

A Web-Based and Mobile Environmental Management System

J. D. Carswell^{1,*}, K. Gardiner¹, M. Bertolotto², A. Rizzini², and N. Mandrak³

¹Digital Media Centre, Dublin Institute of Technology, Ireland

²School of Computer Science and Informatics, University College Dublin, Ireland

³Dept. of Fisheries and Oceans, Canada Centre for Inland Waters, Burlington, Ontario, Canada

Received 9 January 2007; revised 2 October 2007; accepted 28 January 2008; published online 2 September 2008

ABSTRACT. This paper describes a Web-based and mobile system specifically developed to monitor fish species at risk. Such a system integrates spatial functionality to allow users not only to visualise maps and metadata of the area of concern but also to perform context-aware queries and updating of spatial datasets. The spatial datasets are provided by the Canadian Department of Fisheries and Oceans (DFO) and the prototype is customised to the specific needs of the Great Lakes Laboratory for Fisheries and Aquatic Sciences (GLLFAS) Fish Habitat Section requirements for fish species at risk assessment. Currently, researchers, habitat biologists and enforcement officers have access to the fisheries database, containing layers of biological information solely from the office. Delivering these data overlaid on base maps of the Great Lakes region to a GPS-enabled hand-held device and linking it to each task currently being investigated allows for mobile DFO biologists and enforcement officers in the field to make informed decisions immediately. In this paper we describe the system and demonstrate how it is used by the DFO in practice.

Keywords: spatially enabled mobile computing, web-based GIS, fish species at risk

1. Introduction

The diffusion of the Internet and mobile technologies has allowed for the development of many applications which can be accessible anytime and anywhere (Hinze et al., 2003; Schiller, 2000; Talukder et al., 2006). Many such applications can nowadays provide information based on the context in which they are used, including their geographic context. The advantages of these technologies are especially important for those professions that involve field work. Many such professions require the manipulation of spatial datasets, including maps of the areas visited. Examples of professional users that avail of spatial data during their field work are environmental scientists, surveyors, geographers.

This paper describes the development of a Web-based and Mobile Environmental Management System (MEMS) tailored to deliver context-aware functionality aided by visualization, analysis and manipulation of spatial and attribute datasets. The MEMS datasets are provided by the Canadian Department of Fisheries and Oceans (DFO) and the prototype is customized to the specific needs of the Great Lakes Laboratory for Fisheries and Aquatic Sciences (GLLFAS) Fish Habitat Management Group requirements for fish species at risk

assessment. Currently, biologists have only access to the fisheries data from their office. This greatly prevents them from interacting with the data in a real-time environment, reducing their productivity and effectiveness in the field. Spatially enabling a mobile device allows mobile GLLFAS biologists to make informed decision immediately.

This research concerns DFO priorities specifically to administer the fish habitat provision of the Fisheries Act, in particular those that are aimed at preventing the harmful alteration, disruption or destruction of fish habitat. This is done to conserve, restore and develop the productive capacity of habitats for recreational, commercial and subsistence fisheries both in the freshwater and marine environments (Minns, 1997, 2001). The functionality required by GLLFAS biologists includes access to geo-referenced maps and imagery, to overlay the current position on a map and to manipulate (e.g. input/edit/query) attribute data in the field while wirelessly connected (where possible) to the office database. Additional functionality also required is the ability to record, edit and view multimedia annotations, perform scientific/common name conversion and graph generations of results. The traditional “fish species at risk” work-flow, whereby scientists enter textual/pictorial information on paper field data sheets is inefficient, has potential for inaccuracies during both initial recording and subsequent data entry phases, and does not facilitate knowledge-sharing between staff. Also, different types of information may be stored in different locations and valuable time can often be lost trying to correlate data in order to make decisions. The proposed MEMS system has the following advan-

* Corresponding author. Tel.: +353 1 402 3264; fax: +353 1 402 3269.
E-mail address: jcarswell@dit.ie (J. D. Carswell).

tages over such practice:

- Facilitates knowledge-sharing and data analysis/ synthesis;
- Supports effective communication between different staff at different physical locations (e.g. scientists in the laboratory or office, and colleagues in the field);
- Allows important multimedia data and associated annotations to be combined with text-based records;
- Standard web-browser interface allows staff to input and access information anywhere at any time without having to return to dedicated access points (i.e. in the office);
- Reduces errors by eliminating time elapsed between data collection and data entry, saving time as well as paperwork.

The contribution of our work is the development of a system that facilitates the work of the GLLFAS biologists and makes their collected data more reliable and accessible. Such a system has been developed using the most recent GIS and mobile technologies and therefore provides a high level of functionality not just for collection and manipulation but also for spatial analysis of the data that was not available to these scientists before. This powerful system demonstrates the potential of integrating spatial functionality within environmental applications.

This paper describes the development of the MEMS system. In Section 2 the system architecture and the development environment are described. The system relies on the Oracle Spatial Database Management System that guarantees interoperability at the data level (Rigaux et al., 2002). Section 3 details the system requirements, while Section 4 discusses the functionality implemented with practical examples of its use through the Graphical User Interface (GUI). In particular, we describe the development of both the Web-based and mobile functionality. An on-line and an off-line version of the mobile system have been developed to allow for use in areas where wireless coverage is not available. Section 5 concludes with final remarks and future research directions.

2. System Architecture and Development Environment

The MEMS prototype relies on a typical three-tier architecture for enterprise information systems, composed of the client layer, application server layer, and the database layer (Figure 1).

This architecture focuses on the development of services for a versatile, extendible (J2EE) application server, instead of giving GIS capabilities to a large monolithic application. The communication between the client layer and the database are conducted through the application server layer. With this type of architecture, the processing load is balanced, as each tier of the system resides on a separate computer (Carswell et al., 2004; Rizzini et al., 2006).

The MEMS prototypes were developed using eSpatial's

iSmart Suite application development environment (eSpatial Solutions, 2004). This software is a collection of tools that enables developers to build and deploy spatial applications using sets of standard procedures. It offers developers a high-level development environment which is several times faster than developing the application from base Java. It reduces the need for major software engineering prerequisites that developers typically must have in order to understand and deploy a full spatial application. For the implementation and deployment parts of the system, *iSmart Eclipse* was used along with *iSmart Application Server*.

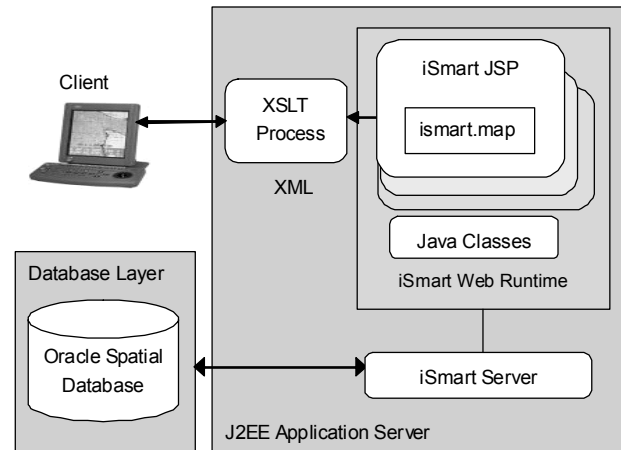


Figure 1. Three-tier architecture.

iSmart technology adopts the strategy of focusing the services on the application server layer of the system. This architecture allows for the development of individual components of the system separately, thus maintaining component independence. In this way, different parts of the system can be developed at different stages, some more than others, without affecting the entire system each time a change is made. For example, this architecture has proven ideal for developing Extensible Markup Language (XML)-based applications because all XML/XSLT processing is carried out on the middle tier of the system, without affecting client and/or database tier manipulation or development (Figure 2). The following sections describe each level of the three-tier architecture, and includes a brief description of the technologies that are required for each layer.

2.1. Client Layer

The client layer of the system consists of a Web-based (i.e. standard web-browser) interface, that enables data-input and analysis functionality to provide biologists with the ability to input, edit and annotate data over the Internet. As the user navigates within the system, the position and orientation of the user is displayed on a geo-referenced map of the area. This information is used when loading data into the database and for contextually querying the database in real-time. Client hardware consists of a Tablet PC or Personal Digital Assistant (PDA) equipped with a Global Positioning System (GPS) re-

ceiver to determine the position of the device. The device also has a General Packet Radio Service (GPRS) network connection for requesting and transmitting data to and from the application server over a wireless phone network. All mobile aspects of the system require a streamlined design, as current speed and bandwidth capabilities of hand-held devices are not yet comparable to those of desktop PCs. Therefore, to deliver content efficiently to these devices it is scaled down using a device-specific content adaptation approach.

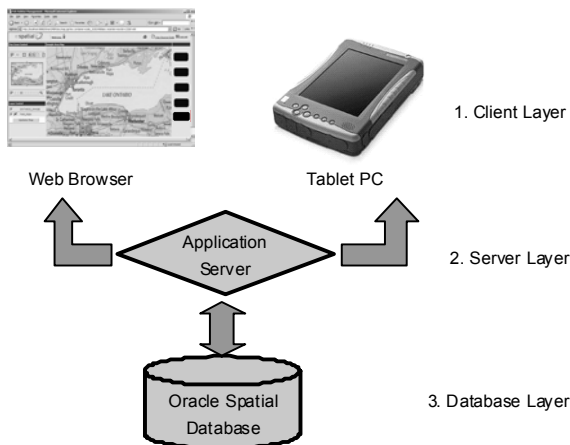


Figure 2. MEMS architecture.

2.2. Application Server Layer

The application server in Figure 2 acts as the main hub between the client and database layers of the MEMS system and consists of a J2EE Application Server made up of two main components - the *iSmartWeb Runtime* and the *iSmart Server*. These two components are used to run applications that have been developed using the *iSmart Web Designer*.

The *iSmart Web Runtime* is a component integrated with the *iSmart Server* that executes the web application. When a web page is requested, this component executes the corresponding JSP pages and returns a HTML page back to the client. Within this component, there is a Servlet called the *iSMARTWebServlet*. This Servlet acts as an interceptor to all incoming HTTP requests and uses control logic specified by the developer to decide what actions to take when, for example, a button is pressed on the application web page.

The *iSmart Server* contains all the functions and controls in the application server layer and handles all requests made to the application server automatically with the help of an XML file called *isw_application.xml*. This file is created at deployment time and is used to process requests to the application. This layer of the system is also responsible for determining device-specific capabilities. Using XML and XSLT, the *iSmart Server* adapts all returned content based on the type of device that makes the initial request. This is achieved by creating an XML file based on the client device request. An existing XSLT stylesheet is then applied to the XML file to create the device specific output (Figure 3).

2.3. Database Layer

The spatial database layer of MEMS is responsible for processing all queries, both spatial and transactional, in the system. Oracle Spatial is used as the database platform and includes the Spatial Data Option (SDO), a spatial extension to SQL introducing new spatial data capabilities, e.g. geocoding and topological queries.

Oracle Spatial provides a platform that supports a wide range of applications from automated mapping/facilities management and Geographic Information Systems (GIS), to wireless location services and location-enabled e-business. Oracle Spatial is integrated into the extensible Object Relational Database Management System (ORDBMS), which allows access to the full functionality and security of the underlying DBMS (Lopez, 2003; Sharma, 2001). Along with the database, the Oracle proprietary OC4J application server is used. The application server is a component of Oracle, and is installed automatically. The application server acts as deployment platform for Oracle applications.

This tier stores all spatial and non-spatial data together, including raster (map/remotely sensed image) data and any metadata as well as the topological properties of these data. It enables Spatial Data Types (SDT) to be inserted, stored, manipulated and queried in the database as they are represented in physical space.

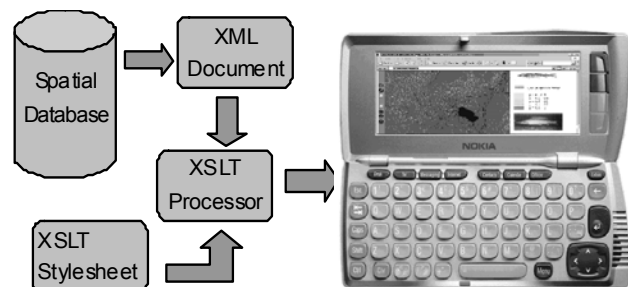


Figure 3. Device adaptation workflow using XML/XSLT.

3. System Requirements

The traditional workflow of the biologists of the DFO includes different phases, i.e. the collection phase, the input phase, analysis phase. The collection phase entails recording the data on paper in the field. Subsequently, at a later stage, the data is inserted manually into the database residing in the office. This process is time consuming and can cause inconsistencies in the data inserted due to human interpretations and typos (e.g. "n.a", "N/A" or "St.", "Saint"). In addition, the data is usually not validated, thus allowing these errors to be entered into the database.

During a field trip, biologists obtain samples from fish caught and then record the data collected. For example, they retrieve a caught fish from the storage tank in the vessel, identify it, measure the size, and record the number caught for

that particular species. This inevitability causes the biologist to navigate through many forms, editing some, until all fish are recorded.

In order to identify their requirements, there was extensive interactions with the targeted end users of our system (the biologists of the DFO). This meant participating on DFO field trips and observing their work, as well as trying to identify any additional functionality that could facilitate their work. We describe the MEMS functional requirements in the following sections both for the web-based version and for the mobile version.

3.1. Web-Based Prototype

The functional requirements for the web-based prototype are outlined based on the layer of the system with which they are associated.

Client Layer

- A real-time, geo-referenced mapping interface that enables the user to view, input and modify data as well as to annotate maps and photos in a variety of ways. The option to add, delete and query annotations will be available;
- An advanced query interface to enable the user to query the database in a number of ways (e.g., through frequently-used static queries along with the functionality to create and store additional queries);
- A query interface that enables the user to create com-

plex queries using a series of drop-down menus and also the facility for more experienced users to enter SQL to query the database directly;

- A context-aware MBR query that enables the user to query using a selected area.

Server Layer

- The ability to process form-based data and map data into the required format for use in the client layer;
- The ability to transform coordinate data from a GPS receiver into a format that can be utilized by the application;
- The ability to construct and execute all queries in the system based on the users location and requirements;
- An annotation processor that manages all annotations in the system. This is advanced functionality that requires an individual processing module.

Database Layer

- A multimedia component management module that stores and processes all the multimedia components in the database.

3.2. Mobile-Based Prototype

The following is a list of the functional requirements of the mobile prototype based on the layer with which they are associated. Note that the database layer is shared by the two versions of the system.

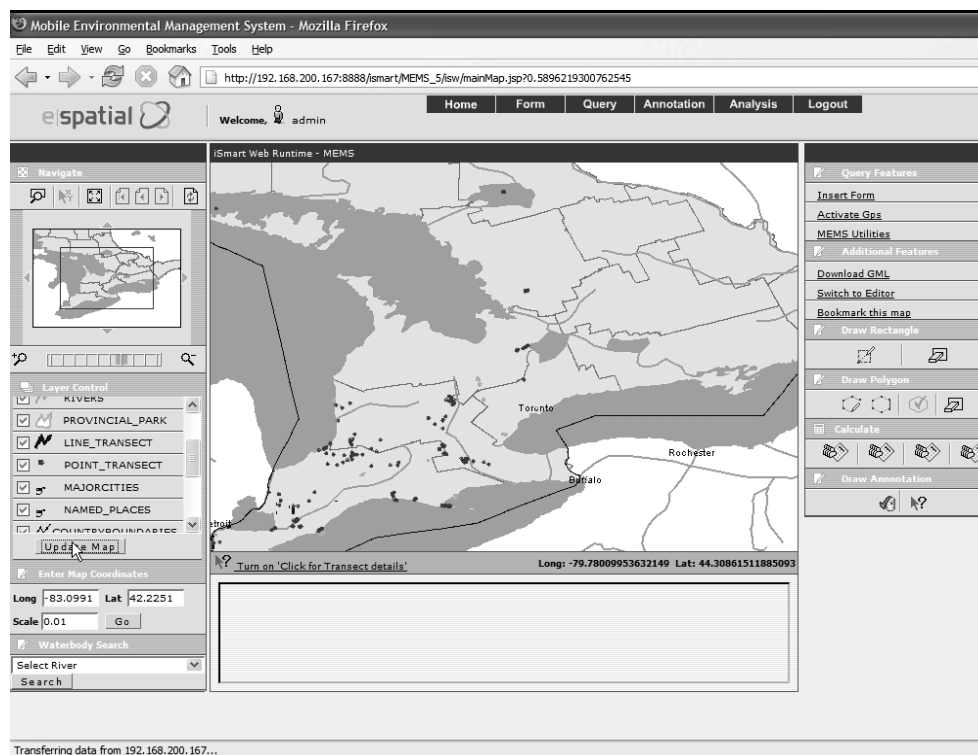


Figure 4. MEMS GUI.

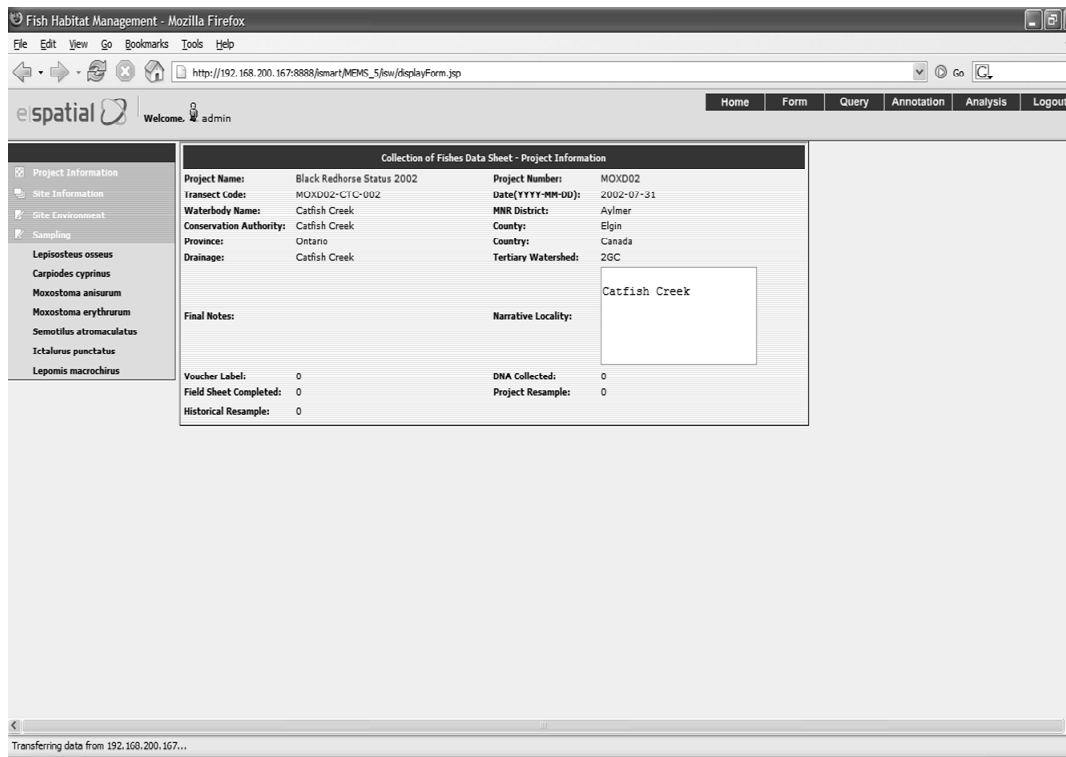


Figure 5. Data display form.

Client Layer

- A Tablet PC or PDA, used as the client; GPRS mobile phone network connection for transmitting data where a wireless connection is possible;
- A GPS receiver connected to the mobile device to determine the current position of the device.

Server Layer

- The ability to process data in online and offline modes, depending on network coverage;
- The ability to transmit data over a GPRS network to a central server;
- The ability to determine the position of the mobile device and deliver it to the application.

4. System Functionality

In this section we describe the implemented system functionality with practical examples of its use through the Graphical User Interface (GUI). The MEMS GUI is divided into three main parts (Figure 4).

The left-most panel is the Navigation panel. It contains the functions used to navigate the map (zoom to area, click x/y coordinates, zoom to extent, back to original view, previous view, refresh map). The navigation map contains a red rectangle which represents the area displayed by the main map. This is designed to help the user navigate more effectively when zoomed into a small area on a map. The Layer

Control is located below the navigation panel and is used to select layers to display on the main map. The last two controls on the navigation panel contain Map Coordinates, used to centre the main map with any given coordinates and the Search function control, used for searching previously recorded habitats.

In the centre of the application the Map Panel contains the main map. The main map offers clickable map functionality that enables the user to select features on the map and displays attribute information about them. The right-most Tools panel contains controls for functions which are explained in the following sections.

4.1. Forms Module

An important objective of our work is to minimise the work for the biologist in the office in order to maximize the hours spent recording new data in the field. To this end, the forms module of the system enables the biologists to view and input data directly while in the field thus reducing the time spent post-processing field data in the office.

Specifically, the forms module enables the biologists to display, insert and edit data using dynamic forms at different stages of the field sampling process. Validation constraints were implemented so that incorrect data is not inserted into the database (for example, drop down lists are used). Biologists can also semi-fill forms and revisit them later in the process to make changes. Figure 5 shows the Data Display Form as an example.

4.2. Utilities Module

This module hosts the following functionality.

- Name Lookup: biologists can identify different types of fish species with their common name, but they are required to record their scientific name in the form. In order to help the biologist remember the scientific names of 269 different fish, this utility offers a quick translation mechanism. The user can select the common name from a drop down menu and instantly the biologist obtains the scientific name (Figure 6).
- Transect Tool: used in order to calculate the distance that a biologist needs to travel in a particular river or water body in order to achieve a satisfactory representative sampling result.
- Accumulation Curve Tool: a graphical generation tool to display the accumulation curve for biologists. The accumulation curve is used by biologists to calculate and display how many fish have been caught for a given set of transects and for a particular species (Figure 7).

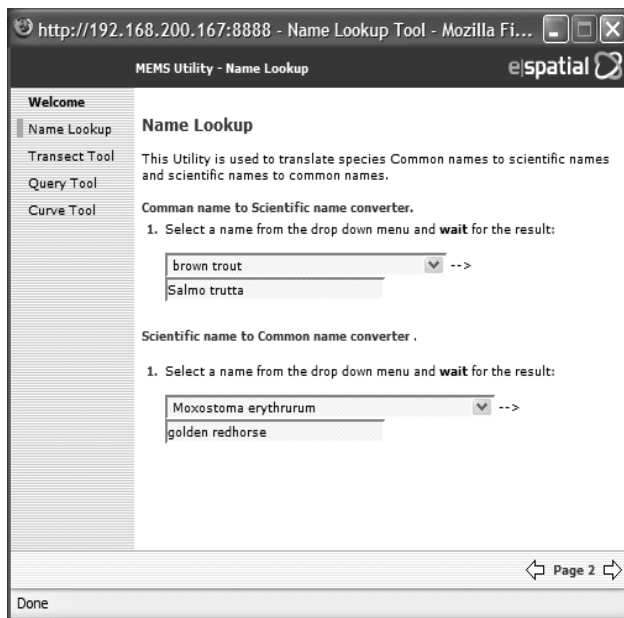


Figure 6. Name lookup.

The Accumulation Curve tool dynamically generates two possible types of graph. The bar chart is displayed by clicking the first icon on the left of the “graph toolbar”. There is also a line chart generated by clicking on the second icon on the left of the “graph toolbar”. The types of graph are illustrated in Figure 8.

4.3. Spatial Queries Tool

The Spatial Queries Tool is an advanced querying tool which enables biologists to quickly and easily query the database without requiring them to have any understanding of Oracle Spatial or the Sequential Query Language (SQL).

Users draw a polygonal area on the map on where it is then possible to execute a number of predefined spatial queries on the selected area. The polygon highlighted on the map is the actual spatial component and the highlighted buttons are the predefined queries requested by the GLLFAS biologists. The procedure starts by drawing a polygon or a rectangle on the map. The dots on the map represent locations where the biologists have previously recorded. The result of any given query is parsed and displayed in a pop-up window (Figure 9).

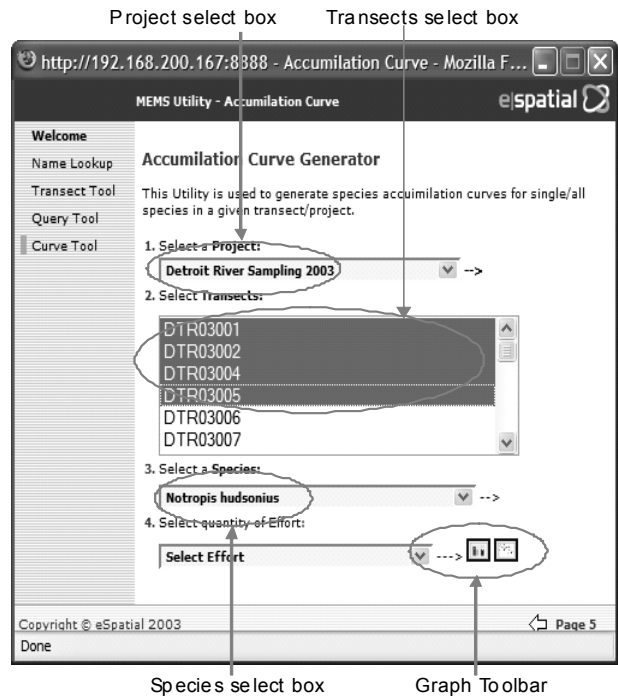


Figure 7. Accumulation curve tool.

Spatial queries are a particular type of query that use location as their primary search parameter. In the MEMS system, Oracle stores non-spatial data relating to a spatial object such as the MNRDistrict and Province of the spatial location (e.g. spawning sites). Although the application will offer advanced non-spatial query functionality, spatial queries can often give a clearer and more simplistic answer to question. Spatial queries are executed by executing Oracle SQL queries against the database and the result is returned as a hybrid ResultSet Java object called GeoResultSet. The GeoResultSet is an iSmart implementation of the ResultSet, which is capable of storing both the spatial and non-spatial attribute ResultSets. The iSmart application can subsequently extract the information from the ResultSet and displays it in a simple visual format. Currently, in the MEMS prototype, there are three main spatial queries possible. The Range or Buffer query, the Locate on Map query and the Clickable Map query. In this section a brief description of each is given.

4.3.1. Clickable Map Query

The *Clickable Map* is an extension of the normal map-

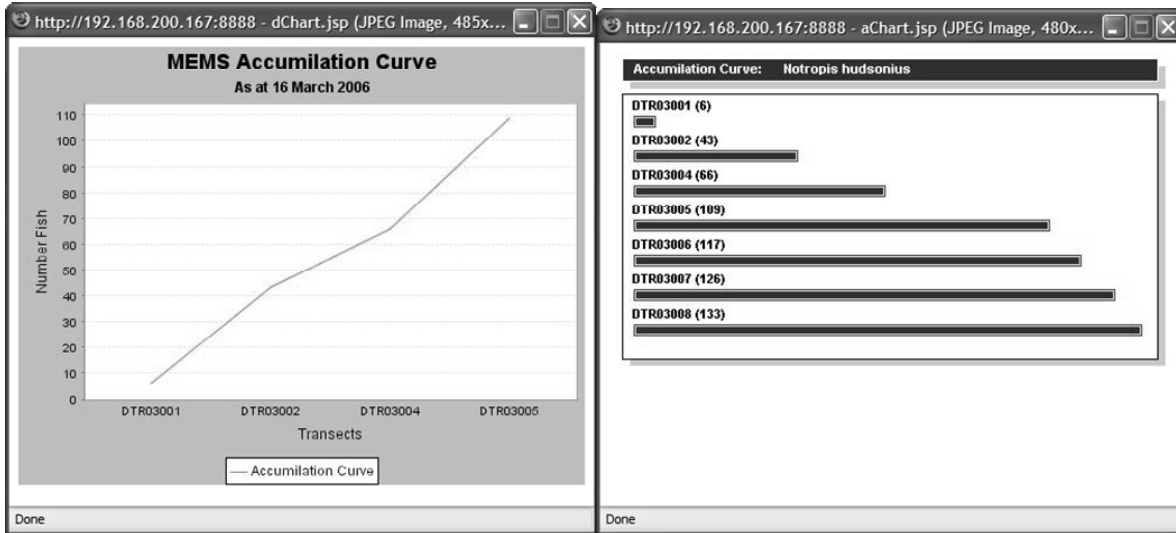


Figure 8. Graphs generated by the accumulation curve tool.

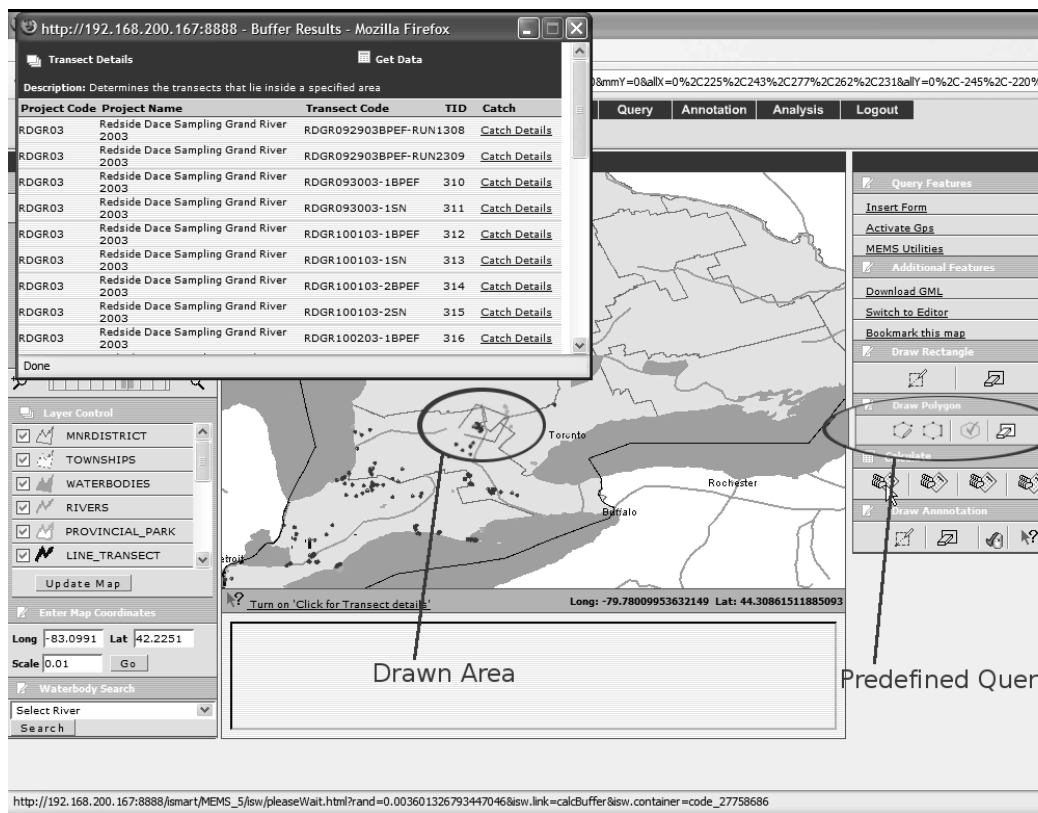


Figure 9. Spatial queries tool.

ping interface, where the particular layer to be queried needs to be specified. The purpose of this query is to enable the DFO biologists in the field to click on objects on the map interface and extract information about the object and display it on the device. This helps the biologists to get familiar with

the surrounding environment and quickly target areas of interest. The spatial query is implemented using JavaScript, JSTL (JSP Standard Tag Library), and the iSmartWeb. The map, which is displayed on the web application, is a JPEG plain image and is refreshed each time the user request additional

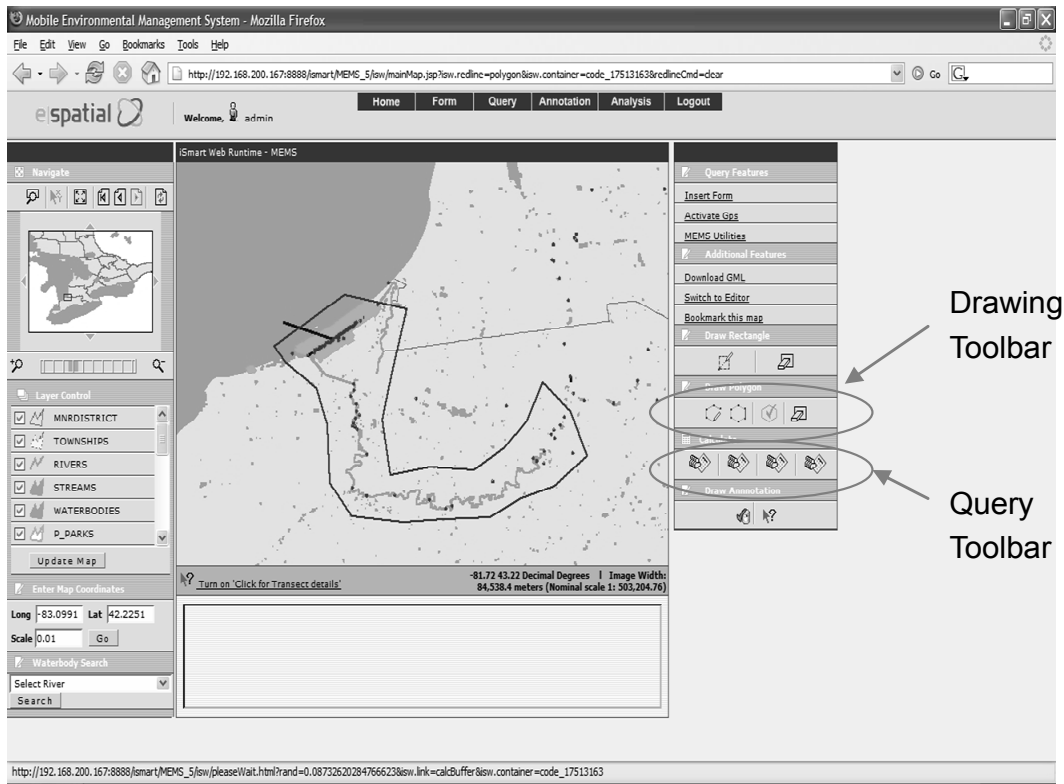


Figure 10. Drawn query.

information.

4.3.2. Locate on Map Query

The Locate on Map query enables the user to locate and display a specific transect on the map. The query consists of two parts. The first part is a text-based search where the user searches for a specific water body. The results of which are displayed with a link to the location of the water body on the map and a display link which displays more information about the transect. The hyperlinks are linked to iSmart actions, which carry out a query when selected. The locate action is the second part of the query and the result of this query centres the Viewport on the selected transect. This query is particularly useful to biologists in the field when looking for a particular site in a water body. In this way, the biologists can locate the transect, and from analysing other details about the position, determine if a sample session is required for the area or not.

4.3.3. Range Query

Up to this point, the types of queries explained are not spatial queries by definition. They are presented as spatial queries because they interact with the applications spatial component but in theory they are not spatial because the query itself does not actually interrogate the spatial component. The Range Query is a common spatial query and is extremely useful and will enable the biologists to interrogate the

database to gather information about sites within a given area. Once the Clickable Map is activated the user is given the option to search the surrounding area for sites. The user specifies a distance in meters from their current location and the query is performed. The result returned is a list of sites that are included in this area.

4.3.4. Drawn Query

This type of query is required when wanting to quickly select a custom area of interest and apply a series of pre-defined queries. The selection of the area is achieved by drawing the area out on the Map Interface. Once the area of interest is created a series of “Spatial queries” can be executed against it. The results will be parsed and displayed in appropriate windows within the web application. This module offers a significant advantage over the other query modules: the ability to effectively draw the area in which the user wants the query to be performed. If the query was to be processed without being able to draw the area of interest, the user would need to know the coordinates which make up the area of interest and feed them into the spatial query manually, making the query very impractical.

Looking at Figure 10, this module is instantiated by selecting the drawing tool which corresponds to the first button on the left of the “Drawing Toolbar”. The drawing tool uses a complex JavaScript in order to draw and record the points which constitute the polygon. Once the user is satisfied with

Transect Details

Get Data

Description: Determines the transects that lie inside a specified area

Project Code	Project Name	Transect Code	TID	Catch
AUC05	Ausable Channel 2005	AUC05053105001	1325	Catch Details
AUC05	Ausable Channel 2005	AUC05053105002	1101	Catch Details
AUC05	Ausable Channel 2005	AUC05053105003	1102	Catch Details
AUC05	Ausable Channel 2005	AUC05053105004	1103	Catch Details
AUC05	Ausable Channel 2005	AUC05060105001	1104	Catch Details
AUC05	Ausable Channel 2005	AUC05060105002	1105	Catch Details
AUC05	Ausable Channel 2005	AUC05060105003	1106	Catch Details
AUC05	Ausable Channel 2005	AUC05060105004	1326	Catch Details
AUC05	Ausable Channel 2005	OAC05062705001	1107	Catch Details
AUC05	Ausable Channel 2005	OAC05062805001	1108	Catch Details
AUC05	Ausable Channel 2005	OAC05062805002	1179	Catch Details
AUC05	Ausable Channel 2005	OAC05062805003	1109	Catch Details
AUC05	Ausable Channel 2005	OAC05062805004	1243	Catch Details
AUC05	Ausable Channel 2005	OAC05062905001	1244	Catch Details
AUC05	Ausable Channel 2005	OAC05062905002	1245	Catch Details
AUC05	Ausable Channel 2005	OAC05062905003	1246	Catch Details
AUC05	Ausable Channel 2005	OAC05080205001	1247	Catch Details
AUC05	Ausable Channel 2005	OAC05080205001	1248	Catch Details

Done

Figure 11. Popup result from the drawn query.

the polygon, s/he must press the accept button (third button from the left) on the “Drawing Toolbar”. The accept button is used to store the geometry of the polygon in the user session. The process starts by reading the points collected in the JavaScript and passing them to a server side component which creates an Oracle Geometry Object called “JGeometry”.

In order to execute the query independently of any knowledge of SQL programming, a series of spatial queries required by the GLLFAS where statically embedded in the application. These queries are shown in the “Query Toolbar” in Figure 10. One example of the query is to return the “Project Name”, “Transect Code” and “TID” for all the “Transects” in the drawn polygon. A popup window contains the results of the query and for each entry in the form there are links that generate further queries. An example of the popup windows displayed by this function is displayed in Figure 11.

4.4. GPS Module

This is a core module as the distributed MEMS application forces the client to acquire GPS coordinates first and then send them to the application server. As the client was not designed to be installation independent it is necessary to execute the Java application remotely.

One of the technologies that Java offers is JNLP (Java Network Launching Protocol). JNLP applications are launched using Java Web Start as part of the Java Runtime Environment. The corresponding GPS application registers each client IP address and host-name with the application server so that multiple feeds can be displayed on the map. The screenshot in Figure 12 shows the GPS acquisition overlaid on the MEMS map. In the top left corner the GPS module control is shown and is launched by the java Web Start while in the centre the black square is a zoomed image of the GPS feed.

This module enables the user to record location data and store it as a spatial component in the database. Once activated the GPS module opens a “GPS Toolbar” and a “GPS Application”. This Java application is used to start and stop the GPS receiver. When the GPS receiver has locked onto a steady signal the “GPS Application” displays the current “Latitude” and “Longitude” position, a green light indicating the status and also the signal strength which is the accuracy of the position in meters.

Subsequently, the current location is displayed on the map interface and is symbolized by an arrow pointing in the direction the user is moving and a trail showing where the user has been. On the tool bar the user has two buttons, the start and stop recording buttons. These buttons are used to save the start and stop position of a transect and the start and stop time for the duration of a particular sampling session (see Figure 12).

4.5. Multimedia Annotation

Multimedia annotation is another advanced feature of MEMS that allows data to be recorded on-the-fly. For example, a new species of fish could be encountered and visual evidence would be of great assistance. This functionality enables the user to embed video, audio, text and image annotations on the map. These annotations are uploaded to the database as BLOB data along with their associated coordinates (Figure 12).

Using JSP and Servlet technology, the user is required to enter some text describing the annotation. If the annotations are video, audio or image annotations a file is also required to be uploaded. When this procedure is completed the map is refreshed and an icon representing the annotation is displayed. The user can view the annotations by clicking on the corres-

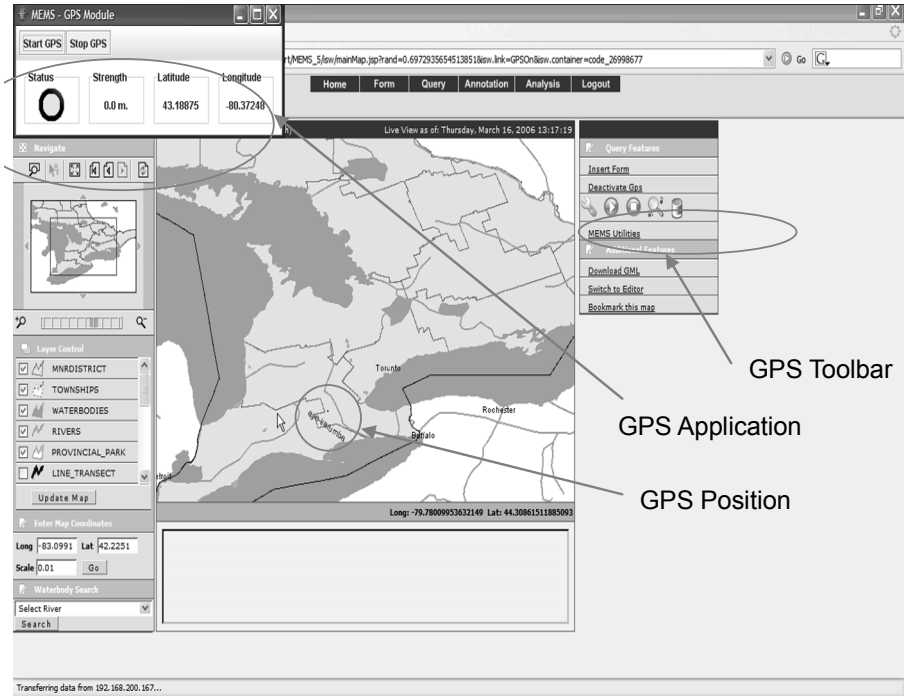


Figure 12. GPS module.

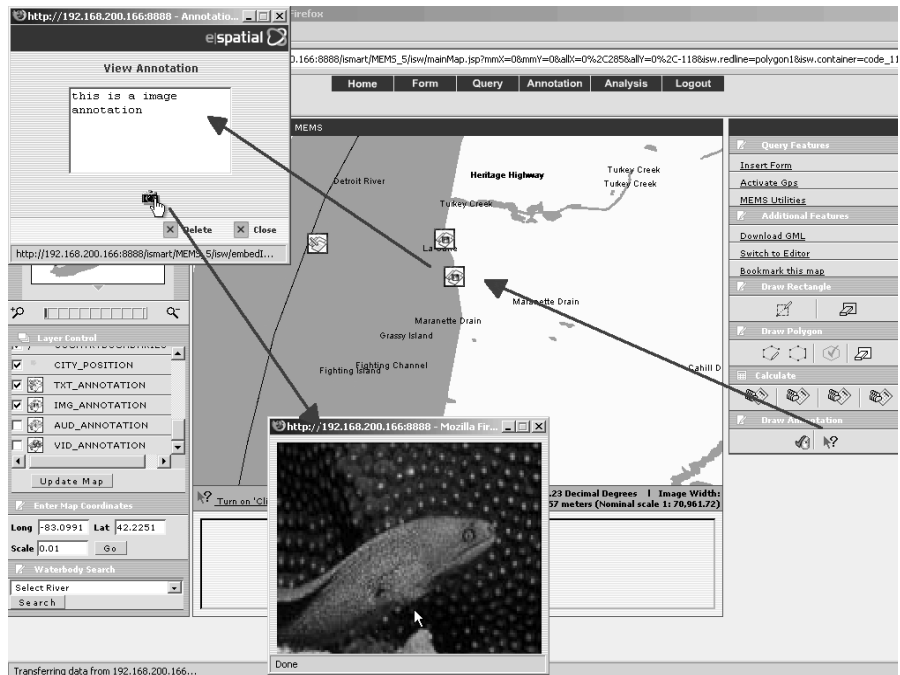


Figure 13. Multimedia annotation.

ponding icon. This action opens a popup window that displays any text associated with the annotation and a link to any multimedia data.

4.6. Offline Module

The offline module was only considered after our first

field visit to the DFO. It was observed that some areas the biologists sample in had only intermittent cellular signal, if at all. As it was difficult to detect or predict network availability before heading to the field, the offline module was designed. The module is implemented using stand-alone Java and developed using the iSmart technology. The Offline Module, also

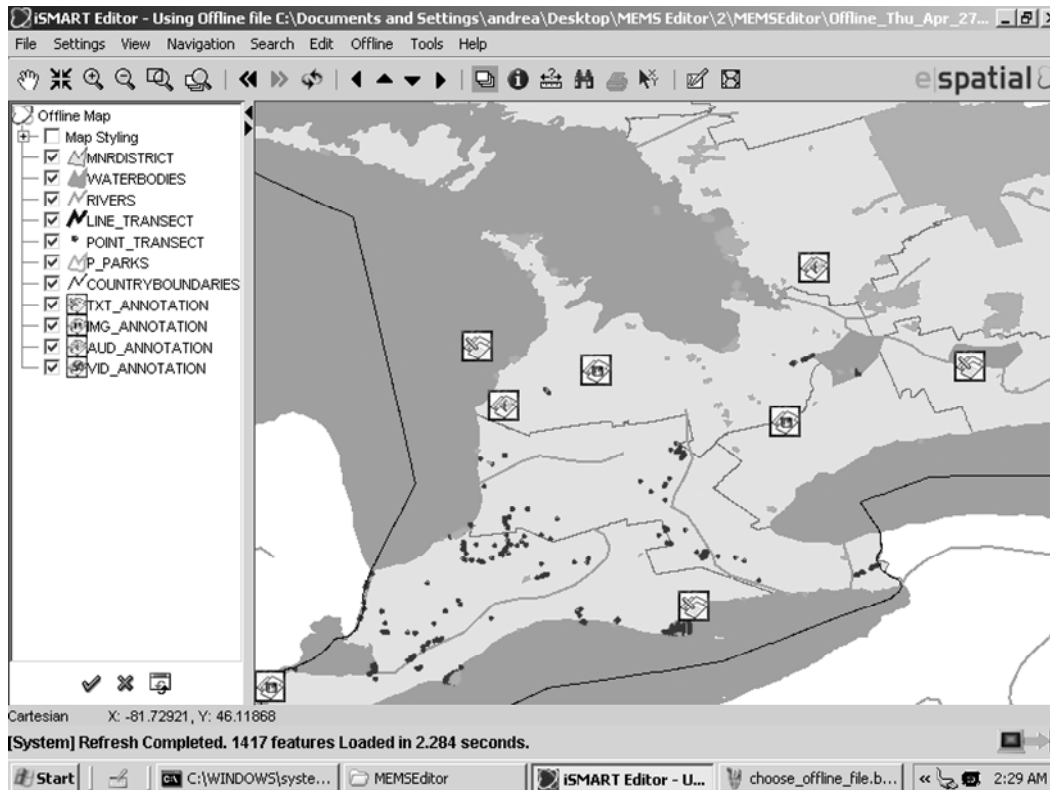


Figure 14. MEMS offline application.

called MEMSOffline, is a standalone application that connects directly to the online application server back in the office where the biologist is asked to select an area of interest. The selected area is compressed into a zip type file and saved locally on the tablet PC. The MEMSOffline compressed offline file contains all the data required by the field biologist and the application (Figure 14).

This offline application enables the user to work remotely when a GPRS connection is not available. In the field, the offline application reads in the compressed file and displays the map using the iSmart Editor application, enabling the biologists to perform spatial queries, insert new forms, look at previously recorded forms, display GPS feeds and navigate the map. It also offers restricted multimedia annotation functionality. Data which is changed or added during the sampling session is stored and saved in the offline file. When the application establishes a network connection and can re-connect to the application server, the data is synchronized with the server and any additions are added to the database.

5. Conclusions

In this paper we have described the development of a Web-based and mobile environmental management system called MEMS. Such a system was implemented to facilitate the work of biologists in the GLLFAS Fish Habitat Management Group of the Canadian Department of Fisheries and Oceans.

The MEMS system has been successfully implemented from the technical specifications design. The MEMS system relies on a Three-Tier Architecture that offers good distribution of processor load, ideal for the development of applications on low-spec devices such as PDAs, and Tablet PCs. Most of the computation is done on the Application Server Layer, and some on the Database Layer, minimizing as much as possible computation on the client layer. The result is a lightweight web application with a map interface that enables the user to display fish-sampling transect information based on a water body name, a location, or by clicking on an element on the map. The user is also able to insert new transect information using an intuitive interface.

The system includes an annotation module which enables the users to annotate points and areas on the maps and also add, edit, delete and query annotations in the database. This functionality has been fully implemented on the web-based application. A GPS module enabling the application to automatically acquire global position has also been fully implemented for the web-based application. In addition a pure Java standalone offline application was developed to mimic the functionality and the operability of the web-based application. Therefore, the same functionality of GPS module, Form Module, Annotation Module and Synchronization Module, along with all the mapping functionality is also available in the offline version and is fully operational.

MEMS is currently deployed at the GLLFAS headquarters in Burlington Ontario, where extensive field testing has

been carried out. The system offers biologists a spatially-enabled mobile and adaptable service that ensures better utilization of resources. However, the advantages of MEMS go beyond the system's functionality to improve both the biologists working environment and in-house data management through automating fisheries data entry.

Overall response from these groups was generally very favourable once the initial learning curve for a new data collection/entry methodology was overcome. However, this would not be an issue for biologists using the system daily. In terms of cost savings, as the filling out of existing field data sheets was generally mimicked on the MEMS tablet PC (i.e. the same biological measurements need to be recorded), its use in the field typically took a similar amount of time for data recording. However, the real cost savings would be given by not having to manually enter the data into the office database after returning from the field. This would save up to 3 months of time every year, depending on the number of sites sampled in the Great Lakes region every summer, and the number of biologists assigned to data entry tasks.

In the future we intend to investigate the adaptability of the MEMS architecture and GUI to different applications and departments within the DFO and elsewhere.

References

- Carswell, J.D., Gardiner K., Bertolotto M. and Mandrak N., (2004). Applications of mobile computing for fish species at risk management. Proceedings International Conference on Environmental Informatics of International Society of Environmental Information Sciences (ISEIS2004), Regina, Canada, August 2004.
- eSpatial Solutions, (2004). iSmart Web Developers Guide. <http://www.espatial.com>.
- Hinze, A. and Voisard, A., (2003). Locations and time-based information delivery in tourism. Proceedings 8th International Symposium on Advances in Spatial and Temporal Databases, Santorini Island, Greece, pp. 489-507.
- Lopez, X., (2003). Oracle Spatial and Oracle Locator, An Oracle technical white paper, Oracle Corporation, Redwood City, CA, http://www.oracle.com/technology/products/spatial/pdf/10g_spatial_locator_twp.pdf.
- Minns, C.K. (1997). Quantifying "No Net Loss" of Productivity of Fish Habitats, *Can. J. Fish. Aquat. Sci.*, 54, 2463-2473, doi:10.1139/cjfas-54-10-2463.
- Minns, C.K., (2001). Science for Freshwater Fish Habitat Management in Canada: Current Status and Future Prospects, *Aquat. Ecosyst. Health Manage.*, 4, 423-436, doi:10.1080/146349801317276099.
- Rigaux, P., Scholl, M. and Voisard, A., (2002). *Spatial Databases*, Academic Press, Morgan Kaufmann Publishers.
- Rizzini, A., Gardiner, K., Bertolotto, M. and Carswell, J., (2006). MEMS Mobile GIS: A Spatially Enabled Fish Habitat Management System. Proceedings W2GIS'06 (6th International Symposium on Web and Wireless Geographical Information Systems), Hong Kong, pp. 113-122.
- Schiller, J., (2000). *Mobile Communications*, Addison Wesley Publishers.
- Sharma, J., (2001). Oracle Spatial, an Oracle technical white paper, Oracle Corp., Redwood City, CA, USA.
- Talukder, A. and Yavagal, R., (2006). *Mobile Computing*, McGraw Hill.