Dynamic Retrieval in an Urban Contextual Databank System using Java-CGI Communications

Development of the SUCoD Prototype

Chengzhi Peng, David C. Chang, Peter Blundell Jones and Bryan Lawson
University of Sheffield

Keywords: Urban Contextual Databank, Dynamic Retrieval, Java, Common Gateway Interface, VRML, HTML, Virtual Cities

Abstract: This paper presents our current development of the Sheffield Urban Contextual Databank (SUCoD) prototype that provides users with a Web-based interface for dynamic retrieval of architectural and urban contextual information of the city of Sheffield. In comparison with past attempts of building Virtual Cities accessible over the Internet, we have experimented with a different system architecture capable of generating VRML models and other related documents *on the fly* according to users' requests. Through working examples, we describe our methods of implementing the data communications between Java applets and Common Gateway Interface (CGI) scripts. The SUCoD prototype has been developed to explore and demonstrate how *user centred dynamic retrieval of urban contextual information* can be supported, which we consider a user requirement of primary importance in its future use for collaborative design and research relating to the city of Sheffield.

1. INTRODUCTION

In the past three years, the Diploma students at the School of Architecture at the University of Sheffield have embarked on the Sheffield Urban Study project, researching and reconstructing parts of the City of Sheffield at around the turn of the twentieth century. In each year's program, the students were organised to work in groups to produce research reports, drawings and scaled physical models of the buildings, streets and places by investigating all sorts of local historical literature and archives. Apart from creating a substantial research database on the Sheffield urban history, the students also learned how to undertake historical research in a collaborative and systematic manner (Blundell Jones, Williams, et al., 1999).

To date, the database resulting from the urban study project covers a region of 40 squares (one square of 200m each side), reaching 160 hectares in total. Subsequently, a public exhibition of the project work has been held at the Mappin Art Gallery in Sheffield for several months. In June 1999, a major research grant from the Arts and Humanities Research Board was awarded to the authors to embark on a separate project with an aim to investigate how the physical database amassed by the Sheffield Urban Study project can be put into electronic form accessible through multiple routes. The project was set out to achieve the following objectives:

- To further evaluate and elaborate the hypermedia-based modelling and authoring methodology established in the previous pilot study (Peng and Blundell Jones, 1999).
- To deliver a prototype of an interactive hypermedia databank that will provide user-friendly facilities for searching and retrieving digital documents created on the basis of the Sheffield Urban Study Project.
- To make the databank accessible nationally and internationally via the Arts and Humanities Data Service and the Internet.
- To demonstrate that a dedicated urban contextual information system of a region can facilitate collaborative research and design regarding the historical or contemporary urban development of that region.

Our pursuit of the research has led to the development of the first prototype named as “Sheffield Urban Contextual Databank” (SUCoD). The SUCoD prototype presents features that are similar to some of the projects related to building “Virtual City” pursued by other research groups. In building the Sheffield model, however, we are particularly interested in developing novel spatial-oriented approaches to retrieving *dynamically* the urban contextual information resulted from the previous research into the city’s history. By adopting the Java™ and Common Gateway Interface (CGI) technologies, we have implemented in SUCoD a Web-based computing framework on which a large amount of contextual datasets can be deposited and maintained constantly. The prototype has been tested to demonstrate the feasibility for users to dynamically retrieve contextual information of any areas of the city through an interactive map interface. Users can browse and navigate city contexts as *dynamic Web contents* that display historical maps, virtual reality models, and other digital resources related to the specific buildings or areas retrieved by the users.

In this paper, we shall present and discuss our current design and implementation of the SUCoD prototype, focusing on our development of its functionality and user interface. The remainder of the paper is organised as follows: In Section 2, a brief review of three related projects on building virtual cities is presented, highlighting the various purposes and intended uses of the virtual cities created. Following a discussion on some of the major system design issues, we describe in Section 3 the stages of developing the SUCoD prototype. Finally, Section 4 reports on the initial responses and feedback from the audiences to whom we have made demonstrations.

2. BUILDING VIRTUAL CITIES: RELATED PROJECTS AND SYSTEM DESIGN ISSUES

In recent years, the constructing of "virtual cities" has become an active research and development topic in architectural computing and urban studies. An important feature of virtual cities is that they are accessible over the Internet and they present users with 3D virtual worlds based on the graphic data format of VRML (Virtual Reality Modelling Language, see [WWW01] for the latest specification of the language). Apart from being Internet-HTTP compatible, VRML provides many other features that modellers can exploit to create rich 3D virtual environments in which the end users have controls over various modes of navigating the models (e.g., walk, fly, pan, rotate, etc).

Applications of virtual cities or virtual environments have been popular in domains where 3-dimensional simulated spaces and architectural metaphors can offer new kinds of graphical user interface. In academic research, on the other hand, the enterprise of building virtual cities seems be driven mainly by its ability to represent/simulate historical or contemporary architectural and urban environments. Past attempts show that constructing virtual urban environments can be an effective way of creating and maintaining a large organisation of online urban and architectural information. Virtual cities can be created to provide users with convenient routes to the models and related resources related to the historical or contemporary aspects of the cities. In the following, we shall briefly look at three related projects.

2.1 Three projects on building virtual cities

The **Bath Model** was created by the CASA group at the University of Bath (Day, Bourdakis, et al., 1996). The aim was to visualise the changes in the City of Bath and how the city was originally constructed. The Bath model was initially built with 3D modelling and texture mapping in AutoCAD, and was subsequently converted into a single VRML model. The model covers a square of 10 km each side with texture-mapped terrain around the city and the Bath abbey in some detail. The model was organised in four levels of detail with data links to images and texts on the city and building history [WWW02]. The Bath model has been used by the local planning authority for development control and public debate on future city development (Bourdakis and Day 1997).

The **Glasgow Directory** was developed by the ABACUS group at the University of Strathclyde [WWW03]. The system was intended to present the city of Glasgow in 3D form with architectural details and a searchable citywide database of her important architectural heritages. A variety of online resources are provided, including 3D volumetric models (in VRML 2), digital photos of architectural interiors assembled in QuickTimeVR™, a map for browsing and retrieving the VRML models, and a search engine for its database of significant buildings of the city. As on the Web, the Directory now contains a total of 47 separate VRML model segments, covering an area of 25 km^2 accurate in height and plan. The Glasgow Directory has been used by developers' visualising proposals in the context of the city and also to promote the City and its architecture via the Web.

The **Virtual Los Angeles** project undertook by the Urban Simulation Team at UCLA aimed to achieve a high quality community and city visual simulation for the City of Los Angeles [WWW04]. By combining aerial photographs with street level imagery, databases of trees, and 3D geometry, realistic urban neighbourhoods were created. The system provides substitution operations for interactive evaluating alternatives and temporal modelling in simulating existing neighbourhoods, historical reconstructions, or proposed new developments. The ultimate goal is to deliver multiple linked VRML models, covering the entire LA basin (10,240 km^2). The project intends to deliver a real-world application capable of supporting hundreds of remote simultaneous interactive users, working in civil engineering, urban design and planning-related problem solving (Snyder and Jepson 1999). There was also the plan of allying the system with other citywide applications such as emergency response, virtual learning environments, etc.

2.2 System design issues

The quick look above is by no means exhaustive. An extensive review of more virtual city projects deserves a separate paper to examine properly the issues and complexity involved. However, related to our interest in building the Sheffield model, the previous projects suggest a number of system design issues of immediate concerns:

a) *VRML and accessibility over the Internet*. The VRML 2 file format was adopted by all three projects in support of wider user accesses through the Internet and the Web. Given that special software for displaying and interacting with VRML worlds are now widely available as free Web browser plug-ins, the acceptance of VRML in building and accessing virtual cities seems here to stay for sometime to come. To our knowledge, no alternatives have been proposed to replace the VRML as a non-proprietary file format for representing 3D worlds on the Internet. Rather recently, the Java 3D™ technology has evolved into a full strength programming language for building network-centric and scene graph-based 3D applications. Java 3D, however, does not define a file or network format of its own; it is designed to provide a lower-level underlying platform API, and many VRML implementations are expected to be layered on top of Java 3D [WWW05].

b) *3D models and other data types*. All three projects have shown that a 3D model is not the only type of dataset required in building virtual cities. Other types of data such as maps, photos, textual documents, and video/ audio clips, can be as important, as they provide aspects of urban information that cannot be recorded effectively in 3D modelling.

c) *VRML navigation and information retrieval*. Interlinking the different types of digital resources created is an important issue. A combined use of the hyper linking in HTML and VRML facilitates users' navigation and retrieving information with a spatial context. The Glasgow Directory clearly demonstrates the usefulness of this facility.

d) *Spatial subdivision in urban modelling*. Subdividing a city region into smaller areas seems necessary in managing the modelling process. Building a city region into a single large file poses at least two problems: difficulties in maintaining the model, and the lengthy download time, which might put off users of the models. The Bath model seen above is an example of a single-file approach with no subdivisions used. Both the Glasgow and Los Angeles models imposed spatial subdivisions, resulting in multiple VRML models that users can retrieve separately. However, we did not find explicit reasons given as how and why the subdivisions were introduced in the first place.

In comparison with the city modelling projects seen above, our Sheffield modelling project is intended to support collaborative research and design in relation to the city contexts. This implies that we should provide not only an adequate urban data repository of a good quality but also a flexible way of retrieving the contextual information for any area of the city. After all, what constitutes an urban context that is relevant or irrelevant to the objectives of a design or research project is open to the participants' interpretations and analyses; it is almost certain that different individuals will define and delineate differently what an urban and architectural context actually consists of (for instance, location of focal point, boundary, shape, and viewpoints, to say a few). It is therefore, in our view, a critical requirement that the Sheffield model should be capable of allowing users to retrieve 3D models and other related contextual resources according to user-defined locations and boundaries of relevance.

Looking further into the requirement of user-centred dynamic retrieval, we realised that while satisfying the need for speedy data transmissions through the Internet, if the separate model sets cannot be combined or re-assembled in ways as requested by the users, the usefulness of these models can be drastically reduced. For example, there will be difficulties if the users' focal points of interest happen to lie on or somewhere near at the lines of subdivision where separate models actually end or meet. We think a better result can be achieved if the users are able to retrieve 3D models together with related information in which the focal points and boundaries requested are located at positions intended by the users.

The points raised above imply that a system should to be capable of generating the models and related information sets “on the fly” according to individual users' retrieval requests. We believe that the urban contextual models and information retrieved in such a manner should better serve the users' needs of contextual studies and modelling. A true test of the usefulness of the online resources that are going to develop lies in whether the contextual information retrieved can serve the users' construction of their own design arguments and narratives. According to our present review, we have not yet seen any virtual cities published on the Web in which users can perform dynamical retrieval of 3D models. In the next section, we shall present how we have achieved this functionality in the Sheffield project.

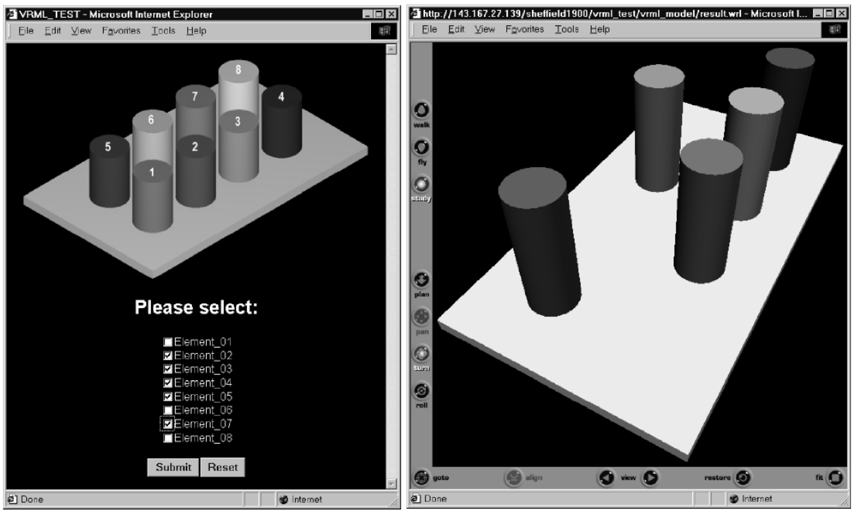
3. DESIGN AND IMPLEMENTATION OF THE SUCoD PROTOTYPE

Our system development at the early stage was most concerned with how we might build a system that would provide the functionality allowing for dynamic retrieval of 3D models in VRML 2.0. We started off with a simple example on the basis of an HTML form sending messages to a Perl script through the Common Gateway Interface (CGI) protocol. We then progressed to further extended examples by replacing the HTML form with Java applets.

3.1 A simple cylinder-on-board example

Our first working example considered a 3D model of eight simple cylinders sitting on a rectangular board. By selecting particular cylinders, a user can retrieve VRML models, showing only the cylinders selected together with the board. The task was simulated in an HTML form, which presents an image of the 3D model, eight check boxes for selecting the cylinders, and "*Submit*" "*Reset*" buttons. A CGI script was written and deployed on a server machine, which can be invoked by user's pressing the *Submit* button on the HTML form. On receiving a request sent from a client's HTML form that specified which cylinders were picked (including none), the CGI script was executed to generate a VRML model, containing only the selected cylinders. The VRML model was then transferred and loaded to the user's Web browser when he or she clicked on the hyperlink pointing to the resultant VRML model generated on the server.

Figure 1 is a screen shot of the cylinder-on-board example, showing the HTML form (left) and the retrieval of a VRML model (right) in two separate Internet browser windows. The VRML model was viewed in the Cortona VRML Client 3.0 (by ParallelGraphics [WWW06]).



Please select:

□ Element_01
☑ Element_02
☑ Element_03
☑ Element_04
☑ Element_05
□ Element_06
☑ Element_07
□ Element_08

VRML_TEST - Microsoft Internet Explorer
File Edit View Favorites Tools Help
8
7
4
6
3
5
2
1
Submit
Reset
Done
Internet
http://143.167.27.139/sheffield1900/vrml_test/vrml_model/result.wrl - Microsoft I...
File Edit View Favorites Tools Help
walk
fly
study
plan
pan
turn
roll
goto
align
view
restore
fit
Done
Internet

Figure 1. Exploring dynamic retrieval of VRML models in the cylinder-on-board example. On the left is the HTML form for users to select numbered cylinders (0-8) via the eight check boxes. For each selection submitted, a VRML model is generated *on the fly* by a CGI script on the server; the VRML model is then displayed to the user in a separate window (right).

Our exploring the above example appears significant in several aspects:

a) All the VRML models were generated on the fly upon receiving client's requests sent through the HTML form interface. The server did not store any pre-built VRML models (if it did, we would have to pre-build a total of 256 files of static VRML models on the server to be capable of responding to all possible user selections).

b) CGI programming with Perl 5.6 (as in ActivePerl 5.6 provided by ActiveState [WWW07]) worked fine for our objectives of processing user requests and constructing dynamically well-formed VRML files. A VRML file can be generated by a Perl script that matches and combines data (DAT) files in which the basic 3D data of individual elements or objects (i.e., the cylinders and board in our example) are stored.

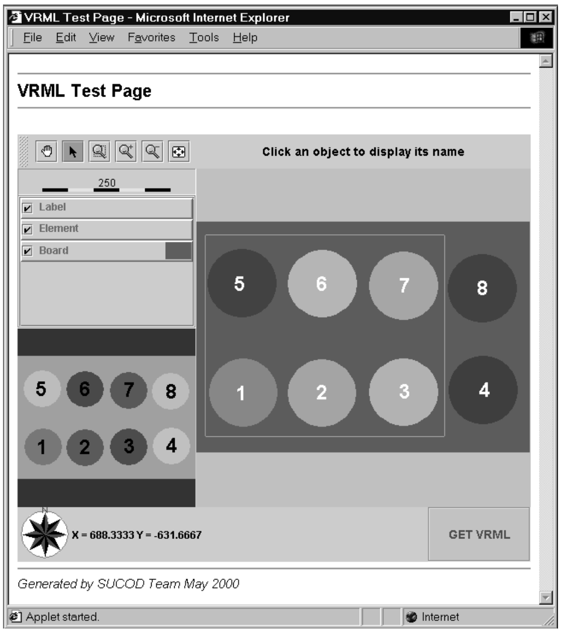
c) The cylinder-on-board example may look trivial but it did bear some degree of correspondence to our modelling of a city environment. Basically, the cylinders refer to individual buildings, and the board to a city's terrain. Although we did not subdivide the board into smaller blocks, we were sure it could be done just like the cylinders. The entire terrain of an urban area, therefore, can be subdivided into smaller elements and retrieved separately just like individual buildings.

With two machines acting as a server (a Pentium III running the Internet Information Server 2.0 under Windows NT 4.0) and a client connected to the Internet through the Ethernet at the School, the above working example proved our initial idea of dynamic retrieval workable. The generating and display of VRML models across the network has been reliable. But two immediate issues were sensed following the working example:

1. The check-box type of interface will not be usable if the model is extended to include, say, a hundred cylinders. How is a user supposed to deal with one hundred check boxes for selecting or de-selecting particular objects in order to retrieve a VRML model? This issue is important as a city model typically contains thousands of objects if not more. An alternative interface has to be sought to enable user selection on a large sum of individual elements.
2. What will happen to the speed of dynamic generation of VRML models if the amount and complexity of elements far exceed the current working example? To find out more, we have worked on further experiments.

3.2 An extended cylinder-on-board example

Given our purpose of modelling the historical city of Sheffield, we considered it logical to move onto a map-based interface for user selection. With a digital map, a user can simply point and drag a rectangle on the map as a way of making selection. All elements located within the rectangle are interpreted as being selected. There was also the need of basic map-reading operations such as *zoom-in* and *out*, *pan*, *fit all*, etc. In our search for digital map building and viewing tools, we considered a Java^{TM} enabled mapping method worth trying. Basically, a Java-enabled map is built on Java's Applet technology that provides a client-side graphical user interface for displaying and interacting with object-oriented vector-based maps loaded into the applets. To avoid implementing a Java-based map system from scratch, we chose to work with the ILOG JViews Component Suite [WWW08] as the basis of building our own Java-enabled map applications. Built with ILOG JView's Maps Beans, an applet was first implemented to display a multi-layered interactive map. The applet provides a viewing area for displaying the Java-enabled map composed in ILOG's Composer package. *Figure 2* shows the Java map applet running the previous eight-cylinder example. Several graphical user interface (GUI) components, including the *Control Bar*, *Map Legend*, *Overview* and a *Coordinate Viewer*, were implemented to provide the interactive functions of manipulating the map (for instance, *select*, *zoom in* or *out, fit all* in the *Control Bar*, check or uncheck particular layers of map in the *Map Legend*, and a fast map navigation in the *OverView* panel).



VRML Test Page - Microsoft Internet Explorer

VRML Test Page

Click an object to display its name

X = 688.3333 Y = -631.6667

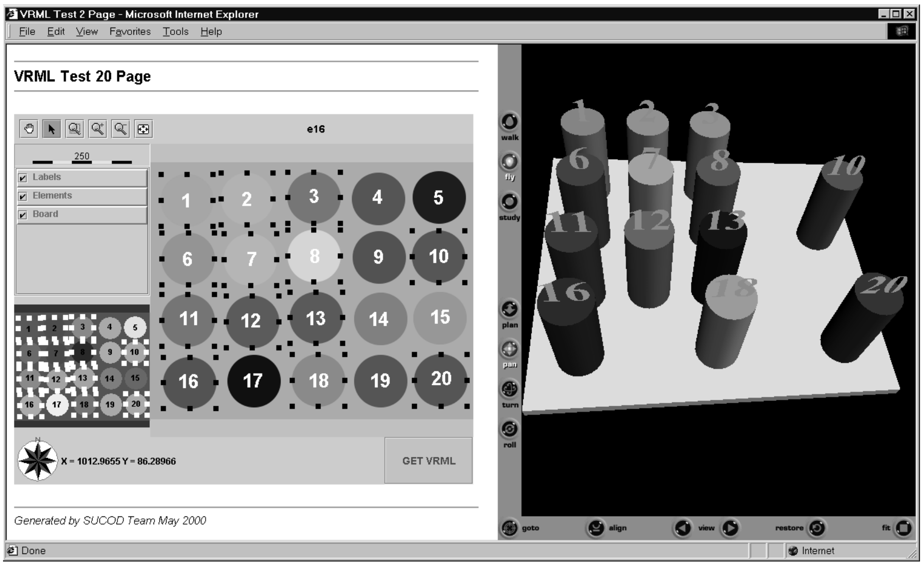
Generated by SUCOD Team May 2000

Applet started.

Figure 2. The *VrmlTest* Java applet was developed to experiment with a multi-layered interactive Java-enabled map to communicate with the CGI script. The cylinders can now be selected on the map by the *drag a rectangle and select* method.

However, the idea of a Java-map applet would not work if there were no communications between the map applets and the CGI Perl scripts. As shown before, the HTML form-based communications with the CGI script was essential in dynamic generation of VRML models. A way of establishing the communications between our Java map applets and the CGI

Perl scripts must be found in order that the entire scheme could succeed in achieving our aim. After numerous coding attempts, we finally had the Java code with which the two-way data communications between the Java map applet and the CGI script was established. The "*GET VRML*" button is the GUI component for invoking the Java-CGI communications. With this functionality in place, a user can now make selections on the basis of a map and VRML models will be generated according to what objects are selected within the Java map management environment. *Figure 3* shows our current implementation of the extended cylinder-on-board example. Note that the Java map and the VRML model are now shown in two frames (left and right) on a single Web page.



VRML Test 2 Page - Microsoft Internet Explorer

VRML Test 20 Page

☑ Labels
☑ Elements
☑ Board

X = 1012.9655 Y = 86.28966

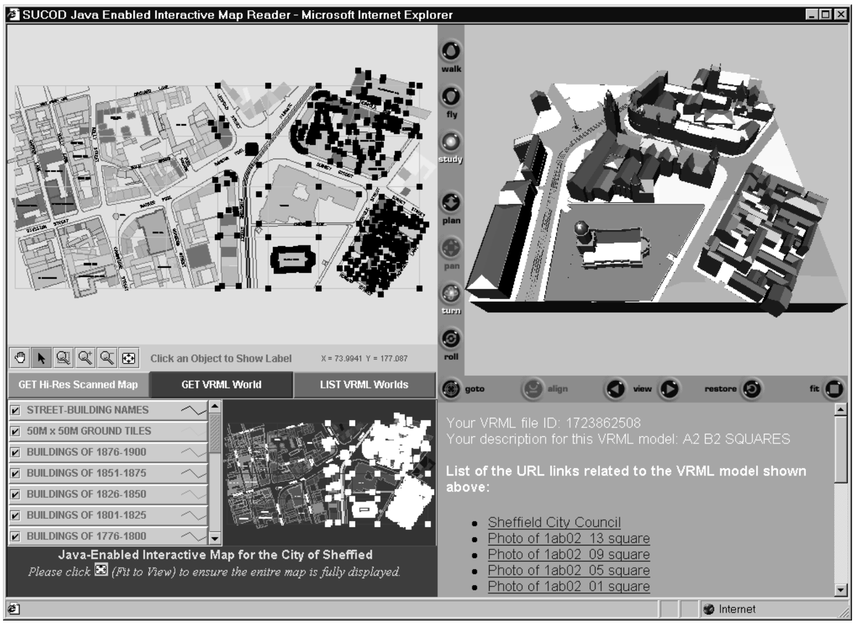
Generated by SUCOD Team May 2000

Figure 3. The extended cylinder-on-board example (increasing the number of cylinders to 20). The left frame is for selecting elements on the Java map and the "GET VRML" button is for invoking Java-CGI communications. The right frame is for displaying resultant VRML models.

Again, in the extended example, there were no pre-built VRML models stored on the server awaiting calls from clients. As before, VRML models are generated on the fly by invoking the Perl script hosted on the server. No changes were required in the Perl program when we switched from to the HTML form to the Java map applet. However, it did take considerably longer in downloading the applet as it contains all the Java GUI classes needed for initialising the applet in a Windows environment. In terms of generating and displaying VRML models, the Java-map approach did not show any significant delay in comparison with the HTML form-based retrieval.

3.3 Work on the SUCoD prototype

Having worked through the two cylinder examples, our next phase of system development was to populate the Java-CGI system framework with real datasets acquired in the Sheffield Urban Study project. Two squares (labelled A2 and B2 originally) were chosen in trying out our first Sheffield Urban Contextual Databank (SUCoD) prototype. The datasets created consists of the following types: (i) the Java interactive map, (ii) data for generating VRML models, and (iii) data for generating HTML pages. *Figure 4* shows a screen shot of the SUCoD prototype when accessed in Microsoft® Internet Explorer 5.



Your VRML file ID: 1723862508
Your description for this VRML model: A2 B2 SQUARES

List of the URL links related to the VRML model shown above:

- Sheffield City Council
- Photo of 1ab02_13 square
- Photo of 1ab02_09 square
- Photo of 1ab02_05 square
- Photo of 1ab02_01 square

Figure 4. The SUCoD prototype runs in Microsoft Internet Explorer 5. The left frame is the applet for managing the Java map, the right-upper frame for loading VRML models, the right-lower frame for listing links to HTML pages related to the VRML model displayed. On the Java map, “graphic handlers" in tiny black squares are shown for every object selected.

In implementing our first release of the SUCoD prototype, we made the following design decisions on the data types and organisations to be supported by the Java-CGI framework:

a) **The spatial subdivision scheme**. The original spatial subdivision in the Sheffield Urban Study project was a grid of 200m squares across the region being studied. All the physical modelling and paper-based documentation tasks were coordinated in accordance with the grid system. In developing the SUCoD prototype, we decided to adopt a grid of 50m squares so that a smaller granularity for user selection could be achieved. Another reason for adopting the smaller square unit is to coincide with the UK Ordnance Survey Mapping Service which publishes digital urban maps (OS Super Plan®) in 50m by 50m grid [WWW09]. This means, in SUCoD, the smallest unit a user can select is a 50m-square (i.e., a single "ground tile" see below). However, the unit of 50m-square only applies to the subdivision of terrain. Every single building shown on the Java map was treated separately so that a user can pick and un-pick all buildings individually.

b) **The layering of the Java map**. The Java-enabled map was an important aspect of data design, as it should accurately reflect the research conducted in the Sheffield Urban Study project. It should also provide the interface for users to navigate and select areas of interest according to different criteria. By gathering all the data available, we have come up with a Java-enabled map consisting of 20 layers, each of which can be turned on and off separately when shown up in the applet. A total of 7 layers were created to depict buildings according to the various periods of Building Ages (ranging from pre-1750 to 1900) as researched in the Sheffield Urban Study project. Another 8 layers were created to indicate buildings of various Building Usages (Public, Residential, Heavy Industries, etc). A layer named "50m Square Ground Tiles" was created to show the 50m square grid plan as imposed onto the original A2 B2 squares. Like the buildings shown on the various layers of building ages, the "ground tiles" are objects that users can select to retrieve VRML models representing the 3D terrain and streets on it. Finally, the names of all buildings and streets as indicated in the historical maps were added as the top layer to aid user navigation.

c) **VRML models of terrain and buildings**. To carry out sophisticated 3D modelling of the terrain, streets, and buildings, the MicroStation/J™ CAD package was used to generate the basic 3D geometric and colour rendering data. These were then taken as the raw data for a new Perl script to construct a VRML model. Apart from the smallest granularity of selection allowed (i.e., a single building or a single 50m square ground tile as depicted in the interactive Java map), there are no other constraints on what a user can select and retrieve. It is entirely up to the users to decide what buildings or terrain units should be included when retrieving VRML models. From the system's point of view, it is perfectly legal to generate a VRML model with only buildings without the terrain underneath and visa versa. A user's selection may also result in a VRML world of disconnected terrain units, which do not add up to form a rectangular boundary.

In the course of developing the SUCoD prototype, we are aware that more advanced technologies like Java Servlets™ and JavaServer Pages™ (JSP) have been put forward to replace the CGI approach in implementing a typical Internet-based server-client application (see Hunter and Crawford, 1998; Hall 2000, for example). Our primary objectives at this stage, however, are to explore and evaluate the kinds of system functionality both from the data creation as well as usability points of views and to deliver a data repository holding contextual resources of a high research quality.

4. PROTOTYPE DEMONSTRATIONS AND FEEDBACK

In the past few months, we had the opportunities to make demonstrations of the SUCoD prototype to students, staff and visitors at the School of Architecture. The initial responses from the audience have been largely positive. Following the feedback, the prototype has been continuously revised in response to several points raised during the discussions:

1. *Speed of downloading and reliability*. During the demonstrations, we experienced a lengthy waiting time for initiating the Java map applet. There was also the problem of reliability: from time to time the system seemed not to be responding for reasons not entirely sure to us: networking congestion during the demos, the capacity of the Web Server in use, or the capacity of the client machine in use, etc. In our attempts to ease the problems of speed and system reliability, we decided to migrate SUCoD to the JAVA™ 2 Platform with the latest Java Runtime Environment provided in the JDK™ 1.3.0_01 [WWW10]. With the installation of the Java™ Plug-In 1.3.0_01 [WWW11] on a client site, the change has resulted in a shorter applet initialising time and more reliable performance during interactive sessions.
2. *Relating to contemporary urban context*. Several people have raised the point regarding the contemporary urban context of Sheffield. They suggested that the SUCoD system would not be very useful if it contains no contemporary contextual information. The Sheffield Contextual Databank project was funded to create online resources based on the Sheffield Urban Study project. However, we believe that we have come close to establish a generic system framework in developing the SUCoD prototype. Adding contemporary contextual data should not render the

current framework obsolete but can be treated as an extension to the existing databank. The SUCoD prototype has demonstrated that contextual data acquisition and accumulation can be sustained in a bottom-up manner whether looking into the past, present, or future.

3. *Scanned historical maps*. In studying the Sheffield city history, historical maps have been used extensively to establish the footprints of the historical urban fabric. In constructing the multi-layered interactive Java map and the 3D models, historical maps played a crucial role in laying the foundation. Suggestions were made to us that the historical maps could also be made available through the SUCoD system. We recognised this as an important resource in addition to the VRML and HTML files. Further work on SUCoD is now underway to include a facility for dynamic assembling of scanned historical maps according to user selection.

5. REFERENCES

Blundell Jones, P., A. Williams and J. Lintonbon, 1999, "The Sheffield Urban Study Project", *Architectural Research Quarterly*, 3(3), p. 235-244.

Bourdakis, V. and A. Day 1997, "The VRML model of the city of Bath", in: R. Coyne, M. Ramscar, J. Lee and K. Zreik (eds.) *Design and the Net: Proceedings of the Sixth International EuropIA Conference*, p. 245-259, 2-3 April 1997, Edinburgh, UK.

Day, A., V. Bourdakis, and J. Robson, 1996, "Living with a virtual city", *Architectural Research Quarterly*, 2(1), p. 84-91.

Hall, M., 2000, *Core Servlets and JavaServer Pages (JSP),* Prentice Hall PTR/Sun Microsystems Press: New Jersey.

Hunter, J. and W. Crawford, 1998, *Java Servlet Programming*, O'Reilly UK, London.

Peng, C. and P. Blundell Jones, 1999, "Hypermedia Authoring and Contextual Modeling in Architecture and Urban Design: Collaborative Reconstructing Historical Sheffield", in: Osman Ataman and Julio Bermudez (eds.) *Media and Design Process: Proceedings of ACADIA'99*, p. 116-127, 28-31 October, Salt Lake City, U.S.

Snyder L. M. and B. Jepson, 1999, "Real-Time Visual Simulation as an Interactive Design Tool", in: Osman Ataman and Julio Bermudez (eds.) *Media and Design Process: Proceedings of ACADIA'99*, p. 352-353, 28-31 October, Salt Lake City, USA.

[WWW01] "http://www.vrml.org/VRML2.0/FINAL/" The Virtual Reality Modeling Language Specification, Version 2.0.

[WWW02] "http://www.bath.ac.uk/Centres/CASA/" The Bath Model.

[WWW03] "http://iris.abacus.strath.ac.uk/new/gintro.htm" The Glasgow Directory.

[WWW04] "http://www.ust.ucla.edu/ustweb/projects.html" The Virtual Los Angeles Project.

[WWW05] "http://java.sun.com/products/java-media/3D/" Java 3D™ 1.2 API.

[WWW06] "http://www.parallelgraphics.com/products/cortona/" Cotona VRML Client.

[WWW07] "http://www.activestate.com/Products/ActivePerl/" ActivePerl 5.6 by ActiveState.

[WWW08] "http://www.ilog.com/products/jviews/" ILOG JViews Component Suite.

[WWW09] "http://www.ordsvy.gov.uk/home/index.html" Ordnance Survey.

[WWW10] "http://java.sun.com/j2se/1.3.0_01/index.html" Java^{TM} 2 SDK, Standard Edition (J2SE) Version 1.3.0_01.

[WWW11] "http://java.sun.com/j2se/1.3.0_01/jre/index.html" Java^{TM} 2 Runtime Environment (J2SE) Version 1.3.0_01.