Streaming Geospatial Data into Virtual Reality

A. Herzig*[1]

[1]Manaaki Whenua – Landcare Research, Private Bag 11052, Palmerston North 4442, New Zealand
*Email: herziga@landcareresearch.co.nz

Abstract

The integration of modelling and simulation with Virtual Reality (VR) contributes to the development of Virtual Geographic Environments. In this paper we present a free and open source prototype software that enables the streaming of environmental modelling results into a web-based VR application. It could potentially be used by GIScience professionals and VR developers for building web-based VR applications intended to help scientists, professionals, and members of the public better understand spatial phenomena and processes relevant for spatial planning processes. The server component of our software is implemented in the Land Use Management Support System (LUMASS). It uses WebSockets to send binary data over a TCP connection to a client application. The prototype client application component is based on A-Frame and the JavaScript WebSockets application programming interface. It visualises the received model results as raster data in a web-based VR application. We successfully tested our application using a Google Daydream headset for mobile phones. Future research needs to investigate the scalability of the presented approach and the adoption of standards for streaming geospatial data.

Keywords: Virtual Reality, LUMASS, A-Frame, Virtual Geographic Environment, Environmental Modelling, Streaming, Geospatial Data

1. Introduction

The visualisation of geospatial modelling results with graphs and maps and more recently with interactive mobile applications is an important element in communicating science to professionals in policy development and spatial planning, but also to members of the public participating in these processes. Virtual Reality (VR) has become an increasingly popular tool in the visualisation toolbox and promises a much more immersive user experience with the hope to facilitate the understanding of spatial phenomena and processes. The integration of VR with spatial analysis and simulation capabilities contributes to the development of Virtual Geographic Environments (VGE) (Voinov et al., 2017; Chen and Lin, 2018). They are used, for example, to conduct experiments on user perceptions on a changing environment (Voinov et al., 2017) or to support city planning (Wang et al., 2018). However, despite the impressive achievements in the development of VGEs, these systems are not easily accessible and the software powering these systems is not always freely and readily available. In this paper, we introduce free and open source prototype software components targeted at GIScience professionals and VR developers for streaming environmental modelling results into web-based VR (WebVR) applications. Since the client application component is browser-based, it can be used with a range of VR hardware, including relatively cheap mobile VR headsets as well as more sophisticated and expensive PC-based devices. The basic software platform of our system, providing environmental modelling and streaming capabilities, is the free and open source Land Use Management Support System (LUMASS) modelling and optimisation framework (Herzig and Rutledge, 2013; Herzig et al., 2013, Spiekermann et al., 2019). In the remainder of the article we provide a short overview of LUMASS (Section 2), the architecture of the LUMASS/VR integration (Section 3), preliminary results, and (future) challenges (Section 4). We conclude the paper with Section 5.

2. The Land Use Management Support System (LUMASS)

LUMASS (Herzig, 2018) is free and open source software designed for geospatial system dynamics modelling (Herzig and Rutledge, 2013), geospatial processing workflows, and spatial optimisation (Herzig et al., 2013, 2016, 2018). It is designed for GIS professionals and scientific modellers and provides an icon-based visual modelling environment to facilitate model development (Fig. 1).

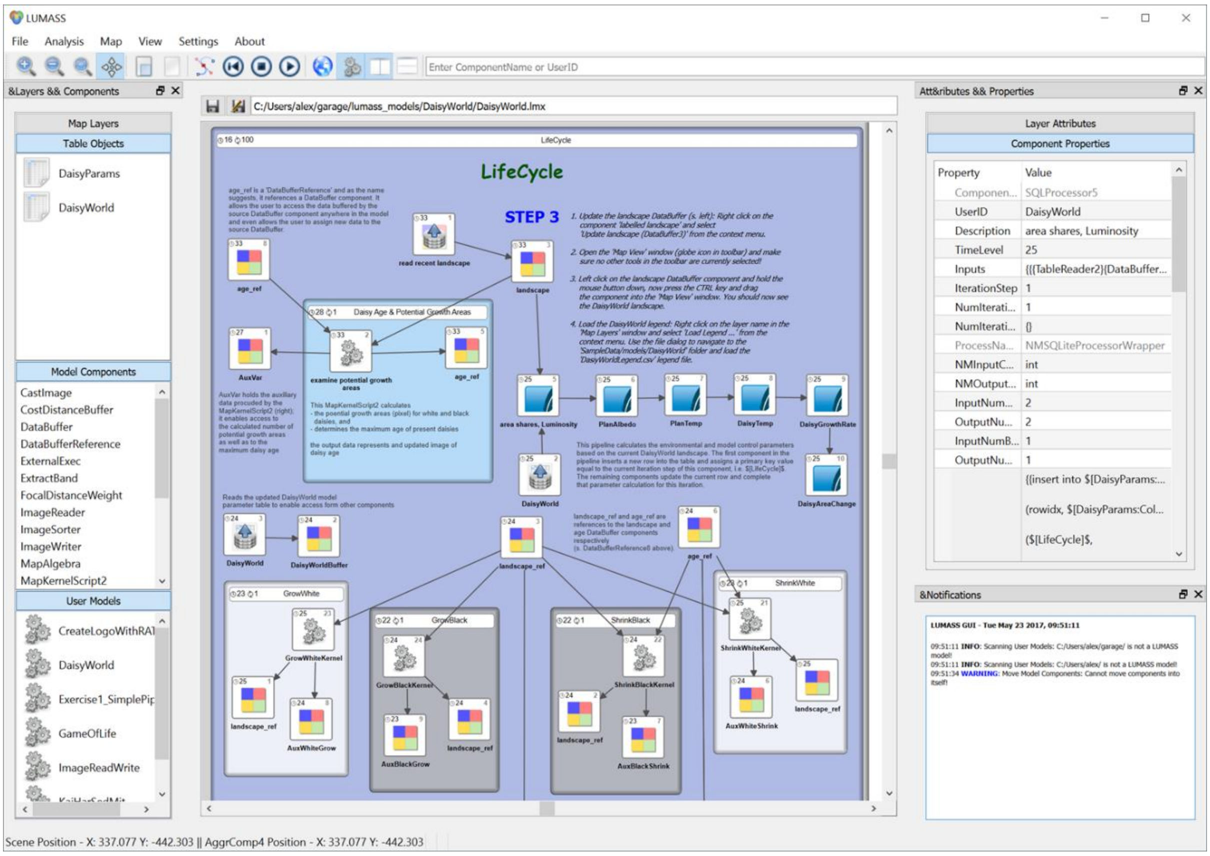

Figure 1: The LUMASS visual modelling environment.

Recognising the importance of remotely sensed imagery (e.g. satellite or drone imagery) for environmental modelling, LUMASS focuses explicitly on raster-based modelling. To be able to process large and multi-dimensional raster data sets sequentially and in parallel, the LUMASS modelling framework utilises a range of open source software libraries, e.g. Insight Toolkit (ITK, Johnson et al., 2018), Orfeo Toolbox (OTB, 2018), the Visualisation Toolkit (VTK, Schroeder et al., 2006), rasdaman (Baumann et al., 2018), and the Qt C++ programming framework (Qt, 2019). LUMASS models are based on processing pipelines that can be grouped and nested into hierarchical processing workflows operating on multiple temporal scales (Fig. 1). Each pipeline is comprised of individual model components providing atomic processing capabilities, such as map algebra, cellular automata, unique combinations, cost-distance analysis, zonal statistics, SQL processing, etc. Additionally, LUMASS enables the integration of any other third-party command line-based processing component (e.g. python, R). Models can be exported into XML-based files and run with the lumassengine application independently of the graphical user interface in server and cluster environments. Fine-grained model provenance information is automatically tracked and stored in the working directory as PROV-N (W3C, 2013) file (Spiekermann et al., 2019). Model results can be visualised as graphs, tables, and maps in the desktop application.

3. Live-Streaming Modelling Results to VR

To stream modelling results from LUMASS into a client (WebVR) application, we developed a specific model component that plugs into a processing pipeline (Fig. 1) and sends any incoming data over the network to a connected client. The communication is based on the Qt WebSockets application programming interface (API) and sends raster data as binary stream of pixel values over a direct TCP connection to the client. To enable interpretation of the streamed data at the client side, we add a small header to the stream carrying information on the image dimensions in x, y, and z direction. In comparison with HTTP requests, WebSockets avoid message overheads and provide a smaller payload, which overall allows for a faster communication between client and server. Our prototype client application component is implemented using A-Frame (Marcos et al., 2019) and the JavaScript WebSockets API (WHATWG, 2019). A-Frame is a JavaScript-based framework to facilitate the development of 3D and WebVR applications in HTML, without requiring any knowledge of WebGL (Khronos, 2019). Figure 2 illustrates the workflow and components of a LUMASS-based client server application potentially involving streaming of model results to the client. Environmental models are implemented in LUMASS' visual modelling environment and tested on local data. The models can then be exported to XML-based files and sent to a server or cluster environment that provides the required data for the execution of the model. These data may be provided by the user or sourced from external providers. The lumassengine executes the model and stores the data on the server. If the model incorporates the dedicated streaming component and runs in an appropriate server environment, selected modelling results can also be streamed via the WebSocket-based TCP connection to remote client components built into a WebVR application.

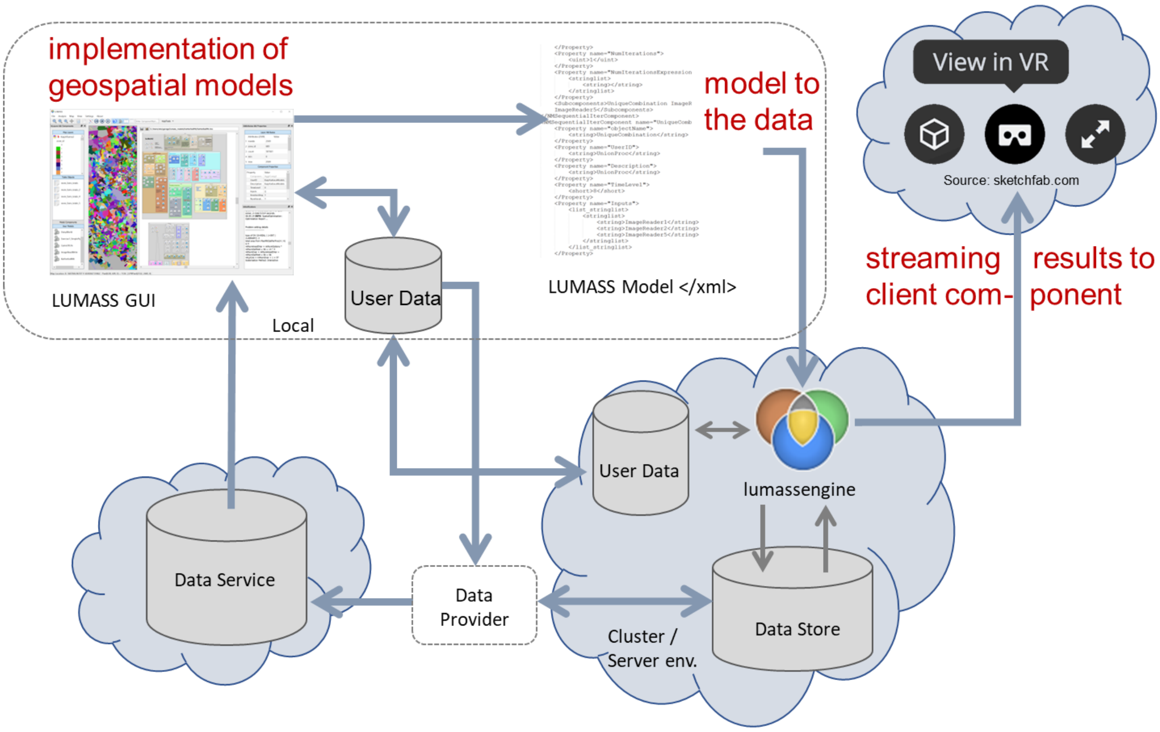

Figure 2: Components and data flow for a LUMASS-based client-server application.

4. Results and Discussion

We have successfully implemented and tested a prototype of a LUMASS model component for streaming modelling data to a client using a WebSocket-based TCP connection. Furthermore, we have implemented a prototype A-Frame-based client application component that renders the received geospatial raster data in a virtual environment. Figure 3 shows a test configuration where the cellular automata model Conway's Life (Gardner, 1970) is run over a small test area (512 x 512 pixel) and visualised in the LUMASS desktop application (Fig. 3, left). The results (Fig. 3, bottom) are streamed at runtime from the modelling component (Fig. 3, bottom left) to the WebVR component (Fig. 3, bottom right) and rendered as texture over a digital elevation model visualised in Figure 3 (top). We have also tested the application on a mobile phone using a Google Daydream VR headset. The implementation is currently limited to an A-Frame component rendering the received modelling data as texture of a 3D model but can be combined with other A-Frame components, e.g. interactive controls and other 3D models, to build a more sophisticated VR application. Conceptually, the presented approach can also be used to stream modelling results to any web-based client that knows how to interpret the received data stream. The implementation of data streaming as a model component enables its integration at different locations in the processing pipeline. This allows users to monitor or visualise specific outputs or intermediary data at different points in the processing workflow. However, at this stage our application only allows for one streaming component in a model and the support of multiple streams at different points in the workflow is subject to future development. More generally, future research needs to investigate the performance of the presented approach in terms of its scalability regarding the number of clients that the server component can support and regarding the incoming data volume the client component can process. Additionally, it needs to investigate the adoption of streaming standards, e.g. the 3D tile specification

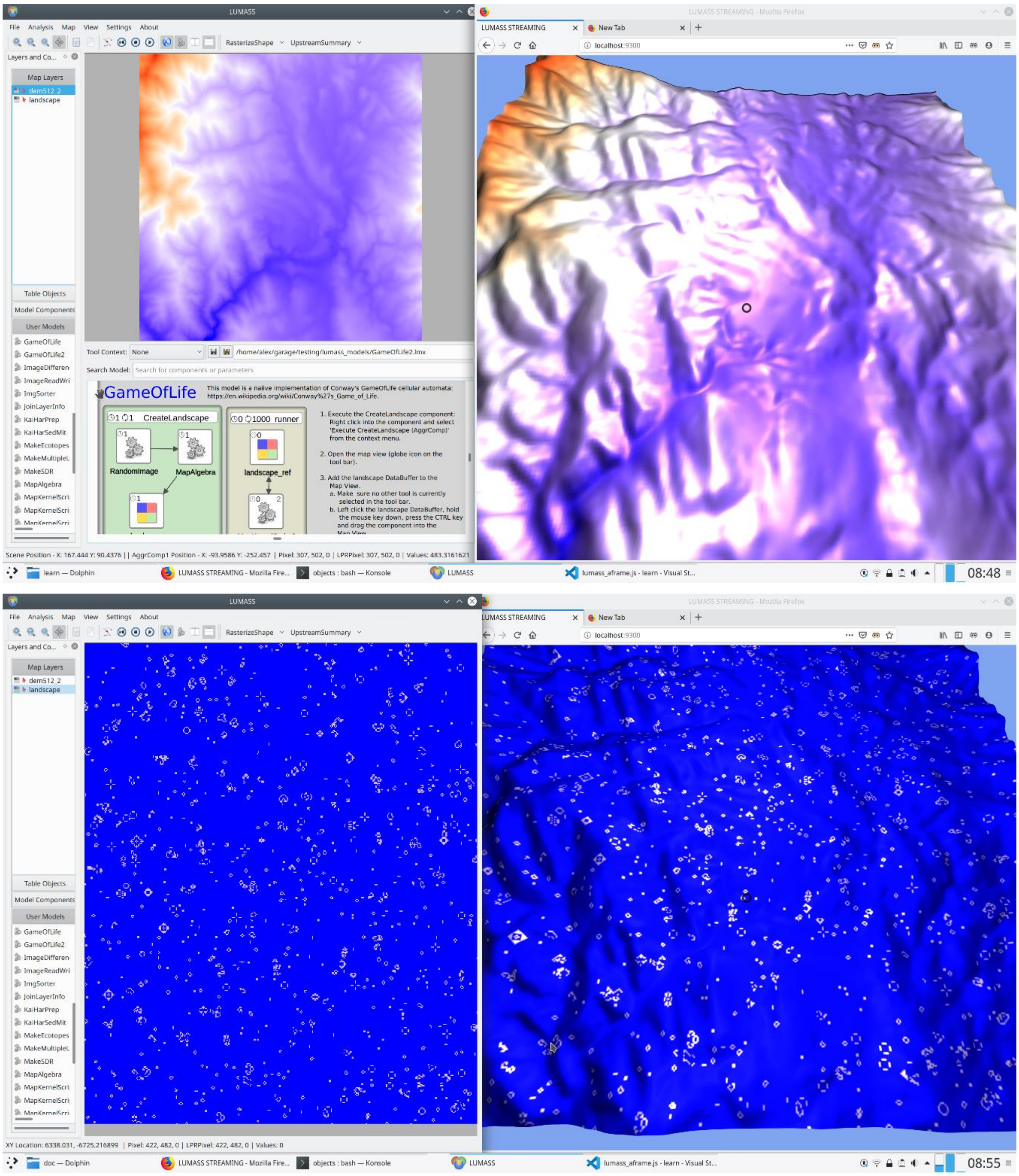

Figure 3. Streaming (left) and client (right) component showing a digital elevation model (top) of the test area and a frame, i.e. time step, of Conway's Game of Life model (Gardner, 1970) (top left, bottom half) run over the test area (bottom).

(OGC, 2019) for streaming massive geospatial content, in order to facilitate the development of clients that are independent of server implementations.

Besides the technical aspects of streaming raster data to VR, future research needs to clarify the role it could play in these environments. Traditionally, it is used as texture to provide objects and surfaces with a photo-realistic look. Unfortunately, abstract surfaces generated by environmental models do not necessarily contribute to the photo-realistic look in a VGE (cf. Fig. 3). However, since raster-based model data are very abstract und difficult to understand for laypersons, visualising them in their ‘natural’ geospatial context may help with interpretation of such data. In the case of spatiotemporal data cubes, VR or 3D environments that are equipped with appropriate controls or widgets are certainly helpful in exploring these data much better than it would be in 2D. In addition to visualise raster in VR, modelled raster surfaces, for example provided by a forest-growth model, could be used to mark locations for more realistic 3D content that is rendered according to the modelled height and diameter by the model. This way, specialised software for modelling and rendering could interoperate well while each would still benefit from its highly specialised and efficient software environment.

5. Conclusions

We have successfully implemented a prototype client-server application for the integration of modelling and simulation with VR using geospatial data streaming. Our preliminary results are encouraging and may prove useful to facilitate the development of Virtual Geographic Environments. More research is required to investigate the scalability of the concept as well as the adoption of streaming standards to enable the implementation of streaming clients that are independent of server implementations.

6. Acknowledgements

We thank the New Zealand Ministry of Business, Innovation and Employment’s Science and Innovation Group for supporting the presented work.

7. References

Baumann, P., Misev, D., Merticariu, V. and Pham Huu, B. 2018. *Datacubes: Towards Space/Time Analysis-Ready Data.* In: Doellner J, Jobst M, Schmitz P (eds.), Service Oriented Mapping - Changing Paradigm in Map Production and Geoinformation Management, Springer Lecture Notes in Geoinformation and Cartography.

Chen, M. and Hui, L. 2018. *Virtual geographic environments (VGEs): originating from or beyond virtual reality (VR)?* International Journal of Digital Earth 11: 329-333.

Gardner, M. 1970. *Mathematical Games – The fantastic combinations of John Conway's new solitaire game "life"*. Scientific American. 223 (4): 120–123.

Herzig, A. 2018. *The Land Use Management Support System*. [Online]. [2019-06-14]. Available from: https://bitbucket.org/landcareresearch/lumass/wiki/Home.

Herzig, A., Rutledge, D. 2013. *Integrated Land Systems Modelling and Optimisation*. In: Piantadosi, J., Anderssen, R.S. and Boland J. (eds) MODSIM2013, 20th International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand, December 2013, pp. 880–886.

Herzig, A., Ausseil, A.-G.E., Dymond, J.R. 2013. *Spatial Optimisation of Ecosystem Services*. In Dymond JR (ed.), Ecosystem Services in New Zealand - conditions and trends. Manaaki Whenua Press, Lincoln, New Zealand.

Herzig, A., Dymond, J.R. and Ausseil, A.-G.E. 2016. *Exploring Limits and Trade-Offs of Irrigation and Agricultural Intensification in the Ruamahanga Catchment, New Zealand*. New Zealand Journal of Agricultural Research 59:216-234.

Herzig, A., Nguyen, T.T., Ausseil, A.-G.E., Maharjan, G.R., Dymond, J.R., Arnhold, S., Koellner, T., Rutledge, D. and Tenhunen, J. 2018. *Assessing resource-use efficiency of land use*. Environmental Modelling & Software 107:34-49.

Johnson, H.J., McCormick, M.M., Ibanez, L. and Insight Software Consortium. 2018. *The ITK Software Guide Book 1: Introduction and Development Guidelines, Fourth Edition Updated for ITK version 5.0.0*. [Online]. [2019-06-14]. Available from: https://itk.org/ItkSoftwareGuide.pdf

Khronos (Khronos Group). 2019. *WebGL: OpenGL ES for the Web*. [Online]. [2019-06-14]. Available from: https://www.khronos.org/webgl/

Marcos, D., Ngo, K. and McCurdy, D. 2019. A-Frame: A web framework for building virtual reality experiences. [Online]. [2019-06-14]. Available from: https://aframe.io/

Open Geospatial Consortium (OGC). 2019. *3D Tiles Specification 1.0*. [Online]. [2019-06-14]. Available from: http://docs.opengeospatial.org/cs/18-053r2/18-053r2.html

OTB (OTB Development Team). 2018. *The ORFEO Tool Box Software Guide – Updated for OTB-6.6.1*.[Online]. [2019-06-14]. Available from: https://www.orfeo-toolbox.org/packages/OTBSoftwareGuide.pdf

Qt (The Qt Company). 2019. *Qt Documentation*. [Online]. [2019-06-14]. Available from: https://doc.qt.io/

Schroeder, W., Martin, K. and Lorensen, B. 2006. *The Visualization Toolkit (4th ed.)*, Kitware, ISBN 978-1-930934-19-1.

Spiekermann, R., Jolly, B., Herzig, A., Burleigh, T. and Medyckyj-Scott, D. 2019. *Implementations of fine-grained automated data provenance to support transparent environmental modelling*. Environmental Modelling & Software 118:134-145.

Voinov, A., Coltekin, A., Min, C. and Beydoun, G. 2018. *Virtual geographic environments in socio-environmental modeling: a fancy distraction or a key to communication?* International Journal of Digital Earth 11:408-419.

W3C. 2013. *PROV-N: the provenance notation*.[Online]. [2019-03-07]. Available from: http://www.w3.org/TR/2013/RECprov-n-20130430/

Wang, W., Lv, Z., Li, X., Weiping, X., Baoyun, Z., Zhu, Y. and Yan, Y. 2018. *Spatial query based virtual reality GIS analysis platform.* Neurocomputing 274: 88-98.

WHATWG. 2019. *HTML Living Standard, 9.3 Web sockets*. [Online]. [2019-06-14]. Available from: https://html.spec.whatwg.org/multipage/web-sockets.html#network