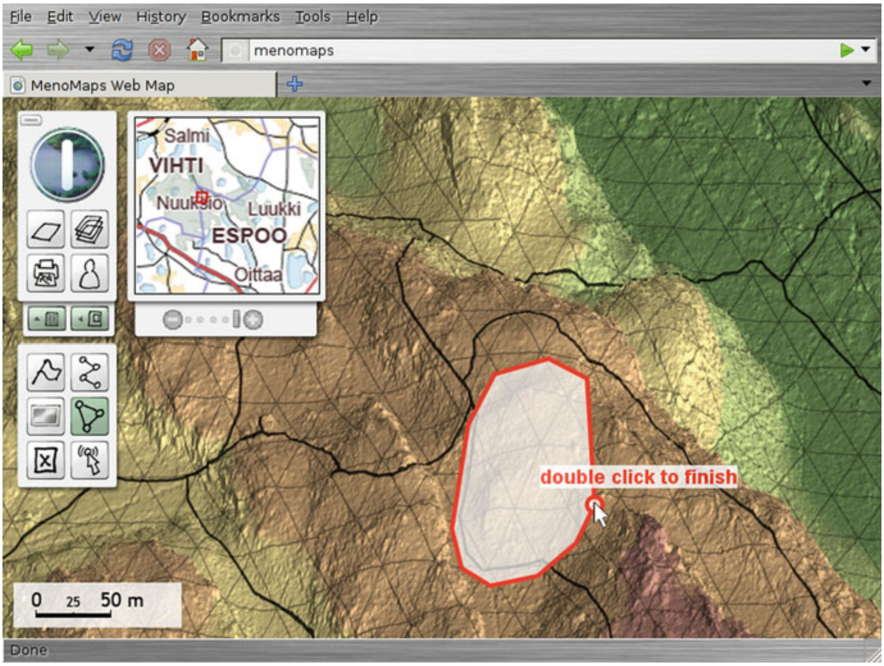

Fig. 12.10 As a result from the usability study, we added guidance so that users know how to complete the drawing action. The large menus and the font size are targeted at elderly users

12.4 Summary and Conclusions

The paper presented on-going research on designing and implementing a web map as one of the channels in a map multipublishing environment to be used for outdoor leisure activities in a national park. The web map client is an end-user application for exploratory cartographic visualization in order to enable creative planning of outdoor activities and to explore detailed information related to the national park.

The design of the web map UI was based upon "minimalist" and "direct manipulation" approaches, aiming at simple usage but versatile functionality. The UI consists of a configuring palette to select display parameters at the home page, a map view for displaying and exploring the map and an expanding tool menu for rich functionalities. The map view is the main usage environment of the web map with a maximized map area that incorporates the basic map functionalities such as panning and zooming. We considered elderly people as a special user group for whom large-size UI components were composed in the design.

The UI design was evaluated by another research group through a usability study. The participants considered the UI usable according to a SUS but encountered difficulties in finding some functionalities and in completing some tasks. Based on the results from the usability study, we made improvements to the UI design.

The web map architecture was fully implemented using FOSS GIS software, which we found to be a functional and flexible solution for creating an effective and robust web map. The FOSS GIS applied thorough implementations of OGC Open Web Services (OWS) for data access in the architecture and offered wide interoperability in terms of data formats and coordinate systems. It was possible to make modifications to the software on both the server and client sides, which was necessary for achieving the design goals of the web map. For example, we replaced the menus and graphical components of the OpenLayers web map client with self-made ones in order to efficiently apply minimalist design.

The MenoMaps web map will be further improved with regards to usability, cartographic design and architecture. The MenoMaps multipublishing service and its channels, including the web map, will be exhibited for the public in the new Nuuksio Nature Centre in southern Finland in the beginning of 2013. This will spur us to conduct additional usability evaluations and, consequently, introduce new UI developments for the web map. In particular, functionalities for sharing content in communities will be advanced.

Acknowledgements This survey is part of research for the MenoMapsI-II projects. These projects, funded by Tekes (the Finnish Funding Agency for Technology and Innovation), are joint ventures by the Finnish Geodetic Institute (FGI), Department of Geoinformatics and Cartography and Aalto University of Helsinki, School of Art and Design.

References

Andrienko GL, Andrienko NV (1999) Interactive maps for visual data exploration. Int J Geogr Inf Sci 13(4):355–374

Brooke J (1996) SUS – a quick and dirty usability scale. In: Jordan PW, Thomas B, Weerdmeester BA, McClelland IL (eds) Usability evaluation in industry. Taylor & Francis, London

Carroll JM (1985) Minimalist design for active users. In: Shackel B (ed) Human computer interaction – INTERACT'84. Elsevier Science, Amsterdam

Cartwright W (1997) New media and their application to the production of map products. Comput Geosci 23(4):447–456

Coltekin A, Heil B, Garlandini S, Fabrikant SI (2009) Evaluating the effectiveness of interactive map interface designs: a case study integrating usability metrics with eye-movement analysis. Cartogr Geogr Inf Sci 36(1):5–17

Dunfey RI, Gittings BM, Batcheller JK (2006) Towards an open architecture for vector GIS. Comput Geosci 32(10):1720–1732

Flink H-M (2009) User centred approach in the concept development of a map-based multi-publishing service. Master's thesis, University of Art and Design Helsinki

Flink H-M, Oksanen J, Pyysalo U, Rönneberg M, Sarjakoski LT (2011) Usability evaluation of a map-based multi-publishing service. In: Ruas A (ed) Advances in cartography and GIScience, vol 1, Lecture notes in geoinformation and cartography. Springer, Berlin, pp 239–257. doi:10.1007/978-3-642-19143-5_14

Gibin M, Singleton A, Milton R, Mateos P, Longley P (2008) An exploratory cartographic visualisation of London through the Google Maps API. Applied Spatial Analysis and Policy 1(2):85–97

González A, Velasco A, Ruiz C, Rubio JM, González J, García Á, Más S (2009) CartoCiudad: national database of all Spanish municipalities available freely including www.cartociudad.es and exploitable through its OGC services. New functionalities implemented. In: Proceedings of the 24th international cartographic conference. The World's Geo-Spatial Solutions, Santiago, 15–21 Nov 2009

Haklay MM, Zafiri A (2008) Usability engineering for GIS: learning from a screenshot. Cartogr J 45(2):87–97

Henrie D (2009) Ordnance survey historic town plans of Scotland (1847–1895): geo-referencing and web delivery with ArcIMS and OpenLayers. e-Perimetron 4(1):73–85

Hutchins EL, Hollan JD, Norman DA (1985) Direct manipulation interfaces. Hum-Comput Interact 1(4):311

Kovanen J, Sarjakoski T, Sarjakoski LT (2009) Studying iPhone as a media for context-aware map-based mobile services. In: Proceedings of the 6th international symposium on LBS & telecartography, CGS, University of Nottingham, Nottingham, 2–4 Sep 2009

Lehto L, Kähkönen J, Sarjakoski T (2001) Multi-purpose publishing of geodata in the web. In: Proceedings of the 4th AGILE conference on geographic information science, Brno, 19–21 April 2001. pp 209–214

MacEachren AM (1994) Visualization in modern cartography: setting the agenda. In: MacEachren AM, Taylor DRF (eds) Visualization in modern cartography. Pergamon/Elsevier Science, Oxford/New York, pp 1–12

MacEachren AM, Kraak M-J (1997) Exploratory cartographic visualization: advancing the agenda. Comput Geosci 23(4):335–343

Moellering H (1984) Real maps, virtual maps and interactive cartography. In: Gaile GL, Willmott CJ (eds) Spatial statistics and models. D. Reidel Publishing Company, Dordrecht, pp 109–132

Nivala A-M (2007) Usability perspectives for the design of interactive maps. Doctoral dissertation, Helsinki University of Technology

Nivala A-M, Brewster S, Sarjakoski T (2008) Usability evaluation of web mapping sites. Cartogr J 45(2):129–138

Oksanen J, Schwarzbach F, Sarjakoski LT, Sarjakoski T (2011) Map design for a multi-publishing framework – case menomaps in Nuuksio National Park. Cartogr J 48(2):116–123
Pulsifer PL, Hayes A, Fiset J-P, Taylor DRF et al., (2008) An open source development framework in support of cartographic integration. In: Peterson MP (ed) International perspectives on maps and the internet. Springer, New York, pp 165–185
Roberts JC (2005) Exploratory visualization with multiple linked views. In: Dykes J, MacEachren AM, Kraak M-J (eds) Exploring geovisualization. Elsevier, Oxford, pp 159–180
Roberts JC (2008) Coordinated multiple views for exploratory geovisualization. In: Dodge M, McDerby M, Turner M (eds) Geographic visualization: concepts, tools and applications. Wiley, Chichester, pp 27–48
Roth RE, Harrower M (2008) Addressing map interface usability: learning from the lakeshore nature preserve interactive map. Cartogr Perspect (Spring) 2008(60):46–66
Sarjakoski LT (2007) Conceptual models of generalisation and multiple representation. In: Mackaness WA, Ruas A, Sarjakoski LT (eds) Generalisation of geographic information: cartographic modelling and applications, series of international cartographic association. Elsevier, Burlington, pp 11–36
Sarjakoski LT, Sarjakoski T, Koskinen I, Ylirisku S (2009) The role of augmented elements to support aesthetic and entertaining aspects of interactive maps on the web and mobile phones. In: Cartwright W, Gartner G, Lehn A (eds) Art and cartography. Springer, Berlin, pp 109–124
Sarjakoski T, Kovanen J, Rönneberg M, Kähkönen J, Sarjakoski LT (2010) Data matrix technology for linking mobile maps in a web-based multi-channel service. In: Proceedings of ubiquitous positioning, indoor navigation and location-based service 2010, UPINLBS 2010, Kirkkonummi. IEEE, Seattle, 14–15 Oct 2010. ISBN 978-1-4244-7878-1, CD-ROM
Slocum TA, Blok C, Jiang B, Koussoulakou A, Montello DR, Fuhrmann S, Hedley NR (2001) Cognitive and usability issues in geovisualization. Cartogr Geogr Inf Sci 28(1):61–75
Webster C (1988) Disaggregated GIS architecture lessons from recent developments in multi-site database management systems. Int J Geogr Inf Sci 2(1):67–79
Zipf A (2005) Using Styled Layer Descriptor (SLD) for the dynamic generation of user- and context-adaptive mobile maps – a technical framework. In: Li K-J, Vangenot C (eds) Web and wireless geographical information systems, vol 3833, Lecture notes in computer science. Springer, Berlin, pp 183–193