

# DomeView: A Tool for Digital Planetariums

M. A. Silva<sup>1</sup>, A. R. Fernandes<sup>2</sup> and A. Pedrosa<sup>1</sup>

<sup>1</sup>Navegar Foundation, Espinho, Portugal

<sup>2</sup>CCTC, Universidade do Minho, Portugal

---

## Abstract

*The content produced for fulldome projection is usually made in a frame (the master) with a format corresponding to a projection of the dome on a plane, commonly the fisheye projection. Due to the distortions involved, looking at the master does not give a clear idea of the final result once projected on a planetarium dome. The common solution when creating and composing content is to go through an iterative process, making successive projections on a planetarium until the desired result is achieved. However, repeated access to a planetarium is difficult, costly, and time consuming. So there is a need for an application that can provide the user with a clear notion of the final result in the planetarium. In addition, if the application could be used to show content in a small (single projector) planetarium, like a portable one, its versatility would increase substantially. This paper reports on such an application: Domeview.*

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.8]: Applications—I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality; I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms

---

## 1. Introduction

The new millennium has brought a revolution to the planetarium world. The digital media are here to stay, and systems that can project digital footage, filling an entire dome, have become available [McC05]. A revolution always implies new challenges, and one of the main challenges is content production.

Fulldome contents include real time sky simulators, mainly for the night sky, to display the celestial sphere and phenomena associated to celestial mechanics. Nowadays there is even freeware that is able to provide such content, such as Stellarium [Ché01] or Celestia [Lau01].

Besides fulldome content specific to a planetarium, the digital era allows for an easier composition of multiple layers of content of both pre-rendered and real-time material, such as movies, live video streams, images, 3D content, amongst others.

Many tools are available for each of these media, however they all produce an output suitable for a 2D flat screen. Even tools that produce appropriate output for a planetarium only allow previewing in a 2D flat canvas.

Domeview is an application that allows the congregation of multiple media from third party applications and is designed to allow for an easy composition of all these inputs. The preview feature of Domeview allows for a precise composition of all these elements in a virtual dome. Previewing inside the DomeView application also allows for an accurate notion of the end result on a real planetarium. Peripheral vision experience still requires a real planetarium though.

Domeview can prepare masters combining all these media in an intuitive and effortless way. These masters could easily be saved for later use in real planetarium systems.

Due to the spherical nature of the surface where the media will be presented, the media must undergo a transformation to be ready for projection, the standard being the azimuthal equidistant projection, also known as fisheye projection [Sny87]. Currently Domeview supports fisheye and spherical mirror projections [Bou03].

Naturally, Domeview can also be used to feed a projector with a fisheye lens, or a standard projector coupled with a spherical mirror system, with the masters being created in

real time, making it a full-fledged tool for small planetariums.

## 2. Previewing

Previewing content, prepared for a dome, in a 2D canvas is a complex task due to the distortions involved in the projections required for a suitable viewing in a planetarium. The digital media that can be presented in a planetarium are usually created in applications that provide a rectangular canvas to visualize the result, such as video editing tools, After Effects from adobe, and 3D modeling tools, to name a few.

These tools were originally created for the 2D screen, but its application to planetariums has the potential to greatly enhance the visitor's experience. Domeview provides the bridge between these tools and the planetariums world.

Some of this content will maintain its rectangular aspect ratio when projected on a dome, for instance, it is a common scenario to include for instance a night sky in the background and other 2D elements on top. However, content developers still need to have a clear idea of what the final composition will look like in a dome environment as soon as possible. As mentioned before, access to a real planetarium may become costly and time consuming.

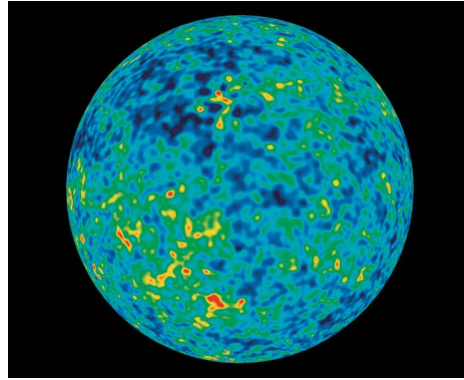
Therefore, previewing is required not only for each media element in its own, but also for the final composition. Content creators need to be able to compose all the elements in an environment that is as close as possible to the real thing, otherwise the master composition can become too time consuming.

In Domeview all this content is projected in a spherical grid, simulating the planetarium's dome. The dome's inclination is configurable so that the preview can match many modern planetariums, see figure 1. The entire dome can be displayed up to  $360^\circ$ , as in the all-sky view of the WMAP4, see figure 2.



**Figure 1:** Tilted planetarium.

The 3D model of the dome can be complemented with a model of the planetarium itself, providing a more accurate preview, figure 3. Adding a model of the planetarium helps



**Figure 2:** Sphere.

to increase the sense of scale, based on the dimension of other objects such as the chairs, and the immersive sensation. The user can navigate in real time in this virtual scenario, the viewing camera can be positioned freely anywhere, inside or outside the virtual planetarium. In this way, the Domeview user can get a perspective of what the visitors of the planetarium will see in different locations, figure 4, or get a global overview of the final result, figures 3 and 1.

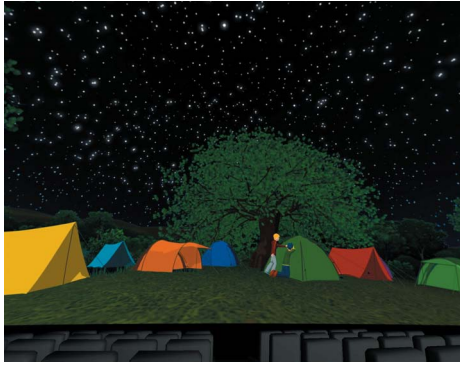


**Figure 3:** Looking through the dome.

## 3. Feeding Content to Domeview

Software such as Stellarium and Celestia, are two of the most relevant packages for small and portable planetariums. Stellarium simulates the night sky, whereas Celestia allows traveling through the Universe, so in a sense they complement each other.

Digital media such as video streams, images, and 3D content can be used to enrich the visual experience for planetarium visitors. Besides importing these types of media, Domeview also supports input from a webcam, allowing the addition of live video streams to the final composition of the master.



**Figure 4:** *Inside a virtual planetarium.*

In addition to importing common video and image formats, Domeview is designed to support communication with third party software via plugins, or source code intervention. The plugin architecture allows communication with third party software, and in most cases updates occur in real time. For instance, when modifying a file in Adobe After Effects, the modifications can be seen in real time inside Domeview. This feature further enhances previewing in Domeview. Not only can Domeview import the final result created on third party software, it can also visualize the content in real time while it is being created.

Tools such as Adobe After Effects are very powerful and require a substantial amount of time to be mastered. The main weak point though is previewing in a dome environment, and this is where Domeview comes in. This approach allows content developers to continue working in their favourite tools, while having improved previewing capabilities, hence shortening the production cycle.

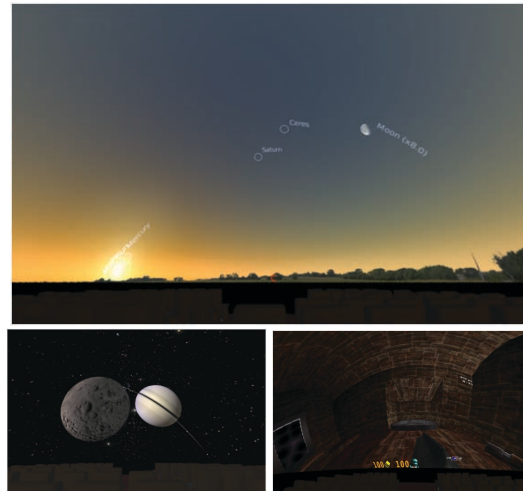
Major software packages allow developers to write plugins to enhance their functionality. Plugins have been developed for some tools, such as Adobe After Effects and Photoshop, to send the content to Domeview. In some cases, such as After Effects, the modifications are seen in real time. Other packages don't allow this functionality. Photoshop is such an example. In these cases the plugin exports the content on demand only.

The approach to implement the communication is based on a client/server scheme where the plugins act as image servers, and Domeview is the client. Each communication channel, between a server and Domeview provides an image stream that feeds Domeview. The system is implemented using shared memory.

For video content, Domeview can work as a standalone tool since it has the ability to read videos, or through plugins.

Open source software such as Stellarium provides access to the source code. In this case it is possible to modify the source so that the frame buffer is sent to Domeview. This has

been done for Stellarium, Celestia and even Quake 3 Arena, see figure 5.



**Figure 5:** *Stellarium (top), Celestia (bottom left) and Quake (bottom right) displaying content inside Domeview.*

Some previously created content has already been transformed according to a specific projection. To enhance the ability of Domeview to import and deal correctly with these cases, the tool is able to process content created with 14 different types of projection, namely: Azimuthal Equidistant, Azimuthal Equidistant Off-Axis, Aitoff, Hammer Aitoff, Gnomonic, Stereographic, Orthographic, Vertical Perspective, Cylindrical Equidistant, Cylindrical Equal Area, Center Cylindrical, Panoramic Equidistant, Panoramic Equal Area, and Center Panoramic.

### 3.1. Filling the Dome

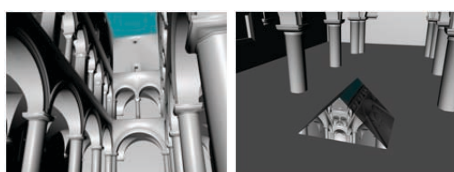
One can classify the content being displayed in a planetarium regarding the area it uses on the dome. Some content is designed to fill the entire dome. Sky simulators or software that allows us to *travel* in the Universe are good examples of content designed to fill the entire dome. Other content uses only a small portion of the dome. These can be used as a support for the main content, the background filling the dome.

If a content is designed to fill the entire dome, then it should ideally be already in fisheye format. In Domeview fisheye content is the easiest way of filling the entire dome. Some applications already produce content in fisheye format. Two such applications, already able to communicate with Domeview, are Stellarium and After Effects, with the fulldome plugin.

However, most applications do not produce fisheye content. Yet it would be great if their content could be used to fill an entire dome. Most 3D applications fall in this category, for instance Celestia and Quake 3 Arena only produce

content suitable for a 2D flat canvas. In this case, having the dome filled with the content requires some extra work.

The approach in Domeview, for OpenGL based applications, is a three step process based on the standard 3D graphics cubemapping technique. In the first stage, four virtual cameras are placed inside a virtual rotated hemicube (only four faces are required) facing each of the relevant faces. The 3D world is then rendered to the four relevant faces of the hemicube, see figure 6.



**Figure 6:** left: *Sponza Palace (model by Marko Dabrovi)*; right: *the rotated hemicube*.

In the second step, an hemisphere is rendered with environment cube mapping technique, where the textures used in the cube are the previously rendered faces, see figure 7.



**Figure 7:** *Inside the hemicube, with the hemisphere at the center*.

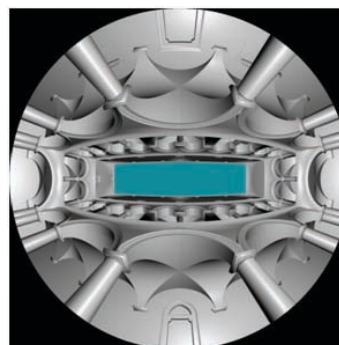
Finally a camera is placed beneath the hemisphere and a new render is computed. This final render corresponds to the fisheye of the scene, see figure 8.

Using this approach any 3D application could be easily modified to provide fisheye content.

### 3.2. Composing the Master

Placing the media from third party software in real time further enhances Domeview's capabilities. In Domeview every individual content has a set of parameters that define its projection and its position on the dome. Figure 9 shows a Domeview dialog to set the parameters for content placement.

Each individual content is placed based on an azimuth



**Figure 8:** *The final fisheye result*.

and altitude in the hemisphere that represents the planetarium's projection area. The altitude is the angular distance of a heavenly body above the horizon, whereas the azimuth is the arc of the horizon measured clockwise from the north point [KKO\*00]. The dimension of the content in the dome, as well as a rotation can also be defined. Based on these parameters, the appropriate texture coordinates on the hemisphere are computed.

One element always present when dealing with digital content is aliasing caused by resampling when projecting and resizing the original content. Aliasing artifacts in a planetarium are very noticeable due to the large projection area.

Several filtering methods have been implemented to deal with this issue, namely bilinear, trilinear, bicubic and bicubic with splines. The filtering methods introduce a small penalty, performance wise, but the quality improvement makes it worth while.

As an example of the composition abilities of Domeview consider the following scenario. The final display should have the night sky, generated with Stellarium. Four characters are placed inside a forest looking at the sky. One of the characters is a tutor, and the other three are children. The forest was created in an image editing package, whereas the characters were created using animation software. Figure 10 shows the different elements used, and figure 11 shows the final master composed in Domeview.

## 4. Projecting

Domeview must support at least a fisheye output to be able to create masters for a planetarium. The output from Domeview is the OpenGL window where content is being presented, however OpenGL does not have a fisheye camera. The field of view can be increased up to 180 degrees but the distortions it produces are not suitable for a fisheye projection.

The workaround for this problem is to create a grid mesh that once the content is rendered in such a mesh it suffers the same distortions as the fisheye projection would cause.



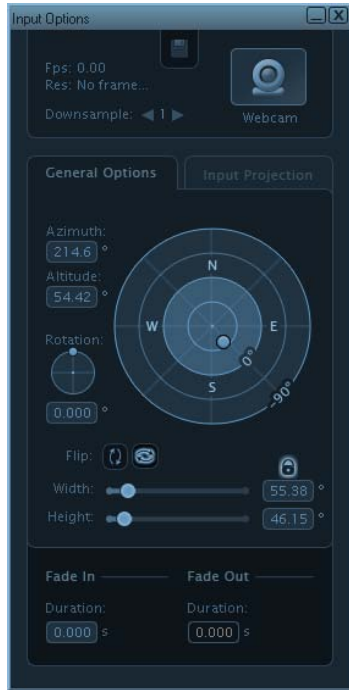


Figure 9: Plugin options.



Figure 10: Elements used to create the master.

Such a mesh is an hemisphere, possibly rotated according to the dome's parameters. The OpenGL camera is then placed beneath the mesh with a field of view of 90 degrees, see figure 12.

The end result is equivalent to a fisheye transformation.

Domeview also supports an output ready to be projected, using a regular projector, for a spherical mirror setup [Alp]. The mesh is created using the spherical mirror parameterizations. The spherical mirror setup, and the OpenGL setup are presented in figure 13

This approach is very flexible and enables the addition

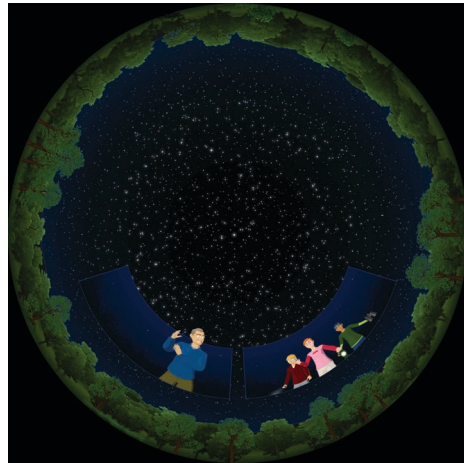


Figure 11: The final composition.

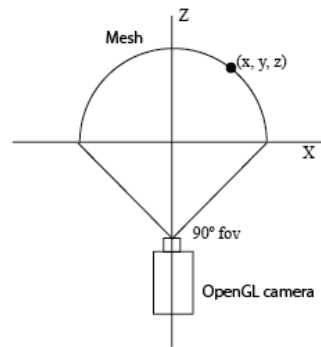


Figure 12: Fisheye Scheme.

of any type of projection, simply by creating an appropriate grid mesh and the respective OpenGL setup.

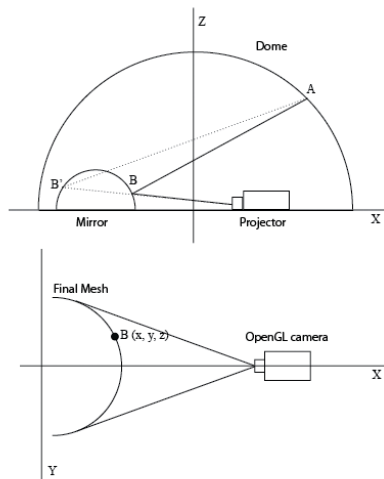
### 5. Performance Tests

The capabilities of Domeview, at least when used for projecting content, depend heavily on its performance. In this section two tests are reported using Domeview with different loads.

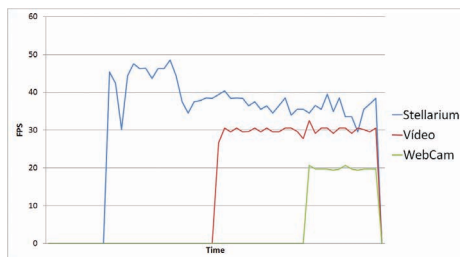
Both tests were performed in a PC with an Intel Core 2 Quad Q9450 CPU, 2GB Ram, and a GeForce 8800GT.

The first test shows the frame rate obtained when using Stellarium simulating the night sky as time passes, a video (400x400), and the input of a webcam simultaneously, see figure 14.

From figure 14 it can be seen that Domeview is able to de-



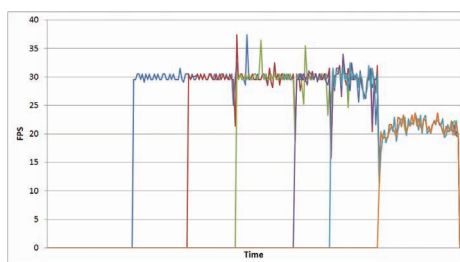
**Figure 13:** Spherical Mirror Scheme.



**Figure 14:** Performance Test with Stellarium, a video and a webcam.

liver the content without any problem. Note that all plugins are running at their maximum fps.

The second test was performed with 6 videos 800x800 running simultaneously. The videos were introduced sequentially, see figure 15.



**Figure 15:** Performance Test with 6 videos.

As shown in figure 15 Domeview manages to get the video's full frame rate up to the sixth video. When the sixth video starts playing the frame rate suffers a sudden drop to

20 fps. The cause for this deterioration in performance is related to the amount of information that travels from the videos to Domeview. For each communication channel we have 73,24 MB/sec (800x800x4x30, four channels, RGBA, at 30 frames per second). When multiplying this amount of data by six, the number of videos, it becomes clear that the data traffic is taking its toll.

Similar results were obtained when testing Domeview with two high definition videos (1920x1080). In this case the introduction of the second video causes performance to go from 30 fps to 15 fps. For each video the communication channel must support a heavy 237,3 MB/s. Two videos imply over 470 MB/s of data transfer, which seems to be an overload for the tested hardware.

## 6. Conclusions

Digital planetariums are capable of offering an amazing experience to visitors. However, the expectations are also very high. Cinema is probably the big culprit as one is used to extremely high standards.

Traditional content creation tools, i.e. tools that produce content appropriate for a 2D flat canvas, are not adequate for creating full-dome content. The distortions caused by the required projections for a full-dome, the standard being fish-eye, make it extremely hard to preview the final effect of the content in a planetarium.

Domeview acts as a bridge between these traditional tools and the planetarium world. It provides previewing capabilities in a 3D virtual dome. A model of the planetarium itself can be included for added realism, and to allow the content creator to test how the content will look from different places inside the planetarium. The large number of plugins already developed, and the number of projections supported make it a very versatile tool.

By nature, due to the 3D representation of the planetarium's dome, Domeview is also a highly capable composition tool. The media are placed in an intuitive fashion and preview is in real-time.

At last but not least, Domeview can feed directly a projector to display fisheye content, in which case a fisheye lenses is required, or work in a spherical mirror setup with a standard projector.

The performance tests show that Domeview is capable of delivering content at highly interactive frame rates, while delivering a substantial amount of content.

The goal of Domeview is not to compete with solutions like DigiStar [ES10] from Evans and Sutherland or DigitalSky [Sky10] from SkySkan. Domeview is designed for small and portable planetariums and it has proven to be an excellent all round tool, that can compose, preview and even project.

## 6.1. Future Work

Although Domeview is already a very capable tool there is still a lot of work ahead.

The performance tests show that there is a bottleneck in the communication with the plugins. Image compression techniques could be used to alleviate this problem. Further work is required to evaluate the impact on the bit rate and the overall performance.

Another really cool improvement for Domeview would be to allow scripting and include a timeline similar to 2D video editing software. In this way, Domeview could be programmed to display complete planetarium sessions without intervention, freeing the presenter to concentrate on the audience and share with them the knowledge.

Each new plugin has the potential to bring new type of content into Domeview. Plugins for 3D modeling tools and video conference software would be nice additions to the list already provided by Domeview.

## References

- [Alp] ALPERIN R. C.: Mathematical origami: Another view of alhazen's optical problem. In *3rd International Meeting of Origami(3OSME)*, A.K. Peters. 5
- [Bou03] BOURKE P.: *Dome projection using a spherical mirror*. <http://local.wasp.uwa.edu.au/~pbourke/miscellaneous/domemirror>, 2003. 1
- [Ché01] CHÉREAU F.: *Stellarium*. <http://www.stellarium.org>, 2001. 1
- [ES10] EVANS, SUTHERLAND: *DigiStar*. [http://www.es.com/products/digital\\_theatre](http://www.es.com/products/digital_theatre), 2010. 6
- [KKO\*00] KARTTUNEN H., KROGER P., OJA H., POUTANE M., DONNER K.: *Fundamental Astronomy*. New York, Springer - Verlag, 2000. 4
- [Lau01] LAUREL C.: *Celestia*. <http://www.shatters.net/celestia>, 2001. 1
- [McC05] MCCONVILLE D.: *The Discourse of Domes - Converging Approaches to Immersive Cinema*. Proc. Workshop in Immersive Cinema, 2005. 1
- [Sky10] SKYSKAN: *DigitalSky*. <http://www.skyskan.com>, 2010. 6
- [Sny87] SNYDER J. P.: *Map Projections - A Working Manual*. U. S. Geological Survey Professional Paper 1395. Washington, DC, U. S. Government Printing Office, 1987. 1