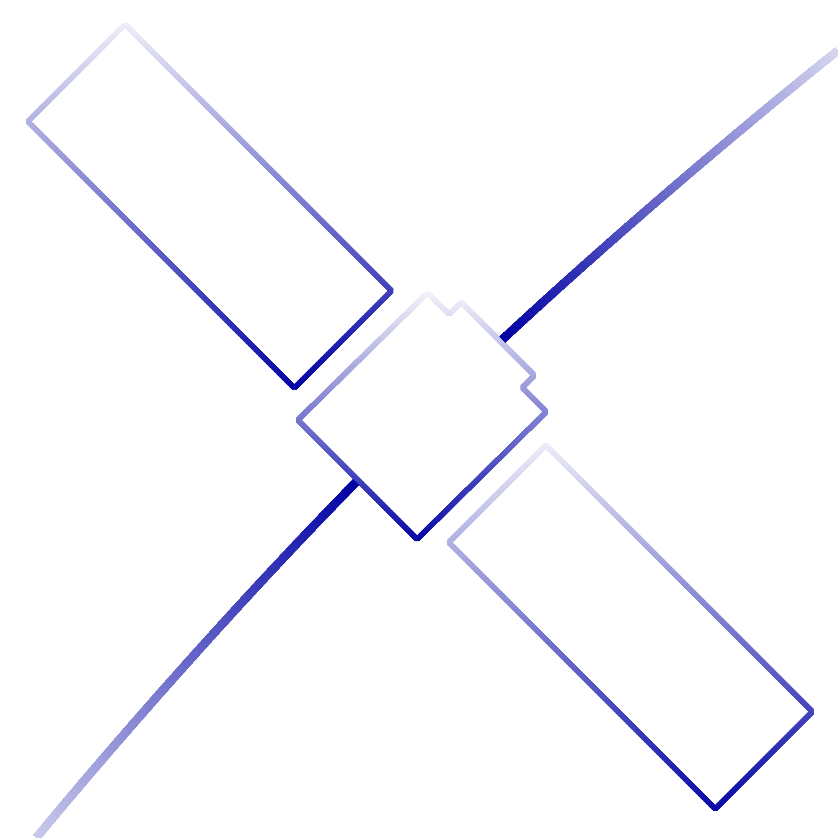


COMPASS

Software Design Document

Embry-Riddle Aeronautical University

Daytona Beach, FL

Introduction

Problem Discussion

Solution Discussion

Requirements

Software Architecture

Reference

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1.0 | The program shall produce renderings of RSO’s. |
| **2.0** | The program shall provide a *pass* mode in which the RSO shall be rendered in orbit and photometry data shall be generated and recorded. |
| **3.0** | The program shall provide a *manual* mode in which the user may define the simulation parameters and see their effects on photometry data. |
| **4.0** | The program shall provide a *playback* mode in which previously recorded simulation data will be used to render the RSO and live graph. |
| **5.0** | The program shall provide a *reconstruction* mode in which the attitude or shape and attitude of the RSO will be generated according to recorded photometry data. |
| **6.0** | The program shall provide render modes which determine the graphical representation of the RSO during a simulation. |
| **6.1** | The Photometry render mode shall provide an “up-close” rendering of the RSO for the purpose of generating photometry data. |
| **6.2** | The OSCOM render mode shall provide a “telescope view” rendering of the RSO for the purpose of simulating the OSCOM system and supporting image processing algorithms within the project. |
| **7.0** | The program shall provide run modes which determine the speed and accessibility of each simulation. |
| **7.1** | The *Real-Time* simulation mode shall render simulations in real-time for the user to observe |
| **7.2** | The *Back-End* simulation mode shall render simulations as fast as possible without displaying them to the screen. These renderings are recorded and may be observed in *playback* mode. |
| **8.0** | The program shall provide a graphical user interface in which simulations are rendered and photometry data is displayed |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COMPASS – V 1.0.0 Requirements Table** | | | | |
| *No.* | *Functional Requirement* | *Operational Requirements* | *Justification* | *Validation* |
| *1.0* | *The program shall generate, record, and plot photometry data for the RSO during all simulations.* | *The sun shall be the only light source considered and it shall emit only white light.*  *No ambient light shall be present.*  *The surface albedo of the RSO shall be assumed to be exactly 1.*  *The observing sensor shall collect 100% of the light reflected by the RSO that falls within its solid angle.* | *Generating accurate photometry data is a fundamental requirement of this program.*  *The stated assumptions shall be maintained in order to ensure simplicity and ease of validation within the initial photometry model.* | *Compare to theoretical calculations made according to the same assumptions.*  *Rendering simplified models in a fixed orientation at a fixed distance, calculate the brightness of the object and compare to simulation results. Repeat for several test cases.* |
| *1.1* | *The sample rate for collecting photometry data shall be defined by the user.* | *The sample rate shall not be allowed to exceed the framerate of the application (60HZ).* | *Sample rate should be definable according to the sensor being modelled.*  *The sample rate is physically limited by the application framerate. Thus, it would be innacurate to assign higher sample rates.* | *N/A* |
| *2.0* | *The program shall provide a manual mode in which the user may define the simulation parameters and observe their effects on photometry data.* | *The user shall define the direction of incoming light and the orientation of the RSO within three degrees of freedom.*  *The angular velocity of the RSO may be defined within three degrees of freedom and up to any rate less than 30HZ.*  *The altitude of the RSO may be defined between 150km and 36,000km.* | *A manual mode provides the user with an intuitive means of correlating the specified parameters to photometry data.*  *The target frame-rate of this application is 60 fps. Any angular velocity at or above 30HZ would yield rendering errors due to aliasing.*  *RSO’s at or below 150km will decay quickly. RSO’s at or above geosynchronous altitudes (~36,000km) are beyond the scope of this program.* | *N/A* |
| *2.1* | *In manual mode, the user shall have the ability to rotate the camera to any desired orientation.* | *Camera orientation shall be defined within three degrees of freedom while always maintaining a focus on the RSO.* | *Allows user to inspect the entirety of the model under any given set of lighting and orientation conditions.* | *N/A* |
| *3.0* | *The program shall implement a pass mode in which the RSO shall be rendered in orbit from the perspective of a ground-based observer and photometry data shall be generated and recorded.* | *The orbit of the RSO shall be modelled and propagated according to its TLE or a set of user-defined Keplerian elements.*  *The time and duration of each pass shall be defined by the user.*  *The Orientation and Angular velocity of the RSO shall be accurate according to its location in space relative to that of the observer, its orbital motion, and a set of user defined orientation and angular velocity parameters.*  *The direction of incoming light shall be accurate according to the relative positions of the sun and the RSO at the time of observation.* | *The purpose of pass mode is to simulate the conditions under which actual observations occur. These simulations may be used to analyze the correlations between photometry data and the motion/orientation of an RSO. It may also be used to predict the photometry curve of an RSO in any particular orbit and any orientation.* | *Compare to actual photometry curves in which the orientation of the RSO is known. Check for similarities in the shape and magnitude of the curves (will not be exactly the same given the simplified photometry model).*  *Compare to photometry curves generated according to alternate methods.* |
| *3.1* | *The program shall notify the user when the RSO is not in direct sunlight at the time of observation.* | *This program will implement the algorithm created by last year’s senior design team. (REVISE THIS ONE)* | *Observing a satellite when it’s not illuminated is pointless.* | *Check according to different RSOs and the associated times in which they are illuminated.* |

Architecture

CpsWindow

* Represents the main program window
* Has a GLWidget and a CpsGraph
* Has a CpsSimulation

CpsSimulation

* Represents a set of simulation parameters

CpsGraph

* Live-Graph Widget
* Graphs brightness in real-time

GLWidget

* Handles OpenGL initialization
* Handles Rendering
* Handles GL viewport resize