

3DAVIS

3-Dimensional Astronomical Visualisation

The Problem

The new generation of radio astronomy telescopesare capturing an increasingly amount of astronomical data and therefore, requires improved software designs and visualization techniques.

Consequently, we present the development of 3DAVis, a system that renders astronomical data into a 3-dimensional visualization, using a client-server hybrid rendering approach.

Its goal serves as a proof of concept for future 3D astronomical data visualization tools, where the computationally intensive tasks are handled by the server and the data exploration as well as interaction are handled by the client.

We explore various techniques and algorithms which would potentially be included in such future applications and discuss their associated benefits, drawbacks, bottlenecks, and opportunities.

The Objective

The aim of our research project was to construct a proof of concept system that implements a strategy for allowing any average computer to produce visualisations of very large data cubes in a lag-free manner.

Consequently, we also intended to answer a series of research questions which would present our investigations in a more formal and statistical format. These research questions are related to determining the usability of our new system approach, the scalability of using increasingly large data sets as well as the effectiveness of using specific libraries such as VTK for volume rendering and ZFP for 3D compression.

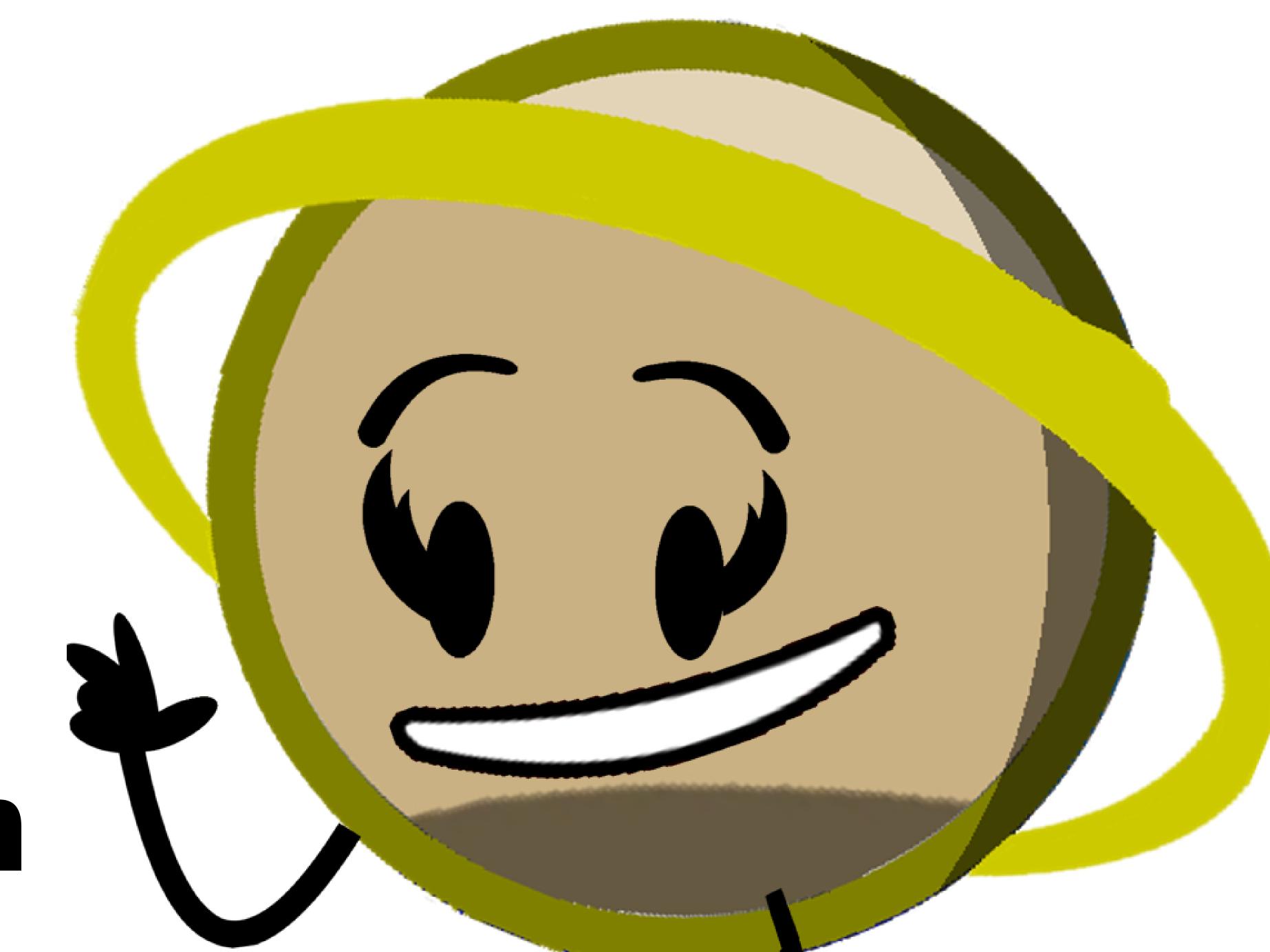
Design

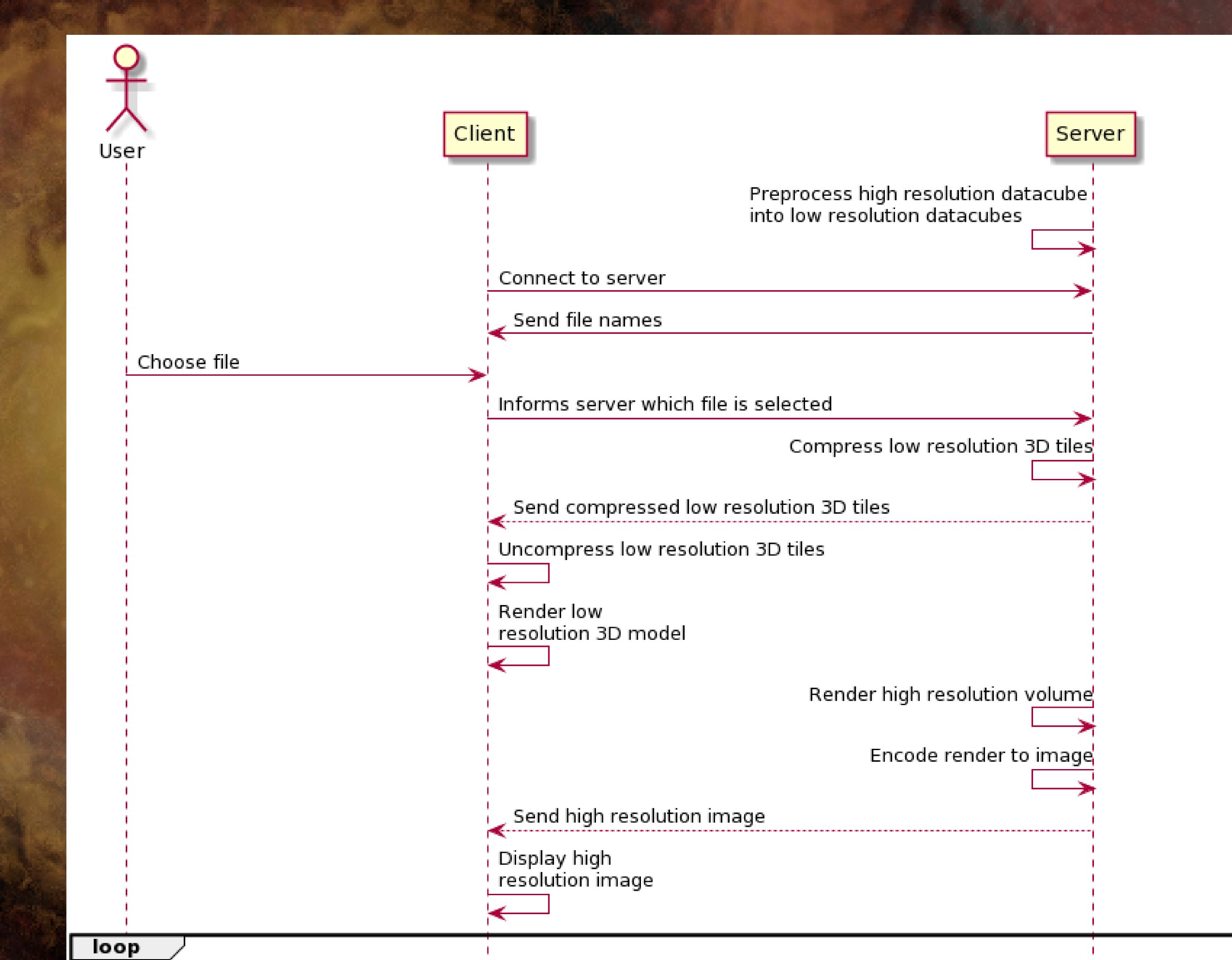
The strategy for addressing the research problem is to implement a system using a server-client model. It would combine server and client side rendering but the majority of the computational overhead would be offloaded onto the remote server.

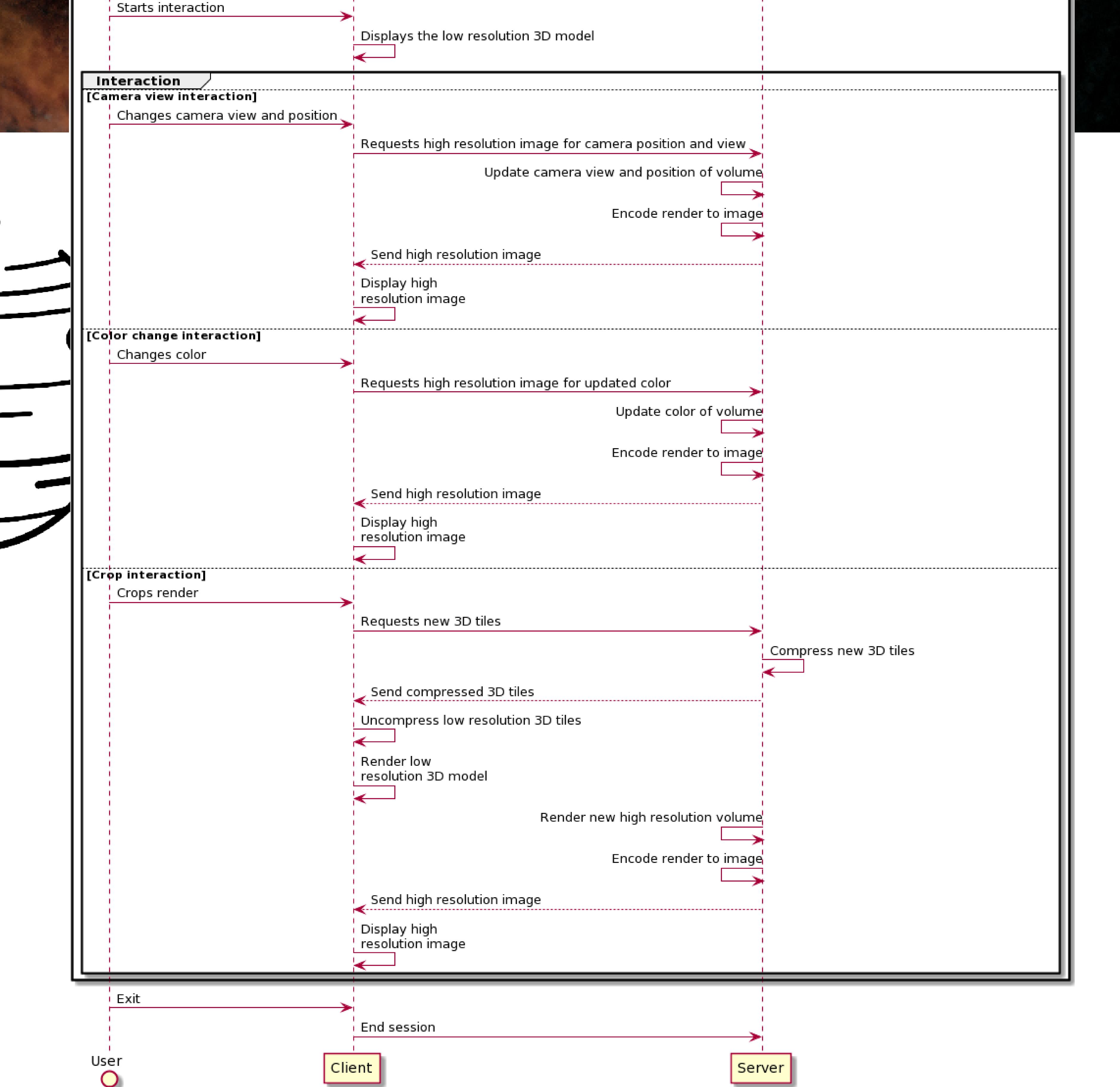
The data cubes would be stored on the remote server and users can access them through a client computer connect to the internet.

The server sends volume and image data to the client on request. The volume data is three-dimensional data cubes depiction a portion of the larger data cube, these smaller cubes have been downsampled by a certain factor, and the image data is a high-resolution rendering of the data cube, these images are generated by the server which would have enough computational resources to produce the visualisation.

When there is no interaction on the client it displays the high resolution rendering and when there is interaction the client switches to rendering the volume data it received, this makes interaction with the data instantaneous and the user does not have to wait for the server to produce more frames and send them to the client.







Results and Conclusion

