The Future of VRML on Large Urban Models

Vassilis Bourdakis

Centre for Advanced Studies in Architecture (CASA), University of Bath, Claverton Down, BA2 7AY, Somerset. UK

Tel: *+44 (0) 1225 826475*
Fax: *+44 (0) 1225 826691*
e-mail: *V.Bourdakis@bath.ac.uk*

Abstract: In this paper, the suitability and usability of VRML for urban modelling is discussed based on the experience gained by the creation of urban models of varying sizes, level of detail and use. Areas where the current VRML specification is failing to satisfy the needs of large-scale urban modelling are highlighted. Properties of a successful alternative or VRML3.0 specification are suggested and briefly discussed.

Keywords: VRML, virtual word design, digital urban models.

1. Background

During the past six years a series of three-dimensional urban computer models have been developed in the Centre for Advanced Studies in Architecture (CASA) at the University of Bath. Computer models were constructed in order to assist in making the planning and development control process more flexible by providing a means by which planning proposals could be visualised and alternative schemes for a site compared; allowing non-experts to comprehend the implications of proposed changes. Developments in the domains of Virtual Reality (VR) and the Internet triggered a series of research projects carried out in CASA to tackle the issues of interactivity, structuring and management of large urban databases.

In 1991 CASA received a grant to construct a 3D computer model of the city of Bath. The model was constructed from aerial photographs using photogrammetry, is accurate to less than half a metre and covers the whole historic city centre, an approximate area of 2.5x2.0

km (Bourdakis et al, 1997). During 1995, and using similar construction techniques, the core of London's West End covering 1.0X0.5km was also modelled followed by Gloucester City centre in 1996 (covering an area of 750X500metres). In the 3rd UKVRSIG conference, the problems faced in trying to construct a VRML model of the city centre of London and more generally, the distance that separates CAD and CAAD from VR were discussed (Bourdakis, 1996). Almost 18 months later, a very small number of the mentioned problems are solved; most remain and a few extremely inefficient and often unacceptable solutions have been suggested.

The current CASA research work is focusing on delivering integrated environments for urban planning, design, decision making, and learning. Having spent almost two years following the developments, learning, using and teaching VRML, its future looks very unclear and unsettled, as far as large scale models are concerned, much worse than expected when one deals with the leading edge of technology and software developments.

In the following sections the particularities of large scale modelling specifically at an urban level is analysed. Subsequently, a comparison of VRML versus proprietary VR systems—focusing on large-scale models—is carried out and the limitations of VRML are highlighted. Finally the future of VRML for urban modelling and proposals for the next release of VRML or of a successful alternative are discussed.

2. Urban Scale Model Specifications and Particularities

Currently available VR software solutions have a fairly clearly defined upper limit of geometry size that can be handled successfully which is quite low and unsuitable for urban scale modelling. In general, software developers and VR designers/artists recommend replacing geometry with simplified repetitive texture mapped shapes. However, this approach can only be successfully implemented in certain types of models and it seems to be producing acceptable results in American towns, skyscraper filled city centres high rise office blocks and generally highly repetitive environments. In the process of converting the existing CAD model of Bath, it was attempted to follow a similar approach. After a careful classification of the existing types of properties, positioning, roof geometry, etc a list of over 200 types of properties was developed. Bearing in mind the generalisations that took place in creating this "restricted" list, the projected utilisation of the model and the inevitable downgrading of the available data quality, it was decided to discard this approach. Repetitive elements indeed exist in urban scale models but due to the level of detail of the available data it was restricted to building elements: windows, doors, cornices and chimney stacks as well as street furniture, trees etc.

The urban computer models built by CASA are used for urban planning applications and relevant visualisations where specific views from points of interest and occlusion tests are of vital importance. Therefore, accurate representation of not only the footprint of a building but of the roof geometry is needed. Consequently the actual detail on the buildings' facades has to be accurately represented and cannot be successfully imitated by photographic techniques which generally look better in head on views but alienate the designer and visitor from typical street level oblique angled perspective views.

It is interesting to note that architectural and urban design is full of assumptions and conventions to an extent not applicable to many other disciplines. That has always been the case with paper based drawings considering the fact that 3D information (the buildings) had to be expressed and communicated on a 2D medium (the drawing). This approach has worked quite well with paper drawings since professionals are trained to read such drawings quickly and with minimum effort, average performance is obtained using a computer though. The main reason being that one has to decide and either imitate the paper based approach and thus do everything in 2D or advance to 3D design. Opting for the second, means that a new set of rules and maybe even conventions have to be invented/developed and the professional has to have an even better understanding of 3D space and how to design in it. Although 3D CAAD packages have been around for a long time, there are still quite a few problems regarding the 3D design stage. No matter which option is adopted, the final product (the building) will be based on paper drawings—almost exclusively 2D ones. In creating a fully 3D interactive Virtual Environment (VE), due to lack of architectural / urban planning experience in the field (with the exception of work carried out in the UCLA Urban Simulation Team AUD, ART+COM and a few others), VR conventions and methodologies are adopted. Furthermore, a close examination of the techniques the above mentioned groups use, reveals that VR methodologies are taken for granted and therefore carefully tuned and appropriately built models are employed rather than custom tools for urban scale modelling being developed.

The above described situation is far from ideal on a building scale, once it expands to urban scale it becomes even worse mainly due to the sheer amount of information that needs to be handled. The 2D-paper version of an urban space with its special conventions could easily handle issues of data size without the need for complicated and difficult to implement Levels of Detail (LODs).

Architects and planners have a concept of LOD but on a different level based on the paper scaled representation of real space (1:5000, 1:1250, 1:500 etc). Projects very often are seen in scales up to 1:100 or 1:50 for a building, which means a building is isolated and examined at a higher level of detail. However, there is no such concept as spatial structure within different levels of detail. The whole area is sequentially "worked" at different levels of detail but it is never visualised with varying levels of detail at the same time (maybe VR's way of "seeing" the environment is going to be accepted and approved by engineers - it is simply the way the paper based representations are structured that causes this behaviour). The closest concept to organising information in building practice is to use BLOCKS/ INSTANCES / GROUPS where a set of objects are joined together, defined as one and used in various other places without the need to redefine their geometry. Very often annotations are used to enable project management and automate the creation of bill of quantities etc. In VRML2.0 terms they are a cross between PROTOTYPES and DEF / USE. Another way is to use LAYERS where objects of different types are organised (and in many cases worked) together. Often, different designers work on different issues (landscaping, interior, structures, parking etc) concurrently, the layering systems certifies that there will not be any overlap.

3. VRML versus Proprietary VR Systems

VRML has a long list of advantages compared to other VR systems for general applications. Starting with, it is freely available, although in certain cases it may prove to be more expensive to run than proprietary systems. The reason being the lack of model creation tools and in many cases the need to consider a new hardware/software platform for VRML development alone. However, being the emerging standard for geometrical description of 3D spaces (rapidly replacing AutoDesk's DXF from the position it has held for almost a decade) means that most if not all 3D modelling packages are and/or will support it. It should be noted that many implementations and translators currently available create quite inefficient VRML files (polygon based instead of primitive based, no texture co-ordinates, whereas traditional CAD software translators ignore material properties, lighting, etc.). Its tight integration to the Internet, the actual fact that it was initially developed to supplement and eventually replace the 2D interfaces we currently use on the WWW (Pesce, 1995), means that the scope of applications that can be created is much greater than when using conventional toolkits. Additionally, it has managed to attract and to a large extend excite a market that was not particularly aware of VR and its potential. The new PC hardware accelerated 3D graphics cards developed for 3D games have also been beneficial to VRML especially since a large driving force behind VRML is focusing on the Internet games market (Sony, SGI etc). Finally, VRML can be used for data mining and subsequently pseudo-Geographical Information Systems (GIS) can be developed. Overall, VRML had a negative impact on the VR industry in general, the only positive influence is that it has created a stir and increased public awareness on VR in general.

As far as the urban modelling work carried out in CASA is concerned, none of the commercially available VR tools are suitable. Interactivity and realism are the two main focal points of all large scale models built up to day (Liggett et al, 1995, ART+COM) and the tools used are therefore not suitable for CASA's work. Large scale VR solutions available, are generally based on SGI hardware using either Performer or customised Iris Inventor applications in certain cases supporting navigation modes, frame constrained distance and importance based LOD culling (The Martin Centre, Cambridge University). Polytrim, a landscape design tool developed by the Centre for Landscape Research (CLR) at the University of Toronto, is also an interesting approach but tightly focused on its initial aim. On a different field, CADCentre in Cambridge have developed an again SGI Inventor based application for offshore design/ management which can handle impressively large datasets.

Concluding, regarding the model size limitations imposed by VR applications, it seems that only SGI Inventor and Performer as well as VRML are the only ones that can handle the size of models involved in urban scale. It should be mentioned that this to a large extend is due to the fact that the above-mentioned applications are not really VR tools with fully developed behaviours, interaction mechanisms, VR hardware support etc but more expandable 3D geometry description languages.

4. Limitations of VRML for urban scale modelling

The limitations of VRML for large models and urban scale models in particular are very substantial and can be broadly classified into two main categories, the *design* related and *implementation* related ones. The former are actual design flaws, omissions and limitations of the VRML specification as it currently stands in Release 2, while the latter are, in theory at least, issues that are to the discretion of the software designer to implement and support.

Starting with the **design limitations**, the lack of support for VR hardware input devices, in particular 3D Mouse, Head Mounted Displays (HMD), Trackers etc, not only in terms of device drivers but also in supporting high degree of freedom input is limiting the usability of VRML since navigation is exceedingly difficult and unintuitive. Furthermore, the whole VRML interface design is developed based on a desktop 2D input metaphor making it almost impossible to customise and improve.

VRML ability to handle the sheer size of the models involved even with careful use of LOD and INLINES is not satisfactory. Software designers must understand that large triangle models should be supported more efficiently. There is a general trend to develop, test and try to market products based on very small models (usually primitives based). As a result, when tested with larger models they fail to perform acceptably even on very powerful graphics workstations. The recommendations when the model size issue is discussed are to use INSTANCEing, often accusing a world designer that fails to do so, rather than understanding the reasons that determine one not to do so. Furthermore, accuracy is not highly valued in VR circles and the need for 3 or more decimal points is often seen as excessive and wasting resources. Current state of the art VRML examples demonstrate a focus on small scale projects, in many cases all geometry tied together in one file, great use of primitives etc which only suits the commercial end of the market (games, Internet based commerce, corporate images etc)

Together with the inability to scale up, the layering of geometry in separate files (INLINEing) is restricting access to behavioural engines. Script triggering controls cannot access resources packaged on different files of the same project leading to awkward file structure and mandating certain design decisions (very often unsuitable and unwanted). Furthermore, certain tasks can simply not be performed unless all geometry is within the same INLINE. On urban scale modelling, where a low-level city model (30-100 triangles per urban block) is initially loaded with all extra levels of detail organised under different files (based on their respective urban block) and INLINED to it, is simply not possible.

Programmatically configurable navigation / interaction / editing is crippled by marketing and political decisions as well as actual limitations in order to make the impact and persuade / dominate the VR and Internet market. However certain decisions were admittedly based on previous experience of the VRML engineers. Nevertheless, the concerns on VRML language capabilities and limitations has resulted in a series of recommendations by the VRML architects that have not been welcomed. The fact that LOD ranges cannot be modified interactively from the scripting / java interfaces affects badly potential urban planning applications. PROTOTYPEs as the be all and end all solution are also quite problematic and unsatisfactory. Incomplete Collision Detection implementation complicates certain interactions, requiring disproportional effort by the world designer. Furthermore the inefficient programming interface, ROUTEing and SCRIPTING and again prototyping limitations result in "spaghetti" coding, extremely complicated and thus inefficient and slow interactions.

The lack of copyright protection or digital signature stamping in the geometric definitions together with the ease of transferring files across the internet is creating even more problems forcing 3D model developing companies to ignore VRML at best and even make sure that their tools will not read or export in this format.

The **implementation related limitations** of VRML are less important and conditions are likely to improve with time without the need for a new standard. To a large extend most of problems are due to VRML Architecture Group (VAG) decision making in the behavioural support design. As a result, programming implementations are poor and too slow once there is more than one SCRIPT node running on a small file. Subsequently, time event loops are extremely inefficient and prohibitingly slow. A clear example of the later is a bus tour animation developed for the Bath model as a means of introducing the city to first time visitors. Using INTERPOLATORs and TIMERs, it slows all interaction and navigation within the Bath city model by a factor of five. Furthermore, optimising the INTERPOLATOR nodes (by creating a series of POSITION and ROTATION ones only) is almost impossible since the animation was created by a CAD software translator and as such is extremely inefficient and non-configurable. Needless to say that accurately following the street contour and undulations maintaining a constant speed on a hand coded animation node is not possible.

VRML memory model is inefficient, paging mechanisms vary from browser to browser and overall perform poorly when the model is larger than the available local resources. As an example, in the current version of the Bath model which is in excess of 700 thousand triangles (15-20MB of uncompressed ASCII VRML2.0 files), a 224MB Max Impact R10000 SGI grinds to a halt if half this geometry is loaded/visited. Bearing in mind that the current version of the model is approximately one quarter of the existing CAD model and with careful use of LOD at any given point no more than 40-60thousand triangles are visible one wonders how will the final model will perform. Furthermore, hardware seems to be badly tuned for VR related tasks—typically available hardware provide either good CPU performance and low graphics (PCs), or the opposite (SGIs). A very powerful, well-integrated affordable system is needed with exceptionally fast CPU-memory subsystem and good software support.

VRML and all VR applications are, as explained earlier, are to a great extent customised for repetitive geometry, tasks, behaviours etc. Once that fails to be the case, VR apps tend to be inefficient and slow. This is clearly demonstrated on landscape representations in urban models. The Bath model covers an area of 5 square kilometres with an average mesh resolution can easily reach tens of thousands of control points. As a result visual tricks and complicated programming techniques have to be employed to decrease the number of polygons needed to create a satisfactory accurate representation of the terrain. Data streaming and / or scripting can be used to dynamically generate terrain models and other large size elements from spatially partitioned co-ordinate sets that cannot be represented using LODs. In many cases memory management is the limiting factor of such techniques.

The problems of security related to Java and the Internet seem to be escalating and restricting VRML capabilities more than initially regarded. Constant developments and changes in Java do not facilitate in improving the overall programming image of VRML. The introduction of Java 3D from Sun and Microsoft's decision to discard Java and switch to proprietary applets is not going to be helping VRML either. Due to all these restrictions, VRML cannot be used as an interactive editing tool / whiteboard support tool for video conferencing which would be extremely helpful in urban planning meetings and evaluation sessions with remotely participating members and critics.

Navigation is another issue that needs serious consideration. In paper based environments, everything is right in front the engineer's eyes—the only tool needed is a large drawing board. On the computer screen, in non-VR approaches, the screen is viewed as a "window" to a much larger drawing, plan view is the one used most of the time. In a VE, the screen size or HMD limitations force us to an approach closer to the CAAD one. However, a plan only view is unacceptable, walking on ground level confusing and generally disorienting. Again new techniques and conventions have to be defined (closer to VR systems approach to interaction).

What is needed for urban scale models is a cross between a clear and well structured programming interface on top of an object structure that is expandable, scaleable and where sharing of information is enhanced rather than inhibited. Instancing of geometry and behaviour along the various files that the model is organised in should be supported efficiently. VRML is a generic geometric description language, trying to cover all shortcomings using programming languages but it doesn't allow enough control over the infrastructure of the file format that would enable customisations for different applications.
Java3D seems to be an alternative, although it will most likely have problems in the amount of information it can handle. Politics are likely to play an important role in the development of internet tools during the next few years so it will always pay off to try and be as independent as possible with the ability to change from one environment to another at minimum cost.

The approach followed in CASA is to be open to new development and suggestions, knowing that VRML (unless extensively modified) will not cover all our research needs. Therefore, all modelling and editing is carried out in a CAD package, with extended entity information providing all supportive data needed for a VR model. Consequently, new approaches can be tested and applied with minimal work replication by simply developing a new translator to the proposed standard/language.

Bearing in mind the above-mentioned problems, the approach followed in CASA is maybe not suitable to available technologies and hardware level. Early in the process, it was decided to avoiding having a large amount of information organised hierarchically in different levels of detail and used as needed, but keep one full level of detail version available and generate the needed levels of detail from it. This makes the process of keeping the urban model up to date easier by only having to update one version of each building or urban block. However, it seems that the VR files generated from these CAD models have too much detail to be useable. Simplification of them via translators is a process that cannot be fully automated and thus discarded as an option. However, it may be that something similar has to be integrated in the translating process to improve the usability of such models.

5. Looking into the Future

There are strong indications that the market is moving towards customised VR Internet aware applications, and in many cases Internet distributed ones are (although most projects are too big and too inflexible for distribution over the current Internet setup).

VR companies that survived the initial shock of VRML expansion will embrace the new ideas that VRML brought in the VR market and will focus mainly into research applications rather than recreational ones - VRML with the backup of SGI, Sony and Microsoft will most likely dominate the latter. Tools will be developed that will progressively let designers move from the VRML “machine code” to wrapped up user friendly applications without restricting functionality/ flexibility and editability of the content created. Many lessons have to be learnt from the problems that PostScript is still facing - a decade after its introduction there are still no decent tools to “edit” PostScript. A move to hyperspaces greater multi-dimensional spaces will enable navigation based on users' ideas and customised controls and not interfaces created by others—input devices will play an important role.

Computer based urban modelling is still too much in theory and discussion and as such of no real interest to the programmers and software developers. However, as the ideas behind the London's West End model demonstrate, it could become the backbone of a series of interface / database links to information that could potentially be extremely important for businesses as well as the public. These could indeed lead to the vision of replacing the current World Wide Web (Pesce, 1995) with a 3D+ version of hyperspace. It is unfortunate that development in the area is slow, because of the lack of tools as well as understanding and belief into its potential.

References

Bourdakis, V. (1996) *From CAAD to VRML: London Case Study*, The 3rd UK VRSIG Conference. Full Paper Proceedings, De Montfort University.

Bourdakis, V. and Day, A. (1997) *The VRML Model of the City of Bath*, Proceedings of the Sixth International EuropIA Conference, europia Productions.

Liggett, R., Friedman, S., and Jepson, W. (1995) *Interactive Design/Decision Making in a Virtual Urban World: Visual Simulation and GIS*. Proceedings of the Fifteenth Annual ESRI User Conference. Palm Springs, CA.

Pesce, M (1995) *Surfing the Satellites: A New Image of Earth* (paper delivered at the Australian Information Superhighway Conference at Darling Harbour Sydney 16-18 Nov 1995 <URL http://www.hyperreal.com/~mpesce/surfsat.html>

World Wide Web Resources

ART+COM	http://www.artcom.de/contacts/city-and-architecture/
Bath Model	http://www.bath.ac.uk/Centres/CASA/bath/bath_low.html
CADCentre	http://www.cadcentre.co.uk/
CLR, Toronto University	http://www.clr.toronto.edu/
London Model	http://www.bath.ac.uk/Centres/CASA/london/
The Martin Centre	http://www.arct.cam.ac.uk/mc/
UCLA Virtual World Data Server	http://nugget.cs.ucla.edu/vwds/
Virtual Soma	http://www.planet9.com/vrsoma.htm
VRML Specification	http://vag.vrml.org/Specifications/VRML97/DIS/