Summary

The SatStatSim project achieved a great deal during it's 4 month contract, but the project remains unready for use. A team of Justin Corwin, Keith Schafer, and Joshua Truett produced design documents, 3D models, simulator code, and Unity® project files. But the final products, the usable designer/simulator game, and modular parts library, are still in the prototype phase.

The Team intends to continue working on the tools to produce a usable game and launch as an open-source project as described in initial planning document, although an interim website has been put up in the meantime.

Research

In order to accurately simulate the characteristics of satellites, we had to establish many of the physical properties of both satellite parts and aerospace materials in general, which turned out to be surprisingly problematic. The majority of satellites we could find records of were made of custom parts, for which very little exact data was public. Even as standardized a component as solar panels (which are made by relatively few companies) under the same type name can have different backing materials that will drastically change the weight and performance.

Unfortunately, providing the kind of infinite variability of one-off custom parts for every design would make the interface both needlessly complex and difficult to support. As a compromise we did a survey of parts and their average properties where they were available; this was used to make a standard library of parts with middle of the road characteristics and somewhat modifiable sizes and types to simplify both use and development.

In addition to parts research, we also did a survey of satellite frame and payload types, both scientific and commercial in order to make a useful variety of options for holding the pieces together. Launcher families were investigated simply to give us some guidelines on size and weight costs and how they affect satellite design.

Development

The primary requirements of SatStatSim were building a simulator and a designer that interact with each other to provide a usable picture of satellite behavior. To constrain the designer, we made a library of basic parts and satellite frames to support them. Mr. Schafer made 3D models for these parts and some other necessary objects, while Mr. Truett assigned them characteristics and behaviors.

Part Name	**Brief description (for art direction)**
Solar Panel-integral	Solar panels, variable number of cells, mounts in-line against other parts

Solar Panel - Cubesat form factor	Tiny flip-out panels, possibly irregular layouts of cells, single stuff board or array of boards, no more than two wide or covering sat package
Solar Array -square wing	Fold out square solar panels, 4x4 or larger variable size, mounts on extending pole, small flip-outs, or hidden accordion compartments
Solar Array -rectangular wing	Fold out solar panels, variable width and length, mounts on extending pole, small flip-outs, or hidden accordion compartments
Solar Array -LARGE rectangular wing	Large number of folded solar panels. Deploys from pods or hidden accordion compartments. The Miura fold? SERT?
Microwave dish antenna	Flatish, low curve dish. Small com-sat style with single small centered arm. Dark/black, round. A few feet in diameter.
Microwave dish antenna -LARGE	TDRS-style dark/black dishes with internal metal bracing, unfolds on deployment
Microwave box antenna	
Microwave spike antenna	
Multichannel Antenna tree	
Antenna Whip	
Long Antenna Whip	
Rigid Spar -metal	Long metal connection pole
Rigid Spar -carbon fiber	Long plastic-looking connection pole
Tether -monofilament	Space rope. Attachment points on both ends.
Tether -braided	Thicker space rope. Attachment points on both ends.
1U Cubesat Frame	Open square aluminum frame for holding mission equipment. Mount points for flip-out solar panels and internal circuit boards optional.
2U Cubesat Frame	Two 1U Cubesat Frames connected

3U Cubesat Frame	Three 1U Cubesat Frames connected in a line
Polyhedron space frame -scalable	Buckminster Fuller in spaaaace. Metal frame. Treat each face and edge as a standard mount point for payload or other frames
Polyhedron solid frame -scalable	Buckminster Fuller in spaaaace. Metal sided structure. Treat each face and edge as a standard mount point for payload or other frames
Cube solid frame -scalable	A large cubical closed satellite body, metal-sided structure
Cube space frame -scalable	A large cubical satellite frame, open metal structure
Cylinder solid frame -scalable	A large cylindrical satellite body, metal-enclosed structure
ESPA Ring	Wide, flattened metal ring with open circular attachment points, symmetrically placed radially
External Camera	Space GoPro in a plastic mount
Telescope lens	Micro-sat size: hole with camera lens looking thing inside
Sensor port -scalable	A hole, optional flip-out cover
Cold gas nozzles	Small RCS-style nozzles
Hot gas nozzle -scalable	Using HTV-II transfer vehicle as example. Small metallic nozzle cone Using Progress M series as example. Almost flush hole on extended cylindrical metal subsection, with small inner ring for exhaust
Rocket engine bell -scalable	Bell-ended engine part, visible extending far outside th craft.
RCS Cluster	External cluster of small nozzles or inset pits
Adapter mount -scalable	
Bolted connection square -scalable	

Bolted connection circular -scalable	
Actuated connector	
Explosive separator	
Explosive separator ring -scalable	
Motorized hinge -scalable	Small metal fitting attached to both sides of moving components, with smaller metal box to one side
Shrouded sensor port cone -scalable	A white conical covering, pole mounted
Glass port	A thick, round window
Phased antenna array panel -scalable	Modern: Flat panel with meshed holes in metal frame
Manual access port -scalable	
Metallic shielding -scalable	Someone wrapped the ship in tin foil
Foamed shielding -scalable	Solid tiles of hard foam, like the space shuttle
Quilted shielding -scalable	Heavy stitched cloth panels used for medium strength heat shielding on delicate components.
Indicator light	
Panel display -scalable	
Spherical pressure tank -scalable	Spherical metallic tank, matte surface with square connecter at the base
Cylinder pressure tank -scalable	Long cylindrical metallic tank, matte surface with square connecter at one end
Collapsible pumped storage -scalable	Plastic, balloon-like structure. Expands into rounded shape with hexagonal, honeycombed sides

Mr. Corwin wrote code for simulating of the parts and the satellite as a whole. The choice was made to simulate them as a series of separate systems, rather than attempting a full, composite physics simulator:

- Having ascribed weight and geometry to the objects, we use the Unity® supported built-in NVIDIA® PhysX® engine for purely physical simulation, rockets and stabilizing systems are simple forces to integrate into the final motion.
- Heat build-up and low-temperature issues are tracked both per-part and then integrated to simulate the thermodynamic issues via simple summing of heat transfer coefficients.
- Electrical sources and draws are assumed to all have common wiring, and undervolt and overvolt conditions are noted, as well as tracking exhaustion/fuel use for batteries/fuel cells.
- Sun incidence is calculated from the orbital and positioning indicated for the satellite, and the heat/solar power generated on any panels and surfaces is attenuated therefrom.

Problems

1. There were a number of simulation issues, as allowing people to add new parts in a generic way allow for a variety of strange errors and conditions due to unforeseen interactions, particularly in thermodynamics and physical modeling.
2. Interface issues arose as we attempted to strike a balance between providing all relevant information required in each stage of the game while staying readable. Also, providing summaries and updates during the simulation phase proved to require some significant re-plotting.
3. The Design phase had issues with positioning, as providing a usable tool for placing a part in the satellite efficiently without restricting parts to particular snap points required some smart logic to evaluated surfaces as to whether a part could adhere successfully without being wedged, or awkwardly connected by too small a portion.

Finances

As outlined in our proposal, the primary costs of this development period were labor. $15600 was dispersed to Mr. Schafer, Mr. Truett and Mr. Corwin over the 3.5 months of full-time work. $2100 was reserved for Unity®Pro, 3ds Max®, and associated programs licensing. The remaining $2300 are unassigned, reserved for web hosting when the project goes live, and any administration costs to come.

Future Plans

Our primary goal at this time is the publication of a Playable Release. The disparate systems and parts we've developed are unusable at this time, so both getting feedback and recruiting further funding or participation is difficult. Once we have a usable game, we can put up a more complete website, and open up our codebase and parts library definition files for open source participation.

It's difficult to estimate the timeline for this release, as all work is happening as spare time is available for individuals on the team. Our hope is to have a usable release in the next few months.

Once we have a playable release, our planned upgrades center around an increased Parts Library with more scientific and commercial packages to provide more flexibility and satellite options. We also plan to open up the part definition to allow people to add new parts.

We believe it would also be useful to have pre built Satellite collections of designs we've found to work or have interesting problems for users to play with attempting to fix or improve upon.

Conclusions and Lessons Learned

Game Development Issues

Significant parts of the development time were taken up in issues specific to games, control programming, event loops, graphics programming, which were unfamiliar to us, and resulted in unplanned delays.

The primary background of the development team is scientific, and practical issues of interacting with game engine APIs and establishing exact values and dimensions as required for modeling took up much more time than anticipated as team members did research and gained experience as necessary.

.

Scheduling

Some delays were due to poor process planning, as one or both other team-members had to wait for another to complete a task required to move forward. As Mr. Schafer had full-time funding for two months, and Mr. Corwin and Mr. Truett were funded for three, it was difficult to synchronize for best efficiency around other commitments and work.

In the future, we hope the more complete base of code and parts will allow people to make good progress on their own without relying on others to complete necessary infrastructure. A complete product is simpler to work on, as changes are easier to test and performance is obvious.

Science vs the Programming as Tasks

While our design process and initial planning focused on the simulationist aspects of orbitals, rockets, and solar panels, the more mundane aspects of game development turned out to be the most key to our timeline. While some delays could be attributed to unfamiliarity with game development or the use of new tools, our tendency to focus on the interesting sub-problems of the simulation led to our neglecting to design and plan for the infrastructure and required game elements.

Our preference for working on scientific subjects and interesting aerospace maths led to a lack respect for the difficulty of some tasks like resolving GUI designs and interacting with game APIs, which contributed to our slower than expected progress.