WWW Delivery of Spatio-Temporal Modelling Results for Catchment Management: A Case Study

Kerry Taylor, Ross Ackland, Gavin Walker, and Dean Jackson
CSIRO Mathematical and Information Sciences
GPO Box 664 Canberra ACT 2601.
{Kerry.Taylor Ross.Ackland Gavin.Walker Dean.Jackson}@cmis.csiro.au

Abstract Mathematical models of physical processes, when implemented on computers, can be useful decision-making tools for environmental management. However, because the models have limited scope and embedded assumptions, they must be applied and interpreted with care. Often effective use requires both understanding of the underlying models and familiarity with the operating interface of the computer implementation. Some problems require access to several models with data flowing between them. Then the difficulty of effective use is compounded, especially when the models are implemented on different computer architectures. These difficulties make the models inaccessible to many stakeholders in the decision-making process. On the other hand, the World Wide Web (WWW) is an enabling technology for the wide dissemination of digital information. Recent developments mean it can now support powerful information navigation and visualization, but the technology brings with it new requirements for software design and usability. This paper reports on a case study for a WWW-based spatial information system for water catchment management. The system is aimed at non-specialists and so does not support invocation of computational models, but instead incorporates interactive tools for spatial and temporal data visualization. The data derives from several models as well as field measurements, and is supported by arbitrary background information.

1. INTRODUCTION

The popularity of the World Wide Web (Berners-Lee et al., 1994) as a vehicle for information distribution is due primarily to the ease of access it allows. Client computer systems of any system architecture need to be equipped only with low cost and readily available browser software for basic functionality. Additional multi-media functionality can be obtained for some architectures using locally-installed "plug-in" tools for data manipulation. The recent introduction of the Java suite, especially the "applet" concept, enables architecture-independent support for flexible local computation. Java is especially useful for data visualization and navigation.

On the other hand, the difficulty of access is a major inhibiting factor in the take-up of decision support system technologies. In addition to the usual difficulties for novice access to computer software, modelling systems are often designed under assumptions about purpose and scope that are only well-understood by domain specialists. Interaction with modelling systems without domain expertise can lead to ill-informed interpretation of results. The problem is compounded when the inter-operation of multiple models is required for adequate representation of a problem. For this reason, wide access to models is less desirable than convenient dissemination of modelling results.

The Web can offer improved access to modelling systems but the design of a system for Web delivery raises some interesting technical and usability issues.

This paper reports the HYDRA4 system that uses Web technology to deliver model-based prediction data to non-specialists. The underlying models are operated by experts prior to delivery. Modelled scenarios are packaged up with empirical data, a map-based navigation interface, and interpretive material to support non-specialist understanding and decision-making. Inbuilt visualisation tools for spatial and temporal data handling are delivered to users along with the data.

2. HAWKESBURY-NEPEAN CATCHMENT MODELLING

The Sydney Water Corporation is responsible for water supply and sewage treatment for Sydney, Australia. Sydney Water is charged with responsibility for maintaining river water quality standards in the Hawkesbury-Nepean river system within their jurisdiction. Water quality is determined by concentrations of nutrients, biological oxygen demand, suspended sediment and so on in the riverine and estuarine water.

Hydrological models are used to evaluate the effect of development options to support decision-making. Three very different models support this work: HSPF (Barnwell and Kittle, 1984), a proprietary package known as SALMON-Q, and CMSS (Davis et al., 1991). HSPF is a large Fortran program, customised to their needs by Sydney Water over many years. It models land run-off and water flow and composition in streams. Input to HSPF is via a complex fixed-column format command and data file. Time-series output is available in a special binary file format and also in a text format intended for processing by a third-party graphics package. SALMON-Q is a proprietary modelling package for predicting flow and composition of in-river water. Input is made in a special-purpose text file format.

Although SALMON-Q and HSPF are implemented on mid-sized Unix-based computer systems, CMSS is a proprietary PC-based model. Mathematically, it is by far the simplest of the three models, and it demands much less detailed input data. CMSS does not model water flows in streams: the drainage from each sub-catchment region is assumed to flow into an adjacent sub-catchment region. Input is made through an interactive menu system and output is presented as textual reports.

Each of the models has its strengths. HSPF is used with a high degree of confidence in the results, but it is expensive to calibrate in a region and cannot handle tidal flows. SALMON-Q is appropriate for river estuaries subject to tides, but does not model surface run-off in the catchment. CMSS is a weaker model, used where fine detail empirical data is not available and accuracy of results is less important. Sydney Water have partitioned the Hawkesbury-Nepean catchment basin into regions appropriate for each particular model. Each model is calibrated for each region it covers with topological and static parameter data appropriate for the region. In some places the hydrological network topology requires outflow results computed in some regions to become inflow data into other regions covered by a different model.

Let us consider a typical investigation for which the models are used. Suppose there is a project comprising a proposal for a major urban land development. The changing land use from rural to sewered urban over the development region will affect the volume and composition of the storm-water run-off into the river system. The increased sewage load will necessitate the upgrade of existing treatment plants or the construction of additional treatment plants that discharge into the river system. The plants must be designed and sited to ensure nutrient concentrations downstream do not exceed acceptable standards. Artificial wetlands can be constructed to remove nutrients at selected locations.

The modellers consider various scenarios comprising alternative management options. The models enable them to predict the effect of these options on the river water flow and quality over an extended period of time and under varying rainfall conditions.

To use the models successfully, the modellers draw on a wealth of experiential and scientific knowledge about the underlying process models represented in their computer implementations. This includes the method of representation of a conceptual scenario in the process model, the applicability of the models to particular situations, and the reliability and accuracy of model results. Furthermore, the modellers draw on considerable experiential knowledge in setting up and operating the computer implementations of the process models.

For these reasons, the models are inaccessible to decision makers and stakeholders who are not specialist modellers. In the case of Sydney Water, these non-specialist stakeholders include engineers and policy makers distributed across the organization. In the first instance, the hydrological modelling group wish to make the outcomes of their modelling work more readily available to those people. Providing political, legal and administrative barriers can be overcome, they are also interested in making the information available to the wider public including community groups, land developers and local councils interested in the Hawkesbury-Nepean catchment.

The solution for Sydney Water is embodied in the HYDRA4 application. The modellers use their choice of models to analyse alternative scenarios for a project. The relevant model inputs, model results and supporting information are loaded into a persistent data store. The information in that data store is made available to the user community over the Web.

3. SYSTEM DESIGN

In this section we describe and justify the system architecture of HYDRA4 in some detail, so that the experience might aid the development of other Web-based spatial decision support systems.

The architecture comprises a slightly "fatter" client model than many Web-based systems, because temporal and spatial data visualization and presentation are performed at the client. The tools for data visualization are delivered from the server, along with the data itself, as Java applets. During development, we found that it was important to consider issues of data transmission size, frequency and format as well as the distribution of functions across the server and the client in order to achieve reasonable response times at the client. There is a balance to achieve between delivering a large amount of data in each request from the client (although some of it might not be used), and delivering smaller amounts of data each time but requiring repeated network access by the client to obtain more.

We also aimed to make efficient use of software licences for the spatial and relational database systems. Under this architecture there are no client-side licences at all. Client requests to the databases are serialised in the servers, so only one concurrent user licence is necessary for each of the two servers. This design is adequate when only small numbers of concurrent users are expected.

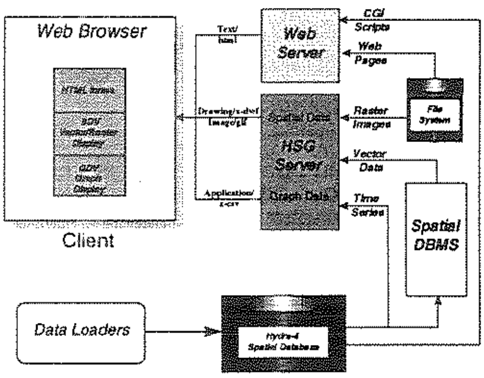

Figure 1: HYDRA4 Architecture

Figure 1 illustrates the main components of the HYDRA4 system. There is an administration subsystem for loading the persistent data store and a run-time subsystem comprising client and server side components.

3.1 Administration Subsystem

The administration system comprises a suite of programs called *data loaders* to assist modellers to define the project and scenario structure and to populate the run-time system. The data loaders extract selected inputs and outputs of model runs from the CMSS, SALMON-Q and HSPF data files. The data is inserted into a spatial database implemented in SDM (Abel, 1989) and Oracle. At this time, the loaders effectively integrate the models, so that the database offers a coherent schema. They rely entirely on the expertise of the modellers who have used the models beforehand to determine the applicability of models to scenarios, to manage the interaction between models, and to appropriately represent scenarios within each model.

The loaders also load empirical data collected from gauging stations including precipitation and stream constituent concentrations. They permit *ad-hoc* viewing and updating of the database. They enable the association of spatial locations with raster data (such as photographs and land use maps), that is stored in the native file system outside the spatial database. The loaders also support the association of other multi-media data with objects via URL references. Usually, the URL references link additional HTML pages maintained by the Web server, but they can refer to any World Wide Web resource.

3.2 Client Subsystem

To access the run-time system, the client needs only a Web browser that supports Javascript and Java—such as recent versions of Netscape Navigator or Microsoft Explorer. The client begins interaction with HYDRA4 by connecting to the HYDRA4 home page at a fixed URL. From this time, the client interacts with HYDRA4 by following hyper-text links in Web pages, filling and submitting HTML forms, and locally interacting with Java applets.

The two Java applets are delivered transparently to the client system on demand: the SDV for spatial data viewing and the GDV for graph data viewing. They are responsible for client-side interaction with the spatial and graph data respectively. They are general-purpose viewing tools without special functionality for HYDRA4. Snapshots of each are given in figures 2 and 3.

The SDV interprets vector and raster data delivered from the server for client-side interaction with a map. Clickable buttons are offered for a range of panning and zooming functions. A layer-manager panel allows user selection of vector layers for presentation and raster backdrops. Requested layers that are not already available locally due to a previous request, are requested back from the HSG server, transparently for the user. In HYDRA4

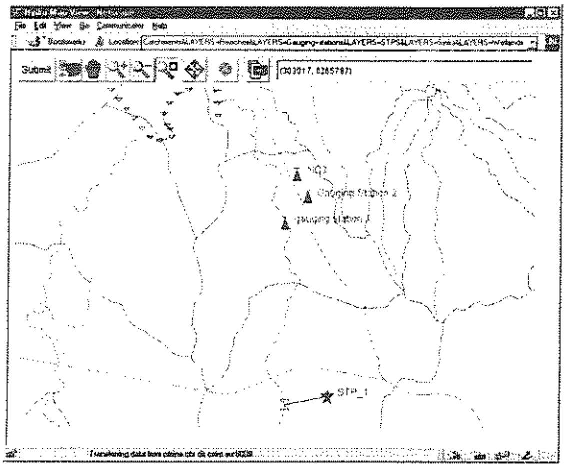

Figure 2: HYDRA4 Spatial Data Viewer

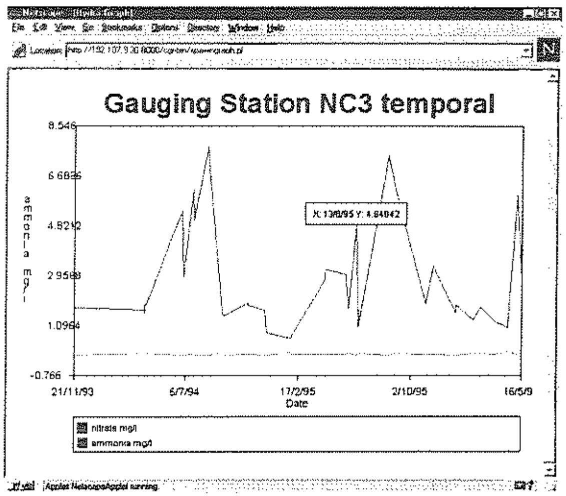

Figure 3: Graph Data Viewer

these layers can include, for example, catchment boundaries, reaches, sewage treatment plants, wetlands and roads. Individual objects displayed iconically on the map, including reaches and sewage treatment plants may be clicked by the user to obtain more information about them. Such a request causes the SDV to pass a reference for the object back to the Web server for the return of aspatial data about the object.

During development of the SDV, effort was focussed on achieving reasonable response times for interactive map navigation. In its current form, the SDV achieves good results for full-screen maps incorporating vector data comprising up to 30 000 vertices.

The GDV applet is delivered to the client when graph data, such as constituent concentrations in a reach or river-bed cross-sections is requested. The GDV offers considerable flexibility for the user to change the range, scale, axis labelling, colour scheme and overall appearance of the graph. Thus the user can choose the presentation format for the data, without further requests to the server.

3.3 Server Subsystem

Transparently for the client, the server-side of HYDRA4 actually comprises two independent HTTP servers. One is a standard commercial Web server. This server plays a role in HYDRA4 but also can be used for delivery of documents and processing of forms for purposes unrelated to HYDRA4, as in the usual Web server configuration. In HYDRA4, this server acts as the initial point of contact for the user, delivering the HYDRA4 home page. During interaction with the client, it delivers all the HTML documents and forms in the HYDRA4 system. It also invokes Common Gateway Interface (CGI) programs that extract the attribute data (non-temporal and non-spatial) from the database system.

The other server involved in HYDRA4, the HSG server, is purpose-built for HYDRA4. The HSG server interprets requests for spatial and graph data made by the client via forms or the SDV, according to the HTTP protocol. The requests are satisfied by interrogating the spatial database or the file system store. Contrasting with the standard approach of spawning external processes to handle such requests via the CGI, the HSG server interrogates the database and file system directly. It also post-processes the data and encodes it as an appropriate MIME type.

Spatial data is encoded by the HSG in GIF format for raster images and DWF format (Autodesk, 1997) for vector data. These are the formats interpreted by the SDV. Image selection by spatial location and scaling of raster images is done in the HSG server. This is preferred to the SDV because Java is unable to handle the large volume of raster data efficiently enough for interactive map navigation.

Temporal data, relating to objects such as sewage treatment plants, wetlands and reaches is extracted from the database by the HSG and filtered in one of several ways according to client parameters. Parameters select from the available constituents, modulate the interval and time-step of data presented and determine the form of presentation from a range including temporal, exceedance, and maximum or average along a river length. The HSG reformats the processed temporal data in to an ASCII comma-separated values format appropriate for interpretation by the GDV.

In fact, the HSG server's primary role is to access, process and reformat spatial and temporal data. The extra development cost required to implement the HTTP protocol in the HSG server is small due to the simplicity of the protocol itself, based on a few primitive commands modulated by sets of parameter-value pairs. The cost is well offset by the run-time advantages. The design improves speed of access to the data-sources because database connections are initiated at startup and remain open for the life of the server. Furthermore, the server-side post processing of data retrieved from the databases is implemented in the same process.

4. DISCUSSION

Although the use of computational models can lead to better informed decisions on environmental issues, they must be used with care. HYDRA4 is designed for the very common situation that domain experts, who have access to and an understanding of computational models, are separated from decision makers who must act on the basis of model predictions and other diverse information. There are two arguments in favour of the retention of models within the control of domain experts.

First, the models inherit the usual problems of special purpose software packages. They run only on computers of specific architectures and performance profiles. They require a heavy investment in learning to operate them and, sometimes, in calibrating them for use in particular geographic regions. Investment in software licences, special computing equipment and operator education can sometimes only be justified by the scope for re-use of the systems for a range of projects.

Secondly, the decision to apply a particular model to a particular problem has to be made in a context of an understanding about input data availability, the range of independent variables, model assumptions, model scope, and model reliability. Assumptions about the physical process modelled are rarely explicitly available and usually only implicitly coded in the input data requirements and internal data manipulations. This is particularly important where distinct models, with differing scopes of applicability, are to be integrated to effectively model an extended scenario.

These two factors reinforce the need for mechanisms for the delivery of information relating to model outcomes to decision makers and stakeholders. The HYDRA4 approach takes advantage of the simplicity and availability of access offered by the Web to achieve this. There is a drawback: through their ability to select models, to choose the range of scenarios for investigation, and to select of supporting information, the domain experts responsible for the HYDRA4 system have considerable scope for prejudicing decisions made on the basis of that information.

HYDRA4 is not the only example of environmental modelling on the Web. The Environment Resources Information Network supports public Web access to two independent models that predict species distribution across Australia (Department of the Environment, Sport and Territories, 1997). Unlike HYDRA4, the models are run on-line in response to user requests. Although the models could be complex, the Web version enables only very simple parameterised input: nearly all parameters and data are initialised already. This falls short of the full scenario description and model integration that is of interest in the HYDRA4 case study. Although the model results are presented on a map, no interaction with the raster image is supported.

The Integrated Catchment Management Information and Support System (ICMISS), (NSW Department of Land and Water Conservation, 1997), shares common goals with HYDRA4. Although it is more ambitious than HYDRA4 with respect to the distribution of data sources, the catalogue search access to data, and the diversity of data presented on the navigation map, it is only at prototype stage at present. The tighter integration of the HYDRA4 system enables more convenient map navigation, a user interface structured around projects and case studies, and convenient presentation of graph-based data, especially temporal data. HYDRA4's more sophisticated map interface is much more responsive to user interaction because, whenever possible, it locally interprets selection of map layers, panning and zooming—requesting only locally-unavailable data from the server.

5. CONCLUSIONS

For Sydney Water, HYDRA4 offers a vehicle for dissemination of a broad range of information associated with catchment management projects. The audience can be broad because of the widespread access capabilities and the minimal requirements of access equipment. The information can be timely because there is no delay in distribution. The information can be heterogeneous because of the support for Web hyper-links, leveraging on the abilities of browsers to interpret a wide range of data types. By supporting the definition and packaging of model predictions as case studies, disparate models are delivered 'safely' into the hands of non-specialists. With the inclusion of Java applets for spatial data navigation and temporal data visualization, the HYDRA4 system achieves flexible user interaction without sacrificing efficiency. The design of HYDRA4 demonstrates a general solution to difficulties with the delivery of integrated spatial decision support systems to non-specialists.

The question naturally arises about whether the approach can be extended to support data transmission in the reverse direction. Is it feasible to use the Web for transmission of spatial and other complex data from client systems to server systems? The server systems could offer computational processing services on behalf of the client, or persistent data storage and redistribution. A whole new class of Web-based activities and distributed spatial decision support systems could be enabled.

Acknowledgements The HYDRA4 project has been a collaborative effort of the CSIRO and the Sydney Water Corporation. Many people have contributed, including Sue Cuddy, Trevor Farley, Dean Kuo, Gerrit Riessen, and Dave Abel of CSIRO and S. Maheswaran and Amir Deen of Sydney Water.

6. REFERENCES

Abel, D. J. SIRO-DBMS: a database tool-kit for geographical information systems. *International Journal of Geographical Information Systems*, 3, 291–301, 1989.

Autodesk, 1997. Drawing web format. URL: http://www.autodesk.com.

Barnwell, T. D. and J. L. Kittle. Hydrologic simulation program—Fortran: Development, maintenance and applications. In *Proceedings, 3rd International Conference on Urban Storm Drainage*, pp. 493–502, Goteburg, Sweden, 1984.

Berners-Lee, T., R. Cailliau, A. Luotonen, H. F. Nielsen, and A. Secret. The world-wide web. *Communications of the ACM*, 37(8), 76–82, 1994.

Davis, J. R., P. M. Nanninga, J. Biggins, and P. Laut. Prototype decision support system for analysing the impact of catchment policies. *Journal of Water Resources Planning and Management*, 117(4), 399–414, 1991.

Department of the Environment, Sport and Territories, 1997. ERIN biological database. URL: http://www.environment.gov.au/database/biodb.html.

NSW Department of Land and Water Conservation, 1997. Integrated catchment management information and support system. URL: http://www.icmiss.dlwc.gov.au.