Chapter XLVI

Technical Outline of a W3 Spatial (Decision Support) Prototype

João Negreiros
ISEGI-Universidade Nova de Lisboa, Portugal

Marco Painho
ISEGI-Universidade Nova de Lisboa, Portugal

Fernando J. Aguilar
Universidad de Almeria, La Canâda de San Urbano, Almeria, Spain

Manuel A. Aguilar
Universidad de Almeria, La Canâda de San Urbano, Almeria, Spain

ABSTRACT

The present research focuses on the first software to offer spatial autocorrelation and association measures, spatial exploratory tools, variography and Ordinary Kriging spatial interpolation in the World Wide Web. Exploiting IE® (Internet Explorer), ASP® (Active Server Pages), PHP® (Hypertext Preprocessor) and IIS® (Internet Information Server) capabilities, SAKWeb© (Spatial Autocorrelation and Kriging Web) was designed in an attractive and straightforward way for any GIS user. Hence, this chapter concentrates on the technical development and design of this Internet application. The differences between server and client side techniques are emphasized in the preamble section while the following one discusses the controversial debate between GIS (Geographical Information System) and SDSS (Spatial Decision Support System) concepts. The opening prospect given by the Internet platform is presented in section three. The next section fully reviews the main technological software used for its construction. References are made to their use within SAKWeb©. Some particular capabilities as an end-user were not forgotten, as well. The conclusion section leads to some future hints regarding its potential.
PREAMBLE

The death of specialized software is often cited as one of the main reasons for the lack of acceptance of spatial data analysis by empirical analysts (Anselin, 1992). Although this situation has much improved in the last fifteen years, SAKWeb© is the first Web prototype in operation that provides access to an E-Learning audience for geostatisticians at New University of Lisbon, Portugal. SAKWeb© version 2.0 is not a comprehensive statistical package in the tradition of solving everyone’s problems. Written for the Internet Information Server® (IIS) environment, it was developed with the philosophy that spatial autocorrelation and Kriging interpolation software is needed as an E-Learning tool by individuals with limited geostatistical knowledge. The incorporation of statistics to explain Earth processes (spatial statistics) has been developed furiously in the last two decades. Interpolation Kriging, the best linear unbiased estimator (BLUE) for spatial domains, is a good example. Using a LaGrangean system of linear equations where the error of prediction should be minimized in some sense, Kriging uses the covariance to measure the spatial autocorrelation among samples (including anisotropy and quadrant search) in order to estimate the value of an unknown site given the values of some other known points. In an elegant matrix layout (cf. Figure 1), each interpolated value is calculated as the sum of weighted known points whose weights are calculated from the (n+1) simultaneous linear equations set: \( A \times W = B \) or \( W = A^{-1} \times B \). The statistical distance between sample points and distances from each sample to the grid point are used to compute the model variance reproduced on matrices \( A \) (between samples) and \( B \) (between each sample and the estimated location). While \( A^{-1} \) underlies the declustering factor, \( B \) represents the structural distance between the estimation and all samples. In addition, the product of \( A^{-1} \) by \( B \) adjusts the raw inverse statistical distance weights in matrix \( B \) to account for possible redundancies between samples. As expected, if no spatial autocorrelation is found among the available samples then the Kriging estimator equals the sample average. This technique has been used in mining, hydrogeology, natural resources, remote sensing and environmental issues (Goovaerts, 1997, Zimmerman, D. et al., 1998).

In addition, it can satisfy the needs of individuals with more training. SAKWeb© deals with deterministic and stochastic interpolation in conjunction with spatial association and autocorrelation measures in a Web continuum process instead of a loose local spatial function. From this viewpoint, an element of its originality and innovation can, thus, be appreciated.

To make this project come to life, several WWW technologies were used. Active Server Pages® (ASP®), PHP® and Dreamweaver® were the main development framework in an Internet application context. WebChart®, ActiveBar®, FrontPage® Server Extensions, Flash® and JavaScript® were the other components required to accomplish this project.

Figure 1. \( \text{Cov}(x_1,y_1) \) represent the variance of sample 1, \( \text{Cov}(x_1,y_n) \) equals the covariance between sample 1 and sample \( n \), \( \text{Cov}(x_1,x_0) \) is the covariance between sample 1 and the estimated unknown site \( x_0 \), \( W_1 \) denotes the first weight while \( \Psi \) stands for the LaGrange multiplier as a result of the constraint of the weights sum to one.
At present, there are two cores Web modus operandi to build dynamic applications: client-side and server-side. The aim of any Web server is to publish HTML contents in order to reply to any request through port 80 (443, if SSL protocol is used). This type of solution is the client-side strategy where the HTML code is interpreted by the browser (cf. Figure 2). JavaScript®, Java Applets® and ActiveX® are included in this category.

If the browser contents are dynamically generated by the Web server through local executables and run-time scripts then the server-side technique was chosen (cf. Figure 3). In this domain, the use of the Common Gateway Interface (mainly written on Perl®, script shell or C language) was the first standard communication available. The most widely known CGI program regards the linkage between the submission of an HTML form (using the Get and Post methods) and a database server to save user results. Note, however, that CGI is not a language. Nor is it a program. CGI is a process, an interface which provides well-defined rules for creating partnerships (Abreu, Carreiro, 2007). The benefit is that if everyone respects the rules of this interface then everyone can talk to each other.

One alternative, especially for Netscape® servers such as iPlanet®, is Livewire®, a development environment that allows the use of JavaScript® on the server-side (Coelho, 2002).

In this new era, the present development is the server-side script languages that allow the generation of dynamic contents such as PHP®, Java Server Pages (JSP®), Java Servlets® (a server-side version of Java Applets®) and ASPs®. As expected, the use of this server-side scripting has its pros and cons, e.g.: (1) Due to every client has no processing to do, the loading of any page requires less time for the browser of the user. Yet, the responsibility of the Web server to generate the HTML must be taken into account and, hence, robust hardware for the server is mandatory. (2) Because the client only receives HTML, there are no special requirements and plug-ins for the browser. (3) The server script code cannot be viewed by the user, a situation not considered by most client-side technologies. (4) More complex solutions can be built on account of ActiveX® components, for instance.

For the majority of Web applications, the server-side versus client-side issue does not make much sense because both methods have specific

Figure 2. The client-side technology (Adaptation from http://www.homebizpal.com/technical/understanding-Web-technologies/, 2008)
and defined goals. The client-side components are used to validate and process local pages while the server-side is used to produce dynamic contents. SAKWeb© follows both trends because both technologies were applied with both purposes.

This chapter reviews these components without focusing on their integration within SAKWeb©, an Internet application of spatial analysis for GIS users. Section 2 focuses on the conceptual difference between a GIS and a SDSS (Spatial Decision Support System). Under the developer approach, major technical innovations and some functionalities regarding SAKWeb© are discussed in the following sections. The concluding section summarizes future trend of this W3 application.

**SAKWEB®: SPATIAL DECISION SUPPORT SYSTEM OR GIS?**

GIS is a system of hardware, software and liveware implemented with the aim of storing, processing, visualizing and analyzing data of a spatial nature (Painho, 1999). Pertinent data would include spatial information, such as regions location and their connectivity. Certainly, GIS directly supports the intelligence phase of spatial decision making (Ali, Sato, 2001). Three frameworks have been developed to describe this spatial decision making process: GIS software, spatial extensions of commercial statistical products and independent research applications. Unexpectedly, the Internet choice was almost never undertaken.

Globally, GIS holds two major components: data and model. The first element contains all spatial and non-spatial attributes. The model holds spatial analysis processes and other specific tools, reporting maps, particular guidance for selecting decision alternatives, problem relationships and advice in interpreting possible outcomes. What-if, goal-seeking and other types of sensitivity analyses can also be used to extend or modify the original analyses and evaluations (Guisseppi, 2003). Certainly, it is this feedback loop founded on spatial maps that increases users confidence recommendations and enable the decision maker to better explain, justify and communicate its decisions during implementation.

If a GIS holds this distinct contribution to handle spatial data, what is, then, the difference between a GIS and a Spatial Decision Support System (SDSS)? No direct answer can be given.
The simplest perspective is that a GIS is implicitly a Decision Support System (DSS) since it can be used to support decision making (Keenan, 2002). Mennecke (1997) sees SDSS as an easy-to-use subset of GIS, which incorporates facilities for manipulating and analyzing spatial data. Confirmed by Armstrong and Densham (1990), any GIS lack the modeling component needed to be accepted as a DSS. However, GIS might be regarded as a form of DSS generator (Sprague, 1980) to which models can be added to make a specific DSS (Keenan, 1996).

Clark and Hosking (1986) see spatial analysis as spatial modeling of a decision support such as GADS (Geodata Analysis and Display System) for solid waste spatial planning. In conjunction with the network and spatial analysis of GIS modules, the Decision Support System Location Planner© analyzes market saturation, retail facilities accessibility, population mobility and demand-supply prediction based on demographic and socio-economic attributes, warehouse locations, distance or travel time between sites and expenditure flows between demand and supply chains (Arentze et al., 1997). Still, according to Openshaw (1998), an emphasis of SDSS is a convenient distraction to hide a lack of the relevant GIS technology.

According to Giuseppepi (2003), if GIS becomes a major input player required for some other type of information decision maker then the GIS might be said to be acting directly as a SDSS. Yet, if the GIS focus on a specific problem with the complete interest lack of GIS features outside that domain, GIS cannot be regarded as a SDSS. Under both visions, SAKWeb© belongs to the second one.

THE INTERNET GATEWAY

During the last decade, there have been great developments with the Internet. From a technological standpoint, cooperative work, computation distribution and networking have contributed to the widespread dissemination of geographical knowledge with more intuitive, more analytical and more diverse embedding technologies (Weiss, Indurkhya 1998). From the commercial perspective, the Internet can be viewed as an extension of the traditional competitive marketplace. In effect, the Internet is capable of dramatically lowering the transaction and agency costs facing most organizations (cf. Table 1).

Equally important, Internet technology is providing the infrastructure for electronic business because its technology can also be used to make information flow seamlessly from one part of the organization to another. For instance, Internet standards can be used to link disparate systems, such as ordering and logistics tracking that previously could not communicate with each other.

In the past, the cost of comparison shopping procedure was very high, because people had to physically travel from store to store. Internet has

<table>
<thead>
<tr>
<th>E-GOODS</th>
<th>Traditional System</th>
<th>Internet</th>
<th>Percent Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airline Tickets</td>
<td>$8</td>
<td>$1</td>
<td>87%</td>
</tr>
<tr>
<td>Banking</td>
<td>$1.08</td>
<td>$0.13</td>
<td>89%</td>
</tr>
<tr>
<td>Bill Payment</td>
<td>$2.22 to $3.32</td>
<td>$0.65 to $1.1</td>
<td>71% to 67%</td>
</tr>
<tr>
<td>Term Life Insurance Policy</td>
<td>$400 to $700</td>
<td>$200 to $350</td>
<td>50%</td>
</tr>
<tr>
<td>Software</td>
<td>$15</td>
<td>$0.2 to $0.5</td>
<td>97% to 99%</td>
</tr>
</tbody>
</table>
changed this relationship. Once everyone is con-
ected electronically, information about products
and services can flow on its own directly and
instantly to consumers (Laudon, Laudon, 2003),
although this situation might create a channel
conflict with the firm’s traditional channels.
Thus, Internet shrinks information asymmetry.
In the same way, using hypermedia capabili-
ties, companies can quickly and inexpensively
provide detailed product information specific
to each customer to a very large numbers of
people simultaneously (Evans, Wurster, 2000),
the richness-reach concept.

Similarly, Internet might help business-to-
consumer (B2C) companies by providing original
products and services such as people making on-
line bids for rock concert tickets, the pure-play
business model. This includes m-commerce,
electronic payment and customer support via
e-mail. Another current trend is Web personali-
tation, the capability to present to each customer
a modified Web page based on that person’s
purchase history.

Business-to-business (B2B) generates effi-
ciencies by enabling companies to electronically
locate suppliers, solicit bids, place orders and track
shipments in transit (Laudon, Laudon, 2006). That
is, trading partners can directly communicate
with each other, bypassing intermediaries and
inefficient multilayered procedures. This means
simpler business processes, fewer employees and
much flatter organizations than in the past in a
redesigned integrated framework.

From the science point of view, the synergy
effect of the online educational material (bibli-
ographies, electronic newsgroups, W3 sites and
E-Learning software) in which the overall infor-
mation may be greater than the sum of its parts
is a conviction. Under the GIS view, one of the
greatest WWW impact is to close the software
access gap among users (“from the privilege of
the few to the right of the many”) with free and direct
retrieval of spatial analysis tools. Technically, this
implementation model is divided into three layers
(the three-tier model): interface, application and
data access (Abreu, 2006). Compared with the
conventional stand-alone and distributed two-
tier structure, Web applications emerge as a new
paradigm by integrating complex and boundless
technologies such as CGI, ISAPI (Internet Server
Application Programming Interface), Coldfusion®
(based on a set of tags, the ColdFusion Markup
Language, instead of a scripting language),
PHP®, JSP® (Java Server Pages), ASP® and ASP.
Net®. To introduce the technical infrastructure
of SAKWeb® becomes, then, the issue of the fol-
lowing section.

SAKWEB® TECHNOLOGIES

Internet Information Server®

At the heart of any Web application, a Web server
is a requirement that might includes FTP (File
Transfer Protocol), IRC (Inter Relay Chat), Mail,
News, Telnet and Proxy servers (Baptista, 2002).
IIS® was SAKWeb® choice. First and foremost,
IIS® is a protocol server that implements the
Internet protocol HTTP (Hyper Text Transfer
Protocol), among others. It also offers standard
Application Program Interfaces (APIs) for extending
and customizing its server’s capabilities. Under
Windows® environment, its management console
can be found in the Administration Tools (Ser-
rão, Marques, 2007). Regarding its architecture,
it is based on WAM (Web Application Manager)
technology with COM (Component Object Model)
and ISAPI functionalities. When IIS® receives
an HTTP request its job is to return the request
resource such as a static or dynamic page. Depend-
ing on the resource extension name, it loads the
appropriate ISAPI extension, forwards the request,
receives a return call and returns that data stream
to the requesting user’s browser. Specifically, when
an ASP® page request is received, IIS® forwards
the request to the ASP.DLL (the ASP® parser),
processes the page and sends the return data to
the Inetinfo executable process which simply returns the data to the user.

With regard to SAKWeb®, the next five items depict setup major steps: (1) The home directory is c:\inetpub\wwwroot; (2) An IP/Port address was required with a DNS (Domain Name Server) name; (3) Login.asp is its default home page; (4) The Executables folder holds execute permission while the Data folder retains write authorization; (5) All other folders hold the default right permissions for read and run scripts. Concerning these last three issues, they are quite critical because implementing security and privacy at IIS® can never be overstated. Basically, this Web server provides four types of authentication:

1. **Basic**: The logon name and password specified by the user
2. **Digest**: This sends the user’s credentials for validation in an encrypted form
3. **Integrated Windows**: It uses the account credentials of the Windows® domain to authenticate the user
4. **Anonymous**: This type uses a built-in user account to request resources from the Web server. Option one was SAKWeb® choice

To secure a virtual directory, the Web manager needs to invoke the Internet Services Manager and, after the directory in question is selected, the Security tab allows permissions to specific computers to be granted or restricted (Alonso et al., 2004). Another option is the Directory tab that allows selection of the appropriate permissions for the Web application such as Read, Write, Script source access, Directory browsing and Log visits. With regard to this last issue, IIS® provides a number of formats for the Web server log files (Stanek, 2003): (1) Microsoft® IIS® log – It records basic information about Web requests, including IP address, date, time and the number of bytes exchanged while processing the request; (2) NCSA common log – This option records fewer details of requests than the previous one; (3) W3C extend log – This format saves the most detailed

**Figure 4. The home and authentication page that includes the keywords Meta tag for search engines robots**
description available of a Web request. Option one was SAKWeb© choice. At last, SAKWeb© implements its own authentication mechanism based on the default authentication form (cf. Figure 4). If the user is not authorized, the Web application directs the user to a pre-defined logon refused page. Otherwise, he or she is redirected to the main Web page.

**From CGI to ASP® and PHP®**

With the wish to create more than just a static display for the Web, Web developers turned to CGIs and Perl® language to introduce some sort of interaction. Since CGI programs are executable applications, it is the equivalent of letting the world run a program on the system, which is not the safest thing to do (Negreiros, Ferreira, 1999). Certain security precautions are, thus, needed: CGIs need to reside in a special directory so that the Web server knows where to execute programs rather than just display it to the browser. If the Web server has an NCSA HTTPd server distribution version, for instance, then the /cgi-bin directory is the chosen one. According to Caphart (2003), this directory is usually under direct control of the Webmaster, prohibiting the average user from creating them.

With CGI applications, the client request is first sent to the server over the Internet via HTTP. The server receives it, determines which program needs to be run and writes the information to an input file. Then, the server launches the CGI program. This program reads the input file, writes the output to another one and terminates itself. The Web server process, which has been waiting for the previous steps, reads the output file and finally sends it back to the client via HTTP (Loureiro, 2005).

CGI applications are easy to write but scale very poorly within Windows® (Microsoft Learning, retrieved from http://www.microsoft.com/mspress/books/sampchap/1394a.aspx, 2008). Because a separate process is spawned for each client request, hundreds of clients create hundreds of instances of the same CGI, each requiring its own memory space and system resources. This is not such a bad thing on Unix®, which is designed to handle multiple processes with very little overhead (Morgan, 1996). However, Windows® expends more system resources when creating and destroying application instances. One way to get around this problem is to embed this processing into the Web server itself by adding logical and processing power (eXtropia, retrieved from http://www.extropia.com/tutorials/devenv/middleware.html, 2008).

Another earliest technology to take advantage of this idea was SSI (Server Side Includes). The concept of SSI is simple. Special tags that are inserted into the HTML document, understood by the Web server and translated on-the-fly by the Web server as the HTML document passes to the browser (eXtropia, retrieved from http://www.extropia.com/tutorials/devenv/middleware.html, 2008). ISAPI (requires all the programming and layout to be contained in a.dll file) came and went primarily because it required more knowledge to create a dynamic filter than Web programmers were prepared to learn (Pandey, 2002). Finally, Web scripting languages emerged including Microsoft ASP®s, a server-side scripting technology for building Web pages that are both dynamic and interactive, a major demand for SAKWeb project©.

This new technology shifted the focus from desktop to distributed computing where a number of applications are integrated to provide a mechanism solution (Mendes et al., 2004). Thus, resources at remote locations can be integrated with Exchange Server®, Internet Security® and SQL Server®. In addition, PHP® and ASP® offers an open server-application environment where the user combines HTML, server-side scripts and reusable COM server components to create a dynamic and powerful Web-based business solution (Microsoft TechNet, retrieved from http://
Technical Outline of a W3 Spatial (Decision Support) Prototype

www.microsoft.com/technet/prodtechnol/windows2000serv/reskit/iisbook/c01_active_server_pages.mspx?mfr=true, 2008). At last, after the server-side script runs, the results are returned to the client browser in the form of a standard HTML document.

This server-client framework holds remarkable programming features: (1) Response, Request, Server, Scripting, Application and Session objects that allow dynamic information access to the ASP® Object Model hierarchy. (2) The capability of including standard COM objects (.dll or .exe files) from third party companies. (3) The faculty to access internal objects such as the Dictionary (a collection of miscellaneous key/value pairs that the programmer may use within any page).

A Web application becomes, thus, a collection of ASP® pages, server components and a Website where its distinction among all applications is made by the root directory within the site. Therefore, all content within each directory structure is considered part of the scope of the same application. Moreover, each application has its own set of variables and attributes that define its current state. These are maintained throughout the application lifetime. This time concern with Web based applications is due to the fact that HTTP has no memory and retains no information from one client request to the next (Coelho, 2000). ASP® gets around this issue by using Application and Session objects to store information during a user’s session. In a further technical detail, both application and session are initialized and destroyed by the global.asa file found in the root directory of any Web application. Internally, this global.asa is an optional file that might contain declarations of objects, variables, events, library declarations and inclusion files directives (Scribd, retrieved from http://www.scribd.com/doc/98067/asp, 2008).

Regarding SAKWeb®, its code was achieved with this technology in accordance with the following hierarchical folder structure:

- **/Chat**: This directory contains all files that allow users to meet in a virtual room. It uses cookies and twelve session’s variables although its home is an HTML file.
- **/Common**: This folder includes three important scripts: (1) Lib_data_to_array.asp reads Excel® and ASCII input data to an internal dynamic array for post-processing; (2) Menudynamic.asp redirects the user to the home page or to the main menu; (3) Menudynamic.htm includes VBScript® code for the call of a third party component named ActiveBar®.
- **/Content**: This folder consists of several types of files whose aims are Excel® management, variogram fitness procedure, Flash® executables, statistical descriptive measures, indicator mapping for samples, bivariate data posting, image mapping, local interactive statistics and nearest neighborhood analysis.
- **/SK**: This folder contains all files that are responsible for Simple Kriging interpolation.
- **/Data**: It holds all background text files that are created during any SAKWeb® interpolation process by any GIS user.
- **/Executable**: It contains the gamv.for, varmap.for, kt3d.for and kb2d.for files and their GSLib® (a set of geostatistical Fortran routines developed at Stanford university) executable versions.
- **/Kriging**: The twenty-five files existing in it computes Ordinary Kriging interpolation.
- **/Moran**: This folder includes seven ASP® files responsible for dealing with the Moran I correlogram and Moran scatterplot, two spatial autocorrelation measures and particularly appreciated by any GIS user.
- **/Olectra**: It encloses seven definition files (3dconst.inc, 3dgentag.inc, 2dconst.inc, bistream.asp, error.inc, color.inc and 2dgentag.inc) that are required for any Web map displayed by this software.
Technical Outline of a W3 Spatial (Decision Support) Prototype

- /HelpSAKWeb: As expected, this directory is responsible for handling all images, Flash® and Java® Applets files need it for SAKWeb© Help, SAKWeb© News, SAKWeb© Email and other E-Learning features.

SAKWeb© also includes two other tools that use PHP® server-side technology:

- The PHP Configuration option displays SAKWeb© parameters of PHP® variables status.
- The DOS Commands (cf. Figure 5) allows the user entering any MS-DOS® instruction while the results are shown in the browser itself.

JavaScript®

JavaScript® is executed when a document is loaded by the client. Often used to create dynamic HTML documents, JavaScript® increases the aesthetics and friendliness of Websites by adding events to static pages, referenced by the HTML tag `<script language="JavaScript" src="corefunctions.js"></script>`. Checking input forms and computing client-side mathematical calculus are two critical capabilities of this language. Another potential of JavaScript® (used within SAKWeb®) is the possibility to control the browser itself through the following objects and methods:

The history object allows the local accesses accomplished by the user. For instance, history.go(-1); is equivalent to pressing the Back button of the browser while history.toString(); displays an HTML table with the history of the browser links.

The document object allows access to the properties of the Web page when it is loaded. For instance, document.bgColor="FF0000"; sets up the background color to red while document.clear(); clears the content of the active Web page.

The location object provides information regarding the current URL. For instance, loca-

Figure 5. Snapshot of the DOS Commands option

<table>
<thead>
<tr>
<th>Type a DOS command:</th>
<th>Submit</th>
<th>Reset</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATTRIB filename</td>
<td>Displays or changes file attributes.</td>
<td></td>
</tr>
<tr>
<td>CHCP</td>
<td>Displays or sets the active code page number.</td>
<td></td>
</tr>
<tr>
<td>CHKDSK drive</td>
<td>Checks a disk and displays a status report.</td>
<td></td>
</tr>
<tr>
<td>CHKNTFS drive</td>
<td>Displays or modifies the checking of disk at boot time.</td>
<td></td>
</tr>
<tr>
<td>COPY file1 file2</td>
<td>Copies one or more files to another location.</td>
<td></td>
</tr>
<tr>
<td>DEL file1</td>
<td>Deletes one or more files.</td>
<td></td>
</tr>
<tr>
<td>DIR drive:directory</td>
<td>Displays a list of files and subdirectories in a directory.</td>
<td></td>
</tr>
<tr>
<td>FIND &quot;string&quot; filename</td>
<td>Searches for a text string in a file or files.</td>
<td></td>
</tr>
<tr>
<td>HELP command</td>
<td>Provides Help information for Windows commands.</td>
<td></td>
</tr>
<tr>
<td>MKDIR new_dir</td>
<td>Creates a directory.</td>
<td></td>
</tr>
<tr>
<td>PATH</td>
<td>Displays or sets a search path for executable files.</td>
<td></td>
</tr>
<tr>
<td>PRINT filename</td>
<td>Prints a text file.</td>
<td></td>
</tr>
<tr>
<td>REN file1 file2</td>
<td>Renames a file or files.</td>
<td></td>
</tr>
<tr>
<td>RMDIR directory</td>
<td>Removes a directory.</td>
<td></td>
</tr>
<tr>
<td>SET</td>
<td>Displays, sets, or removes Windows environment variable.</td>
<td></td>
</tr>
<tr>
<td>START</td>
<td>Starts a separate window to run a specified program or command.</td>
<td></td>
</tr>
<tr>
<td>TREE drive:directory</td>
<td>Graphically displays the directory structure of a drive or path.</td>
<td></td>
</tr>
<tr>
<td>TYPE filename</td>
<td>Displays the contents of a text file.</td>
<td></td>
</tr>
<tr>
<td>VER</td>
<td>Displays the Windows version.</td>
<td></td>
</tr>
<tr>
<td>VERIFY ON or OFF</td>
<td>Tells Windows whether to verify that your files are written correctly.</td>
<td></td>
</tr>
<tr>
<td>VOL drive:</td>
<td>Displays a disk volume label and serial number.</td>
<td></td>
</tr>
</tbody>
</table>
tion.reload(); refreshes the current document while location.hostname; displays the name of the remote host.

The window object includes the document and other objects that handle any Internet page internal structure. For instance, window.open("http://www.nasa.gov") loads the home page of NASA into a new window while window.alert("Your input value is out of range!"); generates a pop-up alert message whose content is its input parameter.

JavaScript® 1.2 also introduces two new functions that are very useful for any programmer, concerning debugging and error corrections. The watch() method is applied to a certain variable and, if something has changed, a particular procedure can be triggered. For instance, if the instruction watch('x',change_var); is setup then an alert message will be displayed every time the x variable changes its content. As expected, the unwatch() method turns off this debugging effect. It is essential to state that JavaScript® cannot create files or establish network connections with other remote hosts.

As a live example of this technology use within SAKWeb®, the following code illustrates how JavaScript® modifies the browser status bar (cf. Figure 6). At first, the Javascript function Date() retrieves from the client the date and time to hoje variable, whose content is displayed on the left bottom of the browser. This process continuous indefinitely due to the setTimeout() procedure that calls the function relogio() recursively every second.

### WebChart® and ActiveBar®
#### ActiveX Components

ActiveX® (formerly known as OLE Controls or OCXs) is a reusable software module based on Microsoft’s Component Object Model (COM) architecture and providing similar functions (animations, spreadsheet emulation, graphics generation…) to Java Applets. This is the technology used by SAKWeb® to display all graphics and maps. Over the Internet, ActiveX® controls can be linked to a Web page and downloaded by a compliant Web browser as if the program were launched from a Web server (cf. Figure 7). Still, like any executable program running in the computer, ActiveX® controls can perform any operation on your data. This is why the default configuration in most Web browsers is to prompt the user, if an ActiveX® control is being requested, so he/she can decide to download it or not (PC Magazine, retrieved from http://www.pcmag.com/encyclopedia_term/0,2542,t=ActiveX+control&i=37472,00.asp, 2008).

VBScript® allows developers to include ActiveX® controls that are then loaded and registered

---

**Figure 6. The JavaScript® clock output**

```html
<html>
<head>
<title>Status Clock</title>
</head>
<body>
<script>
function relogio()
{
  var hoje = new Date();
  window.status = hoje.toString();
  setTimeout("relogio()",1000); }
relogio();
</script>
</body></html>
```
into the user’s system. Yet, browser compatibility is an issue: ActiveX® technology is not currently supported by some browsers such as Firefox® or Safari®. Therefore, SAKWeb® requires IE®.

With SAKWeb® hosting system, two ActiveX® names were registered from ComponentOne®. The following HTML page (cf. Table 2) includes an ActiveBar® ActiveX® interface that communicates with the local user by exposing properties and methods to fire events. This also includes an assigned CLSID for an entry into the Windows® registry that allows the client browser to obtain, register and load the control.

Glancing at previous code, the client-side ActiveBar® interface was included to layout the output presented in Figure 8 which has a unique identifier, the 128-bit CLSID. It is the use of this CLSID that avoids the possibility of name collisions among Windows classes because CLSIDs are in no way connected to the names used in the underlying implementation. Hence, no two components with the same interface can be mistakenly used for each other (Powers, 2001).

The 2D/3D WebChart® ActiveX® gives power to this ASP® environment by generating different graphic charts. Internally, the server-side component shares a common API with the client-side that generates an .OC2 or .OC3 control to be passed to the client with the appropriate HTML tags. If the client does not hold the olch2x8.cab and olch3x8.cab files, the Web server generates a runtime-only copy of WebChart® on the client’s machine. As expected, a runtime license for the client’s computer is created if and only if there is a registered version on the developer’s computer (Côrtes, 2000). The generation of the HTML tags that supports this Web functionality is contained in two routines under the scope of 2dgentag.inc and 3dgentag.inc files: OlectraChart2D_GenerateTag_Control() and OlectraChart3D_GenerateTag_Control(). These are the procedures that allow developers to customize the way that all tags will be generated.

The next ASP® code introduces the generation of the linear trend surface (cf. figure 9), generated from a set of (x,y,z) observations, with the following features: (1) Data values are stored in d(30,30) array; (2) The chart size was setup to 400×300 pixels; (3) The chart type is a 3D bar type; (4) The background color is white while the chart axis becomes blue; (5) The shading surface and zone colors are also included; (6) The IsBatched=true property does not allow changes to the chart while other calculations are being executed; (7) No captions are shown; (8) The Call OlectraChart3D_GenerateTag_JPEG(Chart3D,"Chart3D") instruction is responsible to spawn the chart image to the client.

**DISTINCT FUNCTIONALITIES OF SAKWEB®**

SAKWeb® is not a full geostatistical, autoregressive regression and simulation software package, yet. Its present functionality was depicted in

---

*Figure 7. ActiveX® controls are stored remotely but they are run locally (Adaptation from PC Magazine, http://www.pcmag.com/encyclopedia_term/0,2542,t=ActiveX+control&i=37472,00.asp, 2008)*
Table 2. Snapshot code of the ActiveBar® inclusion within VBScript®

```
<HTML><head><title>Main Menu</title></head>
<OBJECT classid="CLSID:5220CB21-C88D-11CF-B347-00AA00A28331" VIEWASTEXT>=<PARAM NAME="LPKPath" VALUE="ActiveBar.lpk"></OBJECT>
<OBJECT classid="CLSID:E4F874A0-56ED-11D0-9C43-00A0C90F29FC" codeBase=actbar.cab#Version=1,0,6,5 id=ActiveBar1 style="LEFT: 0px; TOP: 0px" width=32 height=32 VIEWASTEXT><param name="_ExtentX" value="847"><param name="_ExtentY" value="847"></OBJECT>
<body><script language="VBScript">
<!--
 Sub window_onLoad()
 ActiveBar1.Attach
 Create_Tools
 Create_Bands
 end sub
 Sub Create_Tools()
 iCat = 100
 Set Tool = ActiveBar1.Tools.Add(iCat + 1, "mnu1")
 Tool.Caption = "&Data View and Tools"
 ...
 end sub
 Sub Create_Bands()
 Set b = ActiveBar1.Bands.Add("mnu1")
 b.Type = 2 ' dBTPopup
 ...
 ActiveBar1.Bands("mnuMain").GrabHandleStyle = 0
 ActiveBar1.BackColor = 33792
 ActiveBar1.ForeColor = 16777215
 ActiveBar1.MenuFontStyle = 1 ' ddMSCustom
 ActiveBar1.Font.Name = "Arial"
 ActiveBar1.Font.Bold = True
 ActiveBar1.Font.Size = 10
 ActiveBar1.RecalcLayout
 ActiveBar1.Refresh
 End Sub
 Sub ActiveBar1_DataReady()
 ActiveBar1.RecalcLayout
 end sub
 ...-->
</script></body></HTML>
```

Figure 10. Still, some distinct capabilities will be reviewed here while major features, under the interaction scenario view point of the end-user, can be appraised in Negreiros et al. (2006, 2008). The control management of the site itself by the local administrator can be achieved by means of three options: (1) DOS Commands - The capability to browse the contents of MS-DOS® commands. (2) PHP Configuration - The ability to display all PHP® configuration of SAKWeb©. (3) Session Variables – It gives the CGI parameters and the contents of the session variables created by the current user session (a useful tool for the developer to control the application flowchart). As well, SAKWeb© may plot a horizontal 3D graphic with the visitor distribution to the site throughout the day by dividing it into six groups of hours (cf. Figure 10). Its aim is to give clues to geostatisticians to what is the best time to connect SAKWeb©.
When spatial analysis is at stake, the possibility for sampling location is achieved with the 3D Data Posting (cf. Figure 11). This tridimensional view includes drop lines and X-Y-Z axis grid lines options. Further, according to samples values, data observations are plotted with three different colors: below 25% of the average, between 25% and 75% and above 75%.

Ordinary Kriging (OK) was the stochastic interpolation method chosen by SAKWeb©. This probabilistic approach tries to find the weighted average estimator that provides an unbiased estimator with the smallest estimation error variance using the LaGrange multiplier and the weights sum to one constraint. That is the reason why Kriging is BLUE (best linear unbiased estimator) and BUE (best unbiased estimator if input data respects the Normal curve). Still, it is the variogram (cf. Figure 12) that underpins Kriging by summarizing the degree of similarity between values for all possible pairs as a distance function. Four major factors embody this spatial autocorrelation measure: (1) Sill – The variogram value that implies no spatial dependence between data points because all variances are invariant with the sample separation distance; (2) Range – The separation distance at which samples are spatially autocorrelated; (3) Anisotropy – It represents the spatial autocorrelation behavior of the variogram according to several directions; (4) Nugget-effect – It embodies the measurement error variance and the spatial variation at distances much shorter than sample spacing, which cannot be resolved (Liebhold, 1995).

According to Armstrong (1998), the nugget-effect (C0) is the most significant and difficult issue to setup the variogram. Hence, three OK approaches are computed by emphasizing this factor: OK with C0, OK without C0 and OK with C0 plus two structures. The capability to rotate, to zoom, to reset and to join the 2D and 3D surfaces together in a W3 environment is a remarkable step for future spatial data analysis software, as well. This advent is a result of the kb2d.exe GSLib® routine that assesses these three OK versions through a PHP® file that uses cookies technology. Cookies are small text files saved by the user browser through the header of.
Figure 9. The SAKWeb© trend surface analysis generated by the previous code

```vbscript
Set Chart3D = Server.CreateObject("C1Chart3D8.ASPComponent")
With Chart3D.ChartGroups(1)
    With .ElevationData
        .ColumnCount = 30
        .RowCount = 30
        For i = 1 To 30
            For j = 1 To 30
                .Value(i, j) = d(i,j)
            Next
        Next
    End With
    With Chart3D
        .Width = 400
        .Height = 300
        .IsBatched = True
        .ChartGroups(1).ChartType = oc3dTypeBar
        .Interior.BackgroundColor = ocColorWhite
        .Interior.ForegroundColor = ocColorBlue
        With .ChartGroups(1)
            .Styles(1).Symbol.Size = 2
            With .Contour
                .Levels.NumLevels = 20
                .IsZoned = True
            End With
            .Elevation.IsShaded = True
        End With
        .Legend.IsShowing = False
        .IsBatched = False
    End With
With Chart3D
    Response.Write("<p align=center>")
    dispChart = Session("DispChart")
    If dispChart = "Jpeg" Then
        Call OlectraChart3D_GenerateTag_JPEG(Chart3D,"Chart3D")
    ElseIf dispChart = "Png" Then
        Call OlectraChart3D_GenerateTag_PNG(Chart3D,"Chart3D")
    ElseIf dispChart = "Png-BinaryWrite" Then
        Call OlectraChart3D_GenerateTag_PNG_BinaryWrite(Chart3D,"Chart3D")
    ElseIf dispChart = "Jpeg-BinaryWrite" Then
        Call OlectraChart3D_GenerateTag_JPEG_BinaryWrite(Chart3D,"Chart3D")
    Else
        Call OlectraChart3D_GenerateTag(Chart3D,"Chart3D")
    End If
    Response.Write("</p>")
End With
Set Chart3D = Nothing
```
Technical Outline of a W3 Spatial (Decision Support) Prototype

Figure 10. The visitor distribution graphic of SAKWeb©

<table>
<thead>
<tr>
<th>GMT Standard Time - Greenwich Mean Time (Dublin, Lisbon, London)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.65% (0 and 4h)</td>
</tr>
<tr>
<td>3.55% (4 and 8h)</td>
</tr>
<tr>
<td>17.59% (8 and 12h)</td>
</tr>
<tr>
<td>24.13% (12 and 16h)</td>
</tr>
<tr>
<td>9.41% (16 and 20h)</td>
</tr>
<tr>
<td>37.66% (20 and 24h)</td>
</tr>
<tr>
<td>100% (reference value)</td>
</tr>
</tbody>
</table>

Total Visitors: 3548 (100%)

Figure 11. The 3D scatter view of SAKWeb©
an HTML file. Since ASP® and PHP® are able to write (response.write("sakWebsessionid")) and read ($_cookie["sakWebsessionid"])) them, SAKWeb© creates a cookie called sakWebsessionid which contains the user session identification (response.cookies ("sakWebsessionid")). Therefore, the possibility of sharing this crucial information between both scripting technologies becomes real and central for this intercommunication process.

Another motivating option for any geostatistician is the Exact OK Differences option that displays the interpolation differences among the three previous nugget-effect approaches (cf. Figure 13). This includes the computation of the following statistics: average difference, maximum and minimum disparity between approaches and average OK variance.

A weighted average of the previous three models (cf. Figure 14) is also a possibility with this software whose weights are based on the variance of the Kriged estimates (more weight is given to those values with smaller estimation variance).

Flash® MX is the SAKWeb© Help standard authoring tool. One of its achievements relies on its success in capturing the feel of a living, thinking and interactive solution without being distracting, confusing and gimmicky. Distinctive animation for each section has also been created. For instance, the navigation system is focused in a sensitive way, with every button having a normal, rollover and active state. In fact, any interaction model should pursue a consistent navigation scheme that allows users to understand what a button is without having to think about it. With yahoo.com, for example, its common interaction model is represented by underlined text links (Street, 2002). With SAKWeb© help, text link buttons are angular and highlighted when the cursor rolls over them. Color consistency and acid audio in a loop context completes this environment.

**FINAL THOUGHTS**

There is an increased demand for systems that do more than display spatial data (Ebdon, 1998). Spatial data holds special features to the researcher: Where does this occur? How does this pattern vary across the study area? How does an event at this location affect surrounding locations? Do areas with high rates of one variable also have high

*Figure 12. The typical variogram shape with its main parameters (Ouyang, Zhang, Ou, 2005)*
Figure 13. The faculty to compare on-the-fly three nugget-effect approaches is a welcome possibility

Figure 14. Combination of Kriging estimates from OK with C0 (model 1), OK without C0 (model 2) and with two structures (model 3)
rates of another? Traditional statistical techniques tend to produce a summary statistic that quantifies the strength of a relationship within a dataset, for example. This approach is undesirable, from a GIS perspective, because it ignores the impact of space. It is important that spatial methods should explicitly incorporate the spatial component to develop a more sophisticated understanding from our data (Negreiros, Painho, Aguilar, 2008).

With the advent of Web technology and modern wireless computing, it has also become necessary to develop a WWW service for GIS interpolation in order to understand the often complex spatial autocorrelation that exist among observations collected in space. The inclusion of both topics in an E-Learning context provided major inspirations for SAKWeb©. Quite often, common users request spatial analysis knowledge in a self-learning view because of global cost reduction, both time and money. In addition, implementing the technological structure that supports E-learning platform is a scalable solution (Painho et al., 2002). Spatial analysis wizards, multimedia tools (including animation and hyperlinks), on-line help, software courses, videos, E-Learning and M-Learning with WML technology are ingredients of this demand. CRM of the Universidade Autónoma de Lisboa and GIS&Sc Master of ISEG, certified by UNIGIS, are some examples of the Learning Space strategy (Semana Informática, 2002). The International Center for Distance Learning (http://www-icdl.open.ac.uk), the AT&T Learning Network Virtual Academy (http://www.att.com/learningnetwork/virtualacademy), La Escuela de Negocios a Distancia de la Universidade Politec-nica de Madrid (http://www.cepade.es), Le Centre National d’Enseignement à Distance (http://www.cned.fr) and the Universidade Aberta (http://www.univ-ab.pt) are others fine examples of this trend (Negreiros and Painho, 2006).

Although this project is still a work in progress, the future of SAKWeb© can be bright. It is expected to be launched in 2009 as myGeooffice.org. Its infrastructure can be applied easily as a WWW interface with GSLib® routines to avoid the reinvention of the wheel for other geostatistical algorithms. This includes UK (Universal Kriging), IK (Indicator Kriging), CK (CoKriging) and Monte Carlo simulation. Morphological geostatistics, cost environmental analysis, K-means clustering, autoregressive regressions and Geary C computation will not be forgotten, as well. Adapting Anselin (1992) software contemplation, all academic research implementation should be a reality in a W3 environment.

REFERENCES


Technical Outline of a W3 Spatial (Decision Support) Prototype


Street, S. (2002). Flash® for the real world: E-commerce case studies. SAMS.


KEY TERMS AND DEFINITIONS

E-Learning: A broad set of applications and processes which include Web-based and computer-based learning, virtual classrooms and digital information. In companies, it refers to the strategies that use the company network to deliver training courses to employees. Lately in most Universities, it is used to define a specific mode to attend a course of study where the students rarely attend the face-to-face traditional classes room because they study online.

Geographical Information Systems (GIS): System of hardware and software used for storage, retrieval, mapping and analysis of geographic data. In its strictest sense, it is any information system capable of integrating, storing, editing, analyzing, sharing and displaying geographically referenced information. In a more generic sense, GIS applications are tools that allow users to create interactive queries, analyze spatial information and present the results of all these operations in maps.

Kriging: A form of statistical modeling that interpolates data from a known set of sample points to a continuous surface. It is the best linear unbiased predictor whether or not data are normally distributed. It is linear since estimations are a weighted linear combination of the available data. It is unbiased because the error mean is zero (no over or under-estimates). It is best since its goal is to minimize error variance.
myGeooffice®: The future marketable name of SAKWeb® that includes other forms of Kriging and spatial autocorrelation measures such as Geary index. Geostatistical simulation, cost analysis and morphological issues will be also covered.

SAKWeb®: It focuses on the first software to offer spatial autocorrelation and association measures, spatial exploratory tools, variography and Ordinary Kriging (OK) in the World Wide Web. In terms of implementation technologies, several different software were used: ASP®, IIS® with Server Extensions, PHP®, FrontPage®, VBScript®, ActiveX®, Dreamweaver®, Ultradev®, Flash®, Director®, Fireworks®, WebChart®, ActiveBar®, Java Applets®, JavaScript®, HTML, DHTML, Fortran and C language.

Spatial Autocorrelation: The degree to which a set of features tend to be clustered together (positive spatial autocorrelation) or be evenly dispersed (negative spatial autocorrelation) over the Earth's surface. As in the data mining process of finding attribute anomalies, spatial autocorrelation measurements look for patterns and relationships within vast spatial digital archives. These indices are categorized into two groups: Distance view and neighboring view.

Spatial Decision Support System: A customized computer-based information system that utilizes decision rules and models and incorporates spatial data. It is designed to assist the spatial planner with guidance in making land use decisions, for instance.