

Cartography, millennium-proof

Visualisation, a production-oriented approach

NOWADAYS, CARTOGRAPHY IS undergoing unprecedented change driven by technological developments in the fields of both computers and telecommunications, posing new challenges for cartography. These challenges are threefold: formalisation, communication, and cognition and analysis. Cartographic visualisation will have importance both to the analytical and cognitive function of cartography, as well as to the communication function and should stimulate the whole field of cartographic design (Taylor, 1991, p.15).

Formalisation refers to the approach in cartography, which sees the main purpose of the discipline and profession in terms of cartographic production (Taylor, 1994, p.334). Developments in the field of computers, telecommunications, and multimedia are reshaping this aspect of cartography dramatically. A key element in the convergence of these technologies is the creation of “information highways”, networked computer environments. The Web provides a new medium for cartographic communication to which cartographic production has to adapt itself.

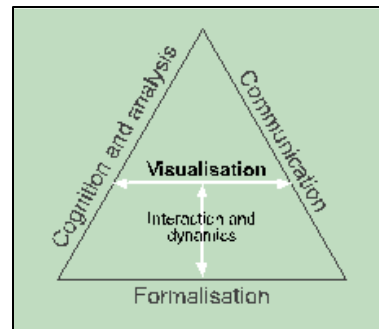


Figure 1. Conceptual basis for cartography

Views concerning cartography and the Web have developed along with the growing popularity of the Internet. In 1985 Monmonnier acknowledged, that “... *information networks provided through improved telecommunications would influence substantially the abundance, availability, timeliness and contents of maps.*” However, at that time he did not grasp its full implications, for he remarked that “*although maps clearly have a role in the telecommunications-information revolution, the role is clearly secondary and more passive than active*” (Monmonnier, 1985, p.178). Nowadays, maps have become an integral part of disseminating information over the Web. Cartography can play an important role in this new medium, for the visual communication of (geospatial) information has always been its central field of interest. In the Netherlands, the first reference of cartography linked to the Web was not until 1996 (Bakker & Schokker, 1996, p.29-43). Nowadays, this relationship has established itself firmly in the Dutch cartographic community. Web cartography is even the main focus of research of the Division of Geoinformatics, Cartography and Visualisation at ITC Enschede.

The link between cartography and the Internet is seen by Kuunders (Kuunders, 1998b, p.32-3) as one of the four trends in the development of cartography. Another trend important to the scope of this research is the link between cartography and GIS. Central to this link is the “... *geospatially related information, normally in graphic form.*” This is the map, whether it is on screen or on paper and whether it is used as a source for input or as a method for representation (Lee, 1995, p.33).

The research described in this thesis concentrates on the central overlap in Kuunder's model, referring to the fourth trend that he envisions for cartography: the maximum combination of the aforementioned trends. It focuses on the role of cartography in WebGIS software. This is a new production tool for cartography to communicate visually geographic information across the Web. However, what are the new possibilities of these production tools? Do they impose new limitations? What is the influence of the characteristics of the Web on cartographic design?

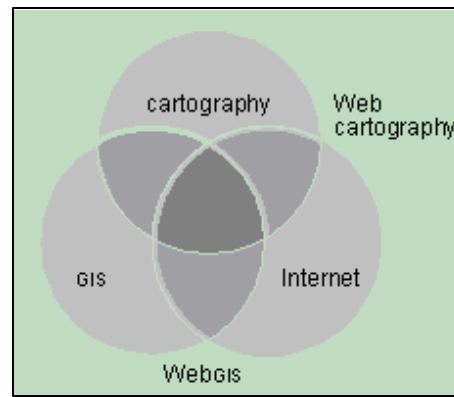


Figure 2. Cartography, GIS and the Internet

Communication

One of the characteristics of the Web is the diversity of data. The volume of data increases every day. Furthermore, there are no organising principles for the provision of these data. The increasing volume of data and its diversity necessitates conversion of this data into useful information to facilitate communication. Communication has always been a key element to cartography. The map and related cartographic products are ideal media for the effective presentation and communication of geographic information in a wide variety of subject areas (Taylor, 1994, p.337). This is even more apparent for the communication of geospatial information that cannot be adequately conveyed by tables of text and numbers alone (Lee, 1995, p.33). The Web provides a visualisation environment with a pure multimedia character. Multimedia changes the way people look at maps in a GIS. This multimedia technology helps people becoming more productive (Huber, 1994, p.52).

The degree of interaction between the user and the cartographic product is crucial to learning and communication (Taylor, 1994, p.338). The Web provides a medium for the display of these multimedia elements. The development of new Internet technologies, especially in the field of scripting and plug-in technology extends the multimedia character of the Web even more, increasing the interaction between clients and the geospatial data. The Web has truly become a multimedia environment for communication, where cartography can play an important role.

With the introduction of distributed geographic information system technology and the establishment of Geospatial Data Infrastructures, even more geospatial data has become available across the Web. Cartographic expertise is necessary to convey the meaning of the geospatial data to the client. These tools enable interaction with the geospatial data to a certain degree as well, by extending the browser capabilities in one way or another. Communication of geospatial information across the Web will therefore be more efficient than with just static view-only map images. WebGIS may be able to provide Webmapping application services that go beyond the provision of static view-only maps, allowing for interaction and taking full advantage of the multimedia character of the Web.

Cognition and analysis

Cartographic cognition is a unique process. It involves the use of the brain in recognising spatial patterns and relationships between the various real-world geographic phenomena. This cognitive process cannot easily be replicated by GIS software with its essentially linear analytical processes. However, cartographic visualisation can considerably enhance this process (Taylor, 1994, p.338).

Cartographic visualisation is a combination of all three aspects of cartography (Figure 2.1 A conceptual basis for cartography). It requires the latest computer technology, whereas at the same time it offers a visual method of understanding complex relationships while communicating to the viewer the reality of the topics considered in new ways on the screen (Taylor, 1991, p.14). WebGIS software provides the tools to interactively communicate the modelled real-world geography and to visualise its complex geospatial patterns and relationships.

Visualisation, a user-oriented approach

WHEREAS TAYLOR EMPHASISES the technology supporting visualisation and thus has a more production-oriented approach, DiBiase and MacEachren stress the use of maps adopting a more user-oriented approach (MacEachren, 1994, p.4). DiBiase defines a framework where maps play a role in scientific research. The major distinction in this framework is that between private visual thinking early in the research process and the public visual communication of the research results. This distinction provides a structure for discussing Webmaps in the fourth chapter.

However, the outlook on visualisation as presented by both Taylor and DiBiase leads to the conclusion, that visualisation equals cartography (MacEachren, 1994, p.6). Central to this outlook, is the idea that map use can be conceptualised as a three-dimensional space. This space is defined by three continua:

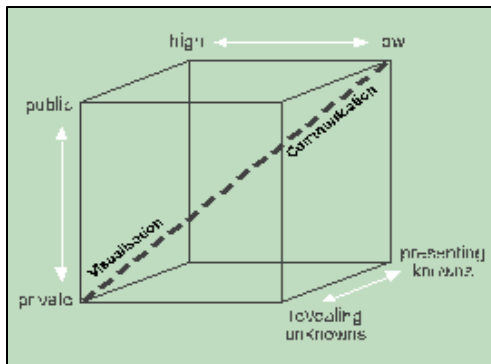


Figure 3. Cartography cubed

- from map use that is private (where an individual generates a map for its own needs) to public (where previously prepared maps are made available to a large audience);
- from map use that is directed toward revealing unknowns (where the user may begin with only the general goal of looking for something “interesting”) versus presenting knowns (where the user is attempting to access particular geospatial information); and

- from map use that has high human-map interaction (where the user can manipulate the map(s) in substantive ways) versus low interaction (where the user has limited ability to change the presentation).

If the Web would be able to cater for various options along the three continua, at least for those options along the diagonal axis from visualisation to communication described by DiBiase, this map use cube could be applied to Webmap use as well. The Web then would provide cartography with a new visualisation environment: ready for the new millennium. In the next two chapters grounds are investigated to substantiate this assumption.

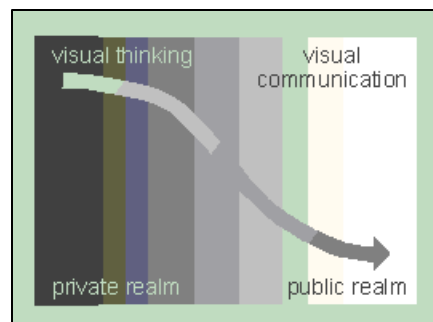


Figure 4. Visualisation in scientific research, DiBiase

Gaining insight into the use of maps may be input for the production of these maps. This feedback of user-information helps to enhance map production.

The Web as visualisation environment

Mapping constraints

THERE ARE VARIOUS factors influencing cartographic design that have to be taken into consideration when developing Webmapping applications. These “mapping constraints” include the map purpose, the geographic reality represented on the map, the available data to represent this geographic reality, the scale of display, the technical limitations to produce the map, the audience and the conditions of use (Muehrcke & Muehrcke, 1992, p.14). As the visualisation environment, in this case the Web, imposes many of these mapping constraints on cartographic design, this chapter looks into these mapping constraints in a structural manner, though they are all interdependent and not equally applicable.

One of the biggest advantages of the Web as a communication medium is the cross-platform compatibility. Different platforms, operating systems, are able to communicate with each other. This advantage, at the same time, is its weakest point: Web surfers access the information on the Web with different computer operating systems, different colour settings, different video memory, different monitors and different screen sizes. Because of this diversity, (carto-) graphic designers and Web (-mapping) application developers never have total control over the way the client views the message in the Web environment. When designing and developing for the Web, one has to cater for the lowest common denominator, or the most used combination, to overcome this diversity.

In the next paragraphs the topic of the Web as the medium of publication is looked further into. Several mapping constraints are discussed. First, the computer network architecture of the Web itself is looked further into, for it puts severe constraints on the amount of data and consequently on the Webmap and its functionality to be transferred to the user. In this sense it is not so much the availability of the geospatial data that poses a mapping constraint as is intended by Muehrcke & Muehrcke (Muehrcke & Muehrcke, 1992, p.14), but it is the quantity of the data that matters, the virtual map size.

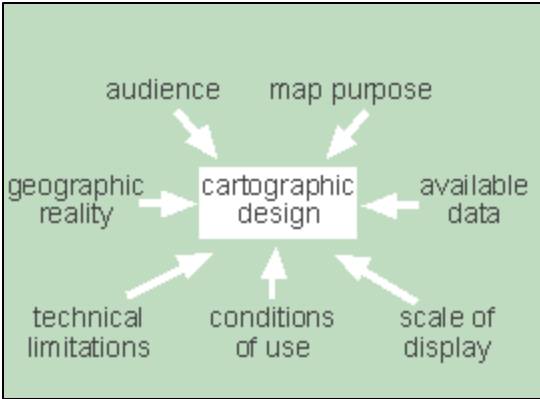


Figure 5. Mapping constraints

Second, as the geospatial data have arrived at the client-side, the Web browser as an interface will visualise the data. The existence of various marks and versions of browsers and the existence of various generations of mark-up and scripting languages is another factor of heterogeneity on the Web to be taken into consideration when designing Webmapping applications. Third, the browsers are depending for some functionality on their computer platform capabilities, one of which is colour, another being text. Fourth, there is the syntactical issue of file formats. Web browsers themselves are only able to display raster file

formats. This puts up another impediment for the vector-oriented GIS-community. These browser-dependent issues can be considered as the technical limitations of the visualisation environment posing mapping constraints. This is a user-oriented view on this mapping constraint, though Muehrcke & Muehrcke apply a production-oriented view.

The final general factor taken into consideration in this chapter is the physical map size, thus addressing the mapping constraint of the scale at which the geographic reality is displayed. This factor also introduces severe implications when designing and developing Webmapping applications.

This chapter does not discuss the mapping constraints of purpose, the geographic reality, and the audience. These issues are addressed in the following chapter. In that chapter these mapping constraints are mapped onto the visualisation cube of MacEachren and the visualisation-flow of DiBiase.

Computer networks, Internet and the World Wide Web

TO DISCUSS THE characteristics of the Web as a new visualisation environment for cartography in this chapter it is necessary to understand some of the basics behind it. The guiding principle in this discussion is the set of mapping constraints described in the previous paragraph. First, the attention is drawn to computer networks in general. Then the concept of the Internet is explained. Since the World Wide Web can be thought of as a subset of the broader concept of the Internet, this explanation forms the introduction to the concluding part of this paragraph that focuses on the Web itself

A computer network is an interconnected collection of autonomous computers (Lemmens, 1999, p.2). In discussing the Internet and the Web, client/server computing is the most relevant computing architecture for a computer network. In such a computing architecture a client computer application requests services from a supplying application, the server. An important feature of this model is that a client in some cases becomes a server and vice versa.

Several of these computer networks connected together form an internetwork. The Internet has its roots in the first start of a computer network project called ARPANet, a network created for the Defense Advanced Research Projects Agency (DARPA). The computer network was built as a distributed, decentralised system with each node of equal importance. By the end of 1969, ARPANet connected four host computers at different universities across the USA.

Internet

Over the years, the ARPANet grew, encompassing more and more computers in the USA. In 1983 a new communication protocol called TCP/IP (Transmission Control Protocol/Internet Protocol) was adopted. The adoption of this protocol by USENET and EUnet and the redirection of other communication protocols, as TelNet, FTP, and SMTP to run on top of TCP/IP are considered to mark the birth of the Internet as known today.

The TCP/IP protocol is very important today. It allows standalone networks to be connected to the Internet or to be linked together to create Intranets by means of IP routers. A router is a computer that transfers information in discrete units of data called IP packets or IP datagrams from one network to another. TCP/IP software makes each computer attached to the network a sibling to all the others; it hides the routers and underlying network architectures and makes everything seem like one big network. Connections to the Internet are identified by 32-bit IP addresses (for example, 131.211.122.139). Given a remote computer's IP address, a computer on the Internet can send data to the remote computer as if the two were part of the same physical network. As the "dotted quad" numeric notation is difficult to remember, Domain

Name Services (DNS) convert the IP address into an alphanumeric string, the host name, to identify the Internet host more easily.

World Wide Web

Besides Internet there is the World Wide Web (www, w3 or simply “the Web”). It started at CERN (the European Laboratory for Particle Physics) in 1989. To enable the dissemination of information among the various research facilities at CERN by building a “distributed hypermedia system”, the existing communication protocols of the Internet were not sufficient. A new communication protocol to be run on top of TCP/IP was developed: HTTP (HyperText Transfer Protocol). This protocol was specifically geared towards finding and retrieving documents from various servers.

Standardised ways to store the information were also needed. For this purpose, the HyperText Markup Language (HTML) was devised. Later in this paragraph HTML will be treated in further detail. Uniform Resource Locators (URLs) specify how to access network resources such as HTML documents. The URL states both the transfer protocol and the server computer’s IP address or domain name, the directory on the server where the resource is located and the name of the resource itself. An additional search string of parameters and their associated values can be part of the URL to request dynamically generated resource contents. Sometimes this requires the statement of a server-side executable in the URL as well. The most important use of URLs is to identify targets of hyperlinks in HTML documents.

A communication protocol (HTTP), a standardised data format for the storage of hypertext documents (HTML) and a way to locate these resources (URL and IP) make up the Web as a means of communication via the Internet. In practice, the Web is now a vast collection of hyperlinked resources, spanning the globe, merging techniques of networked information and hypertext to make a powerful global information system. It makes resources accessible over the network as part of a seamless hypertext information space (Peterson, 1996).

WWW = World-Wide Wait?

IN A CLIENT/SERVER architecture as the Web, computer processing is distributed between the server-side and client-side. The specific client/server balance determines the amount of data transmission taking place over the Web. When targeting a wide audience, it should be borne in mind that still most people are connected to the Web through a modem. Even in a corporate Intranet environment, personnel should not waiste its time waiting. To bypass long downloading times, the data traffic the Web should be kept to the minimum, otherwise the Web turns into the World-Wide Wait.

Dealing with Webmapping applications, not only an HTML document has to be transmitted from the server across the Web to the client. The HTML document however also includes a Webmap that results from requests for new map views, searches and GIS-analyses passing from the client to the Web server (Plewe, 1997, p.63). This points to some specific issues to deal with for Webmapping applications to reduce data transfer: the client/server balance of the geo-processing and the document size. In the next paragraphs the server and client application are discussed for WebGIS. Then the attention is drawn to the aspects influencing document size.

Server-side: interfacing with WebGIS software

As described in chapter one, the main set-up of a Webmapping application service is similar to the client/server architecture of the Web. However, there is need for additional technology to provide a Webmapping application, either at just the server-side or at both the server-side and the client-side. This paragraph discusses specifically the extension of the server-side to provide Webmapping applications.

The server-side of a Webmapping application consists of a Web server and a WebGIS software programme. The Web server consists of a powerful computer and software that contain information to be distributed over the Web on request from one or many clients by HTTP. As the Web server software is not able to do geo-processing, it has to be able to communicate with the WebGIS software to pass on requests from the client for geo-processing. There are several interface standards, the Common Gateway Interface (CGI), and server Application Programming Interfaces (APIs) to enable the communication between Web server and WebGIS. The simplest interface is CGI and it is supported by most Web servers. This simplicity is reflected in its limited GIS functionality. As CGI does not extend HTML, the limitations inherent in HTML prohibit the direct manipulation of Webmaps (Peng, 1997, URL). Server APIs however were developed for specific servers: e.g. ISAPI for Microsoft servers and NSAPI for Netscape servers. The WebGIS software then takes care of the geo-processing. The software returns the result that is reformatted for interpretation by the client's Web browser itself or with additional functionality from a plug-in or Java applet. The Web server then returns the result to the client for display (Plewe, 1997, p.5).

Thin clients: Web browsers

The computing capabilities at the client-side determine to a large degree the amount of data transmission taking place over the Web. These computing capabilities at the client-side are usually referred to as the "thickness" of the client. The thicker the client, the less data is needed to be transmitted for server-side geo-processing and the faster the new map view is being created.

The thinnest client consists of only a Web browser application. In the early 1990s, the idea of using hypertext via the Internet was enhanced by a team at the National Center for Supercomputing Application (NCSA) resulting in NCSA Mosaic, a graphical point-and-click hypertext browser that made Internet easy resulting in a growing public appreciation of the Web. There exist nowadays a lot of different Web browsers, among which Netscape Communicator (NC) and Microsoft Internet Explorer (MSIE) best known. Of each browser several versions are in use simultaneously, not all of them supporting the latest developments in Web technology (e.g. new HTML tags, dHTML, CSS, JavaScript). This issue is addressed in the next paragraph, for Webmapping applications not only consist of Webmaps, but most likely include hypertext as well.

A side step: hypertext documents and graphics file formats

Web browsers are graphical user interfaces (GUIs) for the display of Web pages. Most Web pages consist of hypertext documents in which graphics are embedded. The hypertext documents are stored using HyperText Markup Language (HTML), a universal, cross-platform standardised language for the display of texts and other visual elements in the Web browser. Already soon, HTML posed limitations to the creativity of Web designers and developers. New, browser-specific HTML tags were developed as well as new generations of HTML and new profiles of the SGML standard (e.g. dHTML, VRML, and XML). As computer capabilities in general advanced, client-side computing became an option embedding JavaScript into hypertext documents. Also this scripting language has several generations. Cascading Style Sheets (CSS) have added more control on the layout of hypertext documents. New browsers and browser-versions were developed supporting this new technology (Weinman, 1998, p.29). This has led to the situation, that the contents of Web pages are interpreted differently by different browsers (Schlimm, 1998, p.2).

Developing Webmapping applications, differences in interpretation and in the capabilities of Web browsers has to be kept in mind as these applications consist not only of Webmaps, but also of hypertext. Usually, based on many of the same parameters used for requesting the map itself, this hypertext is generated dynamically by CGI/server API script or an inline macro

language as Microsoft's Active Server Page (ASP) technology or Alaire's ColdFusion technology (Plewe, 1997, p.227-8). Another way to bypass this heterogeneity is to redirect clients to different Web sites based on browser or object detection, each site designed and developed to the specific characteristics of a certain browser's capabilities.

Though the hypertext part needs attention while developing a Webmapping application, the important contents of the Web page displayed in the Web browser is the Webmap itself, conveying the geospatial data in a graphical form. Graphics can be simply incorporated within the Web page. Browsers themselves natively support raster graphics file formats only: the Joint Photographic Expert Group file format (JPEG), specially developed for transmission of photographs using compression techniques and the Graphics Interchange Format (GIF), a file format specifically developed for online transmission. Furthermore, the GIF format provides specific capabilities as animation, transparency, and interlacing (Weinman, 1998, p.46).

With regard to the graphical characteristics of Webmaps, the GIF file format is most the suitable of the two. First, the compression-technique of the GIF file format works very well for graphics with large homogeneous areas, as are polygon maps, whereas the compression-technique of the JPEG file format introduces artefacts in areas of solid colour because of dithering (Weinman, 1998, p.57). This loss of sharpness is also apparent through the fuzziness introduced in the line-work (Peterson, 1998, URL; Schlimm, 1998, p.3). Second, Webmaps being graphics with few colours (less than 256) do not suffer from GIF compression, because this format does not cause a loss of quality (Weinman, 1998, p.48). Third, it usually takes longer to display JPEG files than GIF files in a Web browser due to the need for decompression.

Nevertheless, as far as Webmapping applications are concerned the raster graphics file formats have some disadvantages. For example, interaction between the graphic displayed in the Web browser and the user is limited. Assigning URLs to specific hotspots in the graphic results in a clickable graphic, an image map. Though the name suggests differently, the graphic not necessarily has to represent geospatial data. This is the only means of interaction possible. Vector graphics file formats tend to represent vector GIS data more efficiently than raster formats, resulting in smaller file size and lesser data to be transferred over the Web. Secondly, complex linear features can be represented more precisely by vector images, bypassing anti-aliasing (Strand, 1996d, p.32). Thirdly, the vector format allows for object-oriented manipulation of the map, such as hyperlinking and selection. So, there are several technical limitations adding to the mapping constraints. In the next paragraph several technologies are discussed that extend the capabilities of Web browsers allowing for more interaction and for the display of vector graphics file formats.

Thick clients: plug-ins, ActiveX and Java

To support displaying the results of geo-processing at the client-side several technologies have been developed such as plug-ins, ActiveX controls and Java-applets. GIS plug-ins are small software applications installed at the client-side to extend browser capabilities to handle GIS data formats that current HTML cannot recognise. They enable the user to interact with Webmapping applications from within the Web browser without installing traditional GIS tools. Some simple GIS functionality can be built into the plug-in and performed locally. The execution requires no server-side geo-processing, thus reducing data transfer over the Web.

Another approach to extend browser capabilities for Webmapping applications is the use of ActiveX controls to create GIS controls. However, as Microsoft developed ActiveX control technology, it only works for the MSIE browser on a Windows platform. ActiveX controls are embedded within an HTML document. On opening the document, the ActiveX control is

downloaded onto the client like a plug-in, but needs installing, as it is no executable programme.

The final approach presented to extend client-side computing capabilities is the inclusion applets into the HTML document. Applets are small Java programmes (Java is a platform independent programming language). Java interpreters (Java Virtual Machines) that are included into the browser execute these programmes. Usually the Java applet code is pre-compiled into a Java Class and stored on the server. The HTML document then only holds a reference to the file containing the Java class, which is loaded and run by the Java Virtual Machine (JVM) on the client-side.

The solutions extending browser capabilities have disadvantages with regard to user-friendliness. In the case of Java or ActiveX technology, the size (in bytes) of the applet or control contributes to the total amount of data to be transmitted across the Web. This especially holds true for Java applets: they must be downloaded every time when accessing the HTML document containing the applet. In the case of proprietary plug-ins, users have to knowingly install the extension to their browser software, or allow the software to be extended automatically (Barr, 1998a, p.15). This may pose an obstacle for some users (Koop, 1998a, p.29 Plewe, 1997, p.69). For them this procedure seems too technical. In corporate network environments installing new programmes may not even be allowed by network administrators (Barr, 1998a, p.15). On the other hand however, the network administrators themselves may install this plug-in technology, because these added browser capabilities are supported corporation wide.

The trade-off between different client/server configurations depends on the competence of the users, the power of their computers and the capacity of the network connection. A thin client is best for applications aimed at a large audience with little demand for advanced GIS analysis capabilities, for these would require a lot of geo-processing at the server-side (Marks, 1998, p.27). The thick client-option is suitable in situations in which users frequently access geospatial databases and require a high degree of rapid interaction (Plewe, 1997, p.70).

Browser display of Web documents

APART FROM THE mapping constraints the Web poses on the quantity of data for display, it also imposes some technical limitations. These stem from the hardware, the computer platform, on which the Web browser application runs. There are three technical limitations in the display of Web documents. First the display of hypertext is platform dependent. Second the display of colour depends on the platform as well. Third, there is the size of the map, determined by screen size and screen resolution. The latter technical limitation introduces even yet another mapping constraint: the scale of display.

Text

With regard to text there are several aspects that are important for both hypertext and graphics. First, there is the issue of typeface. As far as hypertext is concerned, the typeface can be defined in several levels of detail. The decision which typeface to display can be assigned to the browser completely by leaving out any specification of typeface. This usually results in all text being displayed in serif type. However, serif types are entirely unsuitable for this purpose (Ditz, 1997, p.575). Only sans-serif type should be considered, but even in this case, some typefaces are more suitable than others. Tests demonstrated that Arial and Univers are best for on-screen display of text. A next level of detail can be introduced by defining this type style (e.g. serif, sans-serif or mono-space). Based on the browser preference file, a specific typeface of the respective type style is displayed. The highest level of detail is the specification of a particular typeface. When specifying a certain typeface it has to be borne in mind, that its display is dependent on the client-side platform. Only those typefaces can be displayed whose definition files reside on the client-side platform.

Achieving a holistic design, integrating the same design principles for both hypertext and graphics making up the Web document, the typeface used in the map should not differ from the hypertext. To overcome this client-side dependency two strategies can be followed. On the one hand, only those typefaces should be specified in the document common to most platforms. On the other hand, state-of-the-art technology can be incorporated to force server-side type face to be displayed by browsers: (OpenType, TrueDoc). However, this technology is only available on the latest browser versions (Will-Harris, 1997, URL).

The second issue regarding text is the limited choice in selecting type size for the display of hypertext. The resolution of computer screens is adequate for displaying 10pt text, but cannot represent at actual size very small text and symbols, typical of analogue maps (Brown, 1993, p.130). Roche, however, states that eight points is the minimum type size if text is to be legible on screen (Roche, 1998, p.4). Larger type sizes are well readable, but require a large area that is barely available. Another point to be considered in the cross-platform environment of the Web is the different way that hypertext is represented on screen: typeface on the PC is represented larger than on a Macintosh. (Weinman, 1998, p.257).

Most limitations regarding text can be overcome nowadays using typeface specially designed for on-screen display as Verdana and Georgia (Schlimm, 1998, p.3; Will-Harris, 1996, URL). The latest technology regarding on-screen text type, ClearType, even takes into account the way text is displayed: the phosphor dots (Will-Harris, 1999, URL). Treating the different phosphor dots individually, typography with sub-pixel resolution design elements is facilitated!

Web-safe colour palette

For displaying colour in Web documents, most Web browsers use the same technique: they use colour look-up tables (CLUTs) drawing from the internal colour palettes of the different platforms. However, programmers rather than graphic designers picked these colours to meet system requirements rather than design requirements (Roche, 1998, p.5; Weinman, 1998, p.141). Although colour palettes of PCs and Macs are different, they have 216 colours in common. To allow for cross-platform differences and to still be able to control the correct colour perception at the client-side, only colours from this “Web-safe” colour palette should be chosen. Otherwise, colours may dither on the user’s screen, reducing the graphics quality (Weinman, 1998, p.136).

With regard to displaying colour, another drawback of storing graphics of Webmaps as JPEG files comes on top of the disadvantages already discussed: some colour information is lost as well. Since Web-safe colours are not always translated into other Web-safe colours during compression and de-compression, control over colours of a JPEG -file is impossible (Weinman, 1998, p.151). The difference in colour is however sometimes important in Webmaps to discriminate between geographic phenomena.

Webmapping size

A mapping constraint in which both technical limitations play a role and the scale of display is the interplay between the size of the computer screen and addressibility, often termed screen resolution. This interplay limits the size of the Webmap (Elzakker & Koussoulakou, 1997, p.9). Displaying a large array of pixels on a small screen leads to illegible fine details and small text. For example, a 14-inch monitor gives good resolution at 640 x 800 or 800 x 600, but not higher (Brown, 1993, p.130). As both the screen size and addressibility differ at the client-side, the actual display area varies by a factor of 9 (Stynes et al., 1996, URL). Therefore, most Web designers take an addressibility of 800 x 600 as a standard for developing Web sites (webreview.com, 1999, URL; lynda.com, 1999, URL).

When developing a Webmapping application, at least the Webmap itself needs to fit on the screen, inside the Web browser. On a standard 14-inch monitor this leaves room for a document of 10.5 by 7.5 inches. Subtracting the space necessary for other Web document elements like navigation tools leaves a Webmap size of just 9 by 5 inch (Plewe, 1997, p.181). To deal with this mapping constraint though still facilitate interaction with an extensive geospatial database, clients often have to zoom out (which will render small details illegible), or scroll and pan around the map, which means that they do not see the whole map at a glance (Brown, 1993, p.130). However, this makes the map less attractive and much less interactive: it will take users more time to see parts of the map (Plewe, 1997, p.181).

There are several options to get over the problem of illegibility introduced by re-scaling. One option is setting scale-dependent layer visibility (Harrower, 1996, URL). In most GISs different categories of data are stored into different layers. As scale is reduced, the less important layers can be switched off or new layers intentionally designed for small-scale visualisation may be shown. Another option is setting scale-dependent cartographic visualisation. In this case, the same geospatial data set is displayed at all scales, but depending on the scale, the cartographic visualisation changes to optimise legibility.

The Web: mapping constraints

IN THIS CHAPTER various factors influencing cartographic design for Webmapping applications, the mapping constraints, that are imposed by the Web as visualisation environment have been discussed: the available data to represent geographic reality, the technical limitations, and the scale at which the geographical reality is displayed.

The client/server architecture of the Web itself puts severe constraints on the amount of data and consequently on the Webmap and its functionality to be transferred to the user. To set the context for the discussion of the mapping constraints, computer networks, Internet, and the Web were first looked further into. Then followed a description of the issues involved when avoiding the World Wide Wait: a thorough overview of server-side processes as interfacing with WebGIS software and of client-side computing, Web browsers, to evaluate the suitable client/server balance for geo-processing involved in Webmapping applications. A side step was introduced discussing the data to be actually transferred, hypertext documents and graphics, to draw attention to the Web browser heterogeneity in capabilities. To finish the discussion of the mapping constraints resulting from the availability of data, the thick client option was looked into as GIS plug-ins, ActiveX and Java extend browser capabilities. This also addressed the next mapping constraint, the technical limitations, as Web browsers themselves are only able to display raster file formats whereas, the GIS-community is mainly vector-oriented.

The technical limitations were further looked into by drawing attention to the computer platform-dependency of browsers for their capabilities, explaining issues of typeface and the Web-safe colour palette. The final general factor taken into consideration in this chapter was Webmapping size, the mapping constraint of the scale at which the geographic reality is displayed.

The discussion in this chapter of some mapping constraints is a partial introduction to the establishment of requirements for cartographic design for Webmapping applications. These requirements may serve to compose a description of the functionality of an ideal WebGIS software package. This description forms the basis for the design of the problem that is used to benchmark several commercially available WebGIS software packages. Lacking in this chapter is a discussion of the mapping constraints posed by mapping purpose, the geographic reality, and the audience. These issues are addressed in the following chapter. In that chapter these mapping constraints are mapped onto the visualisation cube of MacEachren and the

visualisation-flow of DiBiase, to establish specific requirements for cartographic visualisation for the different Webmap use goals: visualisation strategies. These visualisation strategies provide additional requirements to enhance the description that forms the basis for designing the problem for benchmarking.