

# Human outpost creation using multiple data sets and computational design

Thomas Lagarde<sup>1</sup>

*Space Architecture Technical Committee, University of Houston, Paris, France, 75016*

Matko Brandić Lipiński<sup>2</sup>

*University of Zagreb Graduate, Gateshead, NE10 0DU, United Kingdom*

**Designing human outposts in extreme environments is a particularly challenging process due to the multiplicity of factors involved. Resources utilization, protection from radiations, micrometeoroids impacts, atmosphere containment are just some of the co-dependencies that Space architects and mission designers have to consider in the preliminary design phase.**

**To facilitate the process, we considered a new design path that has found application in both engineering and architectural design tools: the computational design approach. Thinking at the different factors and constraints as project parameters, we are designing a software tool based on co-dependent variables to enable scalability while significantly decreasing the process complexity. The parametric software tool or PST considers a large range of factors (weather, terrain conformation, in situ resources accessibility) and constraints (survivability requirements, safety requirements, and infrastructural requirements) to allow a data-based design process, reducing the chance for human errors and inconsistent design assumptions.**

**This paper describes the tool functionalities and the rationale behind the constraint and factor definition. While the tool is still considered in an early-development stage now in which this research is presented, the paper outlines guidelines for future developments and plans for future work on the code. At the same time, new data sets about extreme environments coming from current and future missions are considered of primary importance for the development of reliable data models about the environmental constraint used by the tool.**

## Nomenclature

SIMOC	= Scalable Interactive Model of an Off-world Community
ECLS	= Environmental Control Life Support System
RECLS	= Regenerative Environmental Control Life Support System
CAD	= Computer Aided Design
Rhino	= Rhinoceros 3D
Grasshopper	= Rhino visual scripting objects
PV	= Photovoltaics
3D	= Three spatial dimensions
JSON	= JavaScript Object Notation Format
.epw	= EnergyPlus Weather Data Format
Python	= Interpreted high-level general-purpose programming language
PST	= Parametric Software Tool
GUI	= Graphical User Interface

---

<sup>1</sup> System Engineer, SICSA, [thomas.lagarde@spacearchitecture.fr](mailto:thomas.lagarde@spacearchitecture.fr)

<sup>2</sup> Software engineer, [matko.brandic.lipinski@gmail.com](mailto:matko.brandic.lipinski@gmail.com).

## I. Introduction

The parametric software tool or PST can help with the design process of habitats, greenhouses and other supporting structures in extreme environments (sea, space and others). There has been in recent years an increase in the production of data for that field [1] but most of that data is still not directly accessible in the Computer Aided Design (CAD) software used by most professionals and students.

The absence of direct connectivity means that the documentation phase of a new project can be long and complicated.

Senior engineers and architects have memorized most of the rules and standards that apply to the projects they are familiar with but a new kind of construction requires a long process of self-formation. Inexperienced designers also need to go through the same process when they arrive in the professional world.

By using the tools we have at our disposal, such as Grasshopper, HoudiniFX and Python, we can build a PST to improve the workflow and the communication of the stakeholders of a project. We are using different concepts and techniques to achieve this result, one of the most important being Computational design.

Computational design is a field of CAD that includes many different concepts: designing with data, processing power, parameter setting, generative design, 3D modeling, and visualization tools [2].

This paper describes the reasons and the necessary steps for creating a direct link between external data sets like weather data, energy data, and CAD tools. It then becomes possible to draw lines, shapes, volumes, and dimensions directly and automatically into the design software. The designer doesn't need to translate and import external raw data, he/she is being helped automatically in their design process.

The construction of this PST will rely on three different parts that can be activated at will. The first one is parametric design, it starts with a set of data. This data set is linked to parameter settings, which can scale up or down a design and influence the shapes created. This process is still based on the designer's input. It gives more flexibility to designing objects since parameter settings can allow a project to be adapted to changing conditions (for example, a bigger crew, less energy and others).

The second component, generative design is based on certain constraints in which the scripts will create multiple propositions, analyze them and find out which ones respond best to the limits set by the designer. This process allows the creation of a shape or object that is the optimal solution based on the input from the user.

The last component and probably the most important is the verification layer that can tell the designer if his current design is likely to fail or not, this can be implemented by analyzing the current design and comparing it to optimal solutions calculated internally by the PST.

## II. General Architecture

For the construction of the PST, we will study and incorporate relevant projects and datasets that have been completed or are currently being developed. Here you will find a breakdown of the datasets and how the PST will incorporate them and use them to help the designers.

### A. Scalable Interactive Model of an Off-world Community SIMOC dataset

SIMOC is a research and educational platform for the simulation of a hybrid mechanical and biological regenerative life support system or RECLSS, in a Mars environment. It is hosted by the National Geographic organization and was created by an international team headed by Kai Staats. SIMOC is freely available online [3] and can be used by anybody at any time. The successful mission data sets collected through experimentation on SIMOC will help to balance the required elements to ensure the survival and well-being of a human crew on Mars. The initial parameters of the simulation are the dimension of the greenhouse, the dimension of the habitat, the number of photovoltaic (PV) arrays, the number of battery packs, the type and number of ECLSS modules, the size of the storage tanks and the type and quantity of edible plants.

The location of the base is limited to Mars for now but the code can be modified to include other extreme environments. It also aims to determine the minimum amount of cargo to be shipped and the minimum energy consumption for the duration of the mission.

The team behind SIMOC has developed a comprehensive system based on published research [4]. SIMOC is an agent-based model, which is characterized by the simulation of the actions and interactions of autonomous agents. These independent agents exchange currencies such as Oxygen ( $O_2$ ), Carbon Dioxide ( $CO_2$ ), or Water ( $H_2O$ ). There are nine general categories for these agents, sub-divided into smaller individual categories. These categories are Inhabitants, ECLSS, Agriculture, ISRU, Structure, Fabrication, Power, Mobility, and Communication. More information can be obtained by reading the free and downloadable guide [4].

The SIMOC interface is web-based and can be accessed on any device. An example of the interface is shown in Figure 1.



**Figure 1. Main SIMOC interface**

The main interface of SIMOC shows numerical and graphical representations of the raw data being created after configuring the mission. Those numbers and graphs can be used by a designer to build a 3D representation, for example, the amount of  $CO_2$  stored can be used to size the storage areas.

After a simulation is complete, the data generated can be exported. The data is saved in a JSON file; an open standard file format for representing structured data. This result file or dataset is used by the PST as one of the key elements that will help the designers make good decisions. As an example, it will quantify the quantity of food required for the crew and how much atmosphere is regenerated each day.

## B. Human factor requirements dataset

To build these datasets we are using documents created by space architecture specialists and engineers with a demonstrated track record working with crews and habitats [10] [11]. We first extract the important parameters such as mobility and trafficability spaces or personal space. These concepts are dependent on many factors but minimum requirements can be extracted from previous research [11]. The minimum value is set and is used as a reference for one crew in an Excel sheet. That data can be read and interpreted using Grasshopper for this version of the PST, see figure 2.

This value can then be used and combined with other parameters such as the number of crew and the duration of the mission. The value can be increased to offer more comfort if the budget for the mission allows it, this is also a parameter that can be modified with the PST.

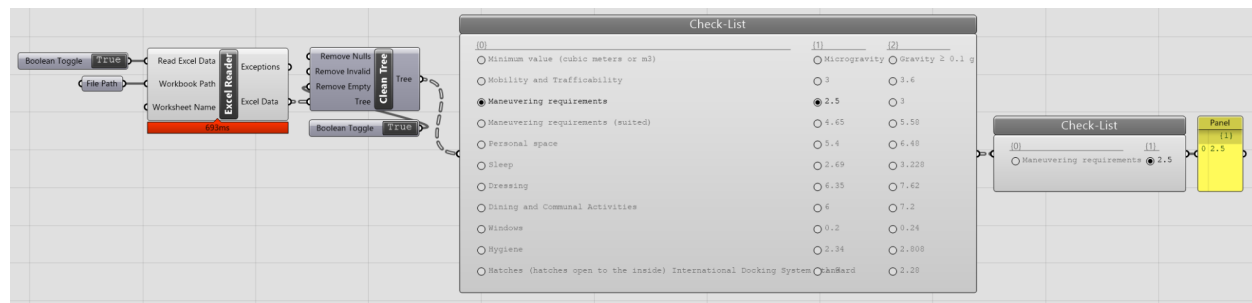


Figure 2. Creation of a dataset for minimum volume for humans, imported in Grasshopper

Many more data sets will have to be created in future iterations such as minimum lighting requirements or basic hygiene elements and the volume required for their operation. This subject will be addressed in the next version of the PST but those new datasets will rely heavily on previous research and standards [10].

Some aspects of the mission and the architecture are really hard to integrate into the PST, such as aesthetics, cultural differences and individual preferences. For those domains, the designer will have to use their own experience and judgment to make a correct decision.

## C. Materials and loads on the system datasets

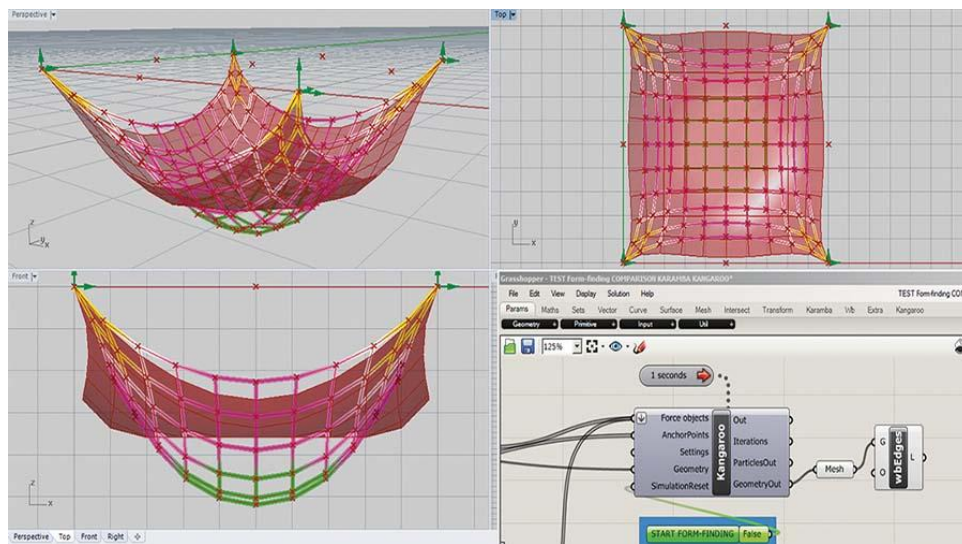
The material data set is also based on Excel sheets for the current phase of the PST. The materials that are currently listed have been picked specifically for their resistance to extreme environments, see Figure 3. The PST will select the right material for the right application when it identifies what the designer is building.

Número	Material Name	Thickness (mm)	Elastic Modulus N/m <sup>2</sup>	Poisson's Ratio	Shear Modulus N/m <sup>2</sup>	Density kg/m <sup>3</sup>	Tensile Strength N/m <sup>2</sup>	Compressive Strength N/m <sup>2</sup>	Tenacity N/tex	Yield Strength N/m <sup>2</sup>	Thermal Expansion Coefficient	Conductivity W/K/m	Specific heat J/K/kg	Price \$/kg
1	Aluminium 6061		69000	0.33	26000	2700	124			55.14	2.40E-05	167	1300	
2	Beryllium Copper		125000	0.3	50000	8250	469			172	1.67E-05	105	376	
3	Brass		100000	0.33	37000	8500	478.41			239.68	1.80E-05	110	401	
4	Rogers 4350 B					1860						0.62	390	
5	Heat pipe					3000						10000	70	
6	Teflon		552000000			2200	34500000			21700000	1.70E-04	0.25	970	
7	FR4					1850						0.25	950	
8	Silicon		112400	0.28	49000	2329				120		124	700	
9	Zerodur					2530						1.65	821	
10	Titanium		120000	0.32	4600	4370	937			910	8.50E-06	6	502	
11	Carbon fiber-reinforced polymers		16000000000	0.1	5000000000	2000	1206582524	2.5E+11		215000000				
12	Kevlar		1.12E+11	0.36		1440	3200000000			2.08		0.04	1.42	
13	Vectran		75000000000			1400	3000000000			2.15		1.5	1.1	
14	Fused silica glass		73000000000	0.17	31000000000	2200		1			5.50E-07	1.36817	740	
15	Polyethylene Cross-Linked		600			950	18	14						
16	Rubber		6.1	0.49	2.9	1000	13.78	0		9.23	0.00067	0.14		
17	CFPR HoneyComb Sandwich													
18	Carbon fiber C T50-4.8/280		2.8E+11			1780	4800000000							
19	Dyneema													
20	Darwin bark spiders' silk (drag line)													
21	Silicon Carbide													
22	Palladium microalloy glass													
23	Buckypaper													
24	Graphene													

**Figure 3. Creation of a library of adequate materials for extreme environments**

As an example, when the designer wants to use a membrane (greenhouses, habitats, others) in the design, the PST can simulate different scenarios and quickly extract some key parameters.

If the tensile strain in this construction is high, the PST will choose the appropriate material for the shape, Vectran is a good solution in this case according to the characteristics compiled in Figure 3. This material will be selected to run physics simulations (quasi-static, frequency) and apply its results to the desired shape. Those physics simulations are run in Grasshopper at the moment because tools have already been developed for that platform such as Kangaroo or Karamba see Figure 4.



**Figure 4. The reaction of a surface to external forces in Rhino using Kangaroo**

The designer will be able to see how the shape created and the material picked by the PST behave in the environment they are designing for. Many parameters can be set for the optimization of the design, weight or cost for example can greatly impact the mission requirements (available payload, initial budget).

The designer will have at their disposal a standardized set of data that can be upgraded as new materials or new alloys are developed. The current data has been extracted from previous paper and open-source datasets[5, 6, 7, 8].

After selecting an environment (Mars, Moon, Earth and others), the designers will have access to the existing and expected conditions of the celestial body. This will include i.e. gravity or pressure, as part of the environment dataset. Currently, our team has only created a dataset for Mars with weather and gravity conditions, see Figure 5.

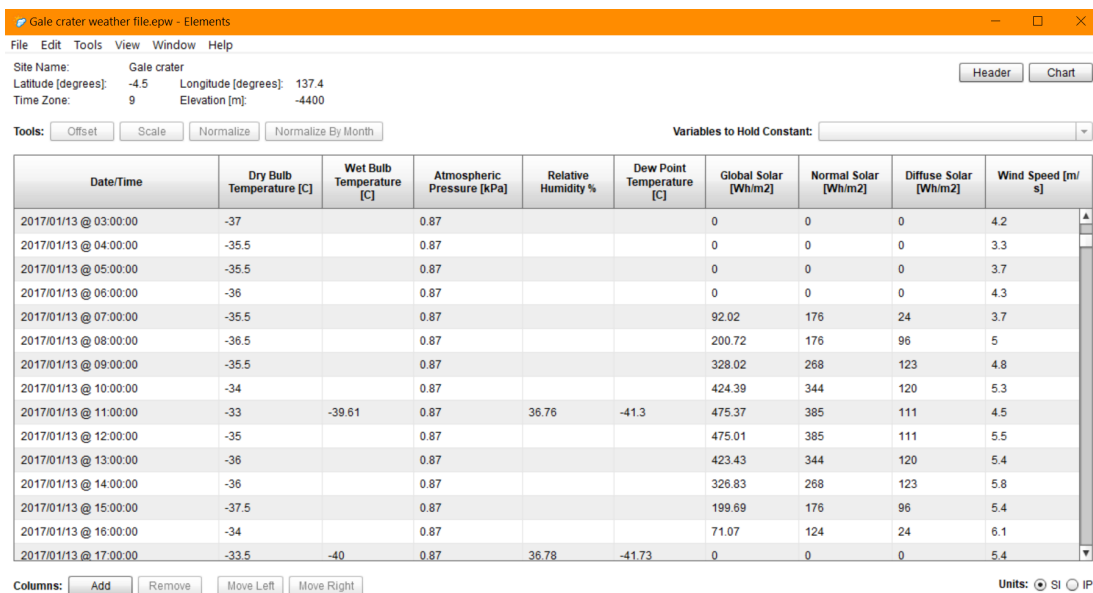


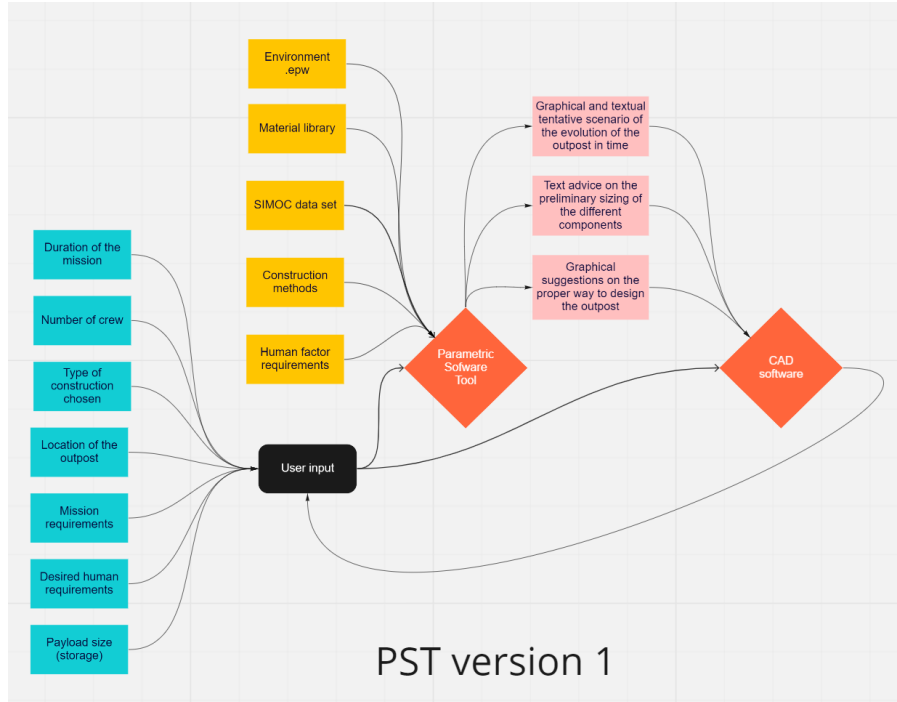
Figure 5. Mars weather dataset using the EPW standard

These external loads extracted from the data set will be applied to the construction and live simulations will be executed while the designer creates shapes in the CAD software. Pressure for example will be a very important factor in the design of the base. The PST will be able to advise you on the possibility of sharp angles and/or if rounded angles are a requirement that has to be met for the structure to work properly.

## D. Front-end and back-end development

Our approach to software design and development is guided by the planned evolution of the PST, from the Minimal Viable Product (MVP) to the future capability of the PST, and beyond. For the programming language, we decided to use Python, for its portability and data handling.

The first milestone consists of a text-based method in which the user inputs the number of crew members, the duration of the mission (in days), a celestial body for which the user is designing, together with a location on that celestial body (i.e. Gale Crater on Mars), mission requirements, desired human requirements (i.e. Net Habitable Volume) and the quantity of storage being sent. The user is able to specify the type of structure they want to design: crew quarters, greenhouses, a science center, or a laboratory, see Figure 6.



**Figure 6. The general architecture of the current version of the PST**

Using the different databases we collected and/or created, the PST outputs the minimal dimensions for the habitat as well as the payload needed to sustain the crews for the duration of the mission (see Figure 11). Those datasets can be maintained with the most up-to-date information related to Human factor research or the Moon and Mars environment. For the MVP, that method is accessible to the user via a Graphical User Interface (GUI) where the user inputs the values and is given the output in the form of a JSON file.

The different datasets consist of data extracted from SIMOC, environment data on various locations, a material library, data on various construction methods, and data on human factor requirements. The environment data is manipulated by reading the files using *epw*, a lightweight Python package for EnergyPlus Weather files.

The SIMOC simulation data payload is rich with numbers and categories. See Figure 7 for an example of how the information is being written and compiled, this extraction was done in PyCharm using the Matplotlib library in Python.

```

14
15     for day in data['storage_capacities']:
16         print(day)
17         print(data['storage_capacities'][day]['air_storage'])
18         print(data['storage_capacities'][day]['water_storage'])
19         print(data['storage_capacities'][day]['nutrient_storage'])
20         print(data['storage_capacities'][day]['power_storage'])
21         print(data['storage_capacities'][day]['food_storage'])

```

if \_\_name\_\_ == '\_\_main\_\_': for day in data['storage\_capaci...

main X

```

{'1': {'biomass_totl': {'value': 0, 'unit': 'kg'}, 'sold_n': {'value': 100.032625, 'unit': 'kg'}, 'sold_p': {'value': 0, 'unit': 'kg'}}}
{'1': {'enrg_kwh': {'value': 970.613446, 'unit': 'kWh'}}}
{'1': {'food_edbl': {'value': 83.138244, 'unit': 'kg'}}}
68
{'1': {'atmo_o2': {'value': 255.947674, 'unit': 'kg'}, 'atmo_co2': {'value': 3.168302, 'unit': 'kg'}, 'atmo_n2': {'value': 0.0, 'unit': 'kg'}}}
{'1': {'h2o_potb': {'value': 987.893424, 'unit': 'kg'}, 'h2o_urin': {'value': 0.985615, 'unit': 'kg'}, 'h2o_wste': {'value': 0.0, 'unit': 'kg'}}}
{'1': {'biomass_totl': {'value': 0, 'unit': 'kg'}, 'sold_n': {'value': 100.032625, 'unit': 'kg'}, 'sold_p': {'value': 0, 'unit': 'kg'}}}
{'1': {'enrg_kwh': {'value': 967.302543, 'unit': 'kWh'}}}
{'1': {'food_edbl': {'value': 82.886576, 'unit': 'kg'}}}
69
{'1': {'atmo_o2': {'value': 255.860892, 'unit': 'kg'}, 'atmo_co2': {'value': 3.188466, 'unit': 'kg'}, 'atmo_n2': {'value': 0.0, 'unit': 'kg'}}}
{'1': {'h2o_potb': {'value': 987.230092, 'unit': 'kg'}, 'h2o_urin': {'value': 2.434656, 'unit': 'kg'}, 'h2o_wste': {'value': 0.0, 'unit': 'kg'}}}
{'1': {'biomass_totl': {'value': 0, 'unit': 'kg'}, 'sold_n': {'value': 100.0348, 'unit': 'kg'}, 'sold_p': {'value': 0, 'unit': 'kg'}}}
{'1': {'enrg_kwh': {'value': 963.273568, 'unit': 'kWh'}}}
{'1': {'food_edbl': {'value': 82.634908, 'unit': 'kg'}}}
70
{'1': {'atmo_o2': {'value': 255.77456, 'unit': 'kg'}, 'atmo_co2': {'value': 3.20713, 'unit': 'kg'}, 'atmo_n2': {'value': 0.0, 'unit': 'kg'}}}
{'1': {'h2o_potb': {'value': 991.31676, 'unit': 'kg'}, 'h2o_urin': {'value': 0.684656, 'unit': 'kg'}, 'h2o_wste': {'value': 0.0, 'unit': 'kg'}}}

```

Figure 7. Data is easily accessible with SIMOC JSON payload

Once extracted it can be easily accessed and manipulated, i.e. plotting energy production per day, as shown in Figure 8.



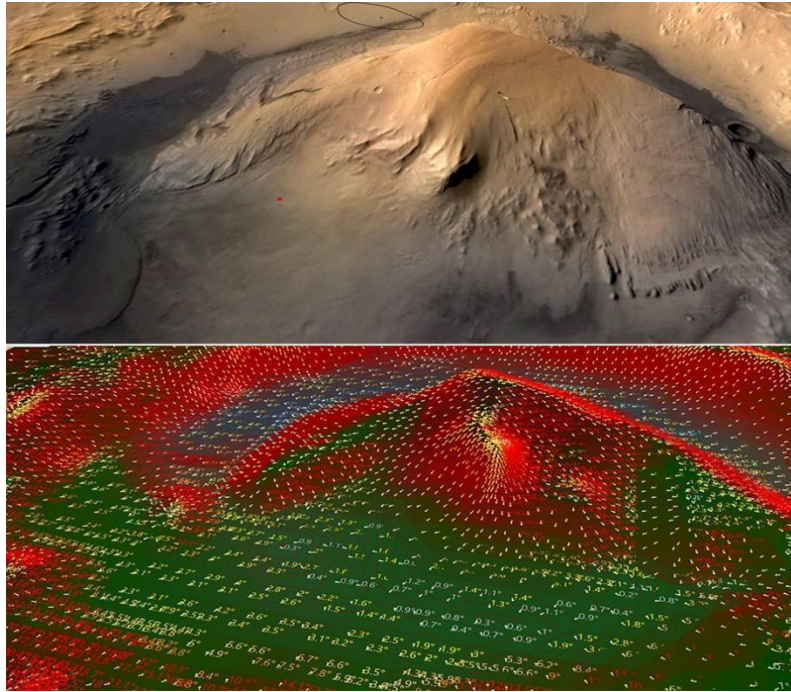
Figure 8. Using the SIMOC generated data to plot the energy production per day on a 240-day mission of 4 crew members

Other than the habitat dimensions and the payload needed for the mission, the tool adds a visual representation of the habitat, as well as the environmental impact on all the outer walls of the habitat. That way, the tool informs the



user of the atmospheric pressure on that specific site, the radiation levels, and the characteristics of the ground, see Figure 9. for an example of the graphical representation of data on a topography.

Not only does it inform, but also gives out warnings if the given location is on steep ground, the radiation levels are too high, or any other environmental factor that would make the construction of the habitat unfavorable in that particular location.

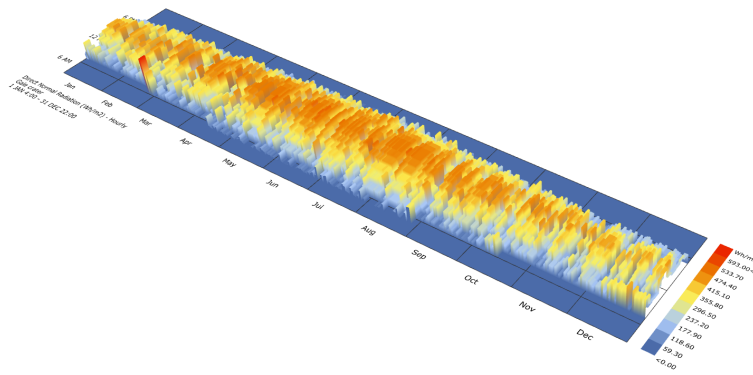


**Figure 9. Gale crater ground data layer on/off**

### **III. Practical Application**

In order to demonstrate the capability of the PST, we will go through the design of a greenhouse. To build a greenhouse in a martian environment, a designer will have to integrate a lot of different information, radiation is the first subject that they should study. Experienced designers will know that due to the low amount of solar radiation that reaches Mars, having a transparent Greenhouse is not sufficient to properly grow earth-based plants. A good design will require supplemental artificial lighting.

The PST is used to optimally orient the greenhouse and calculate the quantity of power required for lighting. For this purpose, it will use the solar radiation data, the sun vectors coming from the .epw file, see Figure 10.



**Figure 10. A 3D representation of the annual solar radiation intensity at Gale Crater.**

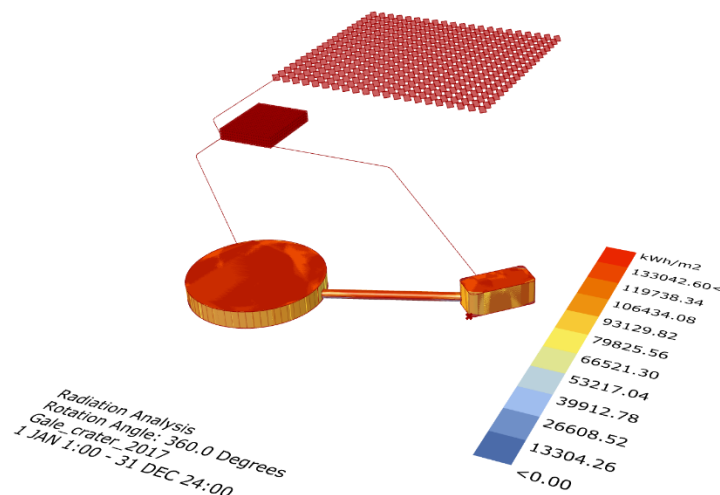
With the .epw data set, the tool will orient the building at different angles and will calculate which orientation will provide the maximum solar exposure with the least amount of heat collected.

Once the correct orientation is determined and the location is chosen, the greenhouse is placed in the model window, its size is determined by the SIMOC dataset. After being placed, it can be populated with the right number of plant trays.

The SIMOC uses plant trays [4] that measure one meter by one meter. In the SIMOC simulation used for this paper, it was determined that the crew needed 20 trays of wheat, 30 trays of cabbage, 10 trays of strawberries, 50 trays of radish, 50 trays of red beet and 50 trays of onions.

Future development of the PST will automate the process of placing those trays in the greenhouse. It can be done by giving certain qualities to each plant (pruning required, luminosity, growing-time) and selecting some spaces that are better configured to accommodate those qualities. For example, the trays placed closer to the entrance will grow faster and require more attention. Those principles are also used in permaculture [9].

The number of PV arrays and batteries required for the mission is also quantified by the simulation, it studies the needs of the greenhouse and its general use. The PST can optimally orient and place the PV arrays and the batteries on the topography and can calculate the amount of radiation that will hit each structure, see Figure 11.



**Figure 11. Quantity of solar radiation that hits the base**

That information about solar radiation on the structures is useful for cooling and heating purposes, and future evolutions of the PST will allow the designer to automatically calculate the quantity of power required for maintaining a comfortable temperature for the crew. The current PST can already tell the designer how much power can be generated by a panel in a year.

The footprint of the greenhouse is also an important information, the PST can calculate how much material needs to be removed for the greenhouse to lay flat on the topography. This gives the designer helpful information and can improve his decisions regarding the layout of the base.

The greenhouse is a good example of the utility of the PST in the planning phase. The designer is given the tools they need to correctly size a base and to choose the best location for its different parts (best implies the most economical in terms of setup time, storage transported, heating loads and other parameters).

#### **IV.Future developments**

The existing tools for Earth-based construction can be used for design in extreme environments. The creation of a PST to link all relevant data sets and 3D design software is a necessary step in exploring new pathways of simulation and planification.

The possibility of using existing software and data sets allows designers to start their projects with a strong basis, validated by previous research and successful simulations.

As the project evolves, it might follow a transition from surface-based geometry to voxel-based, where the rasterization logic of voxels will provide efficiency in translating various input data into a design decision.

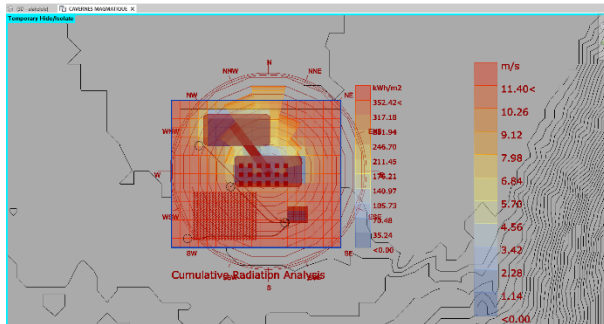
The current version of the PST called A1.01 is built using Grasshopper and can be used on Rhino and Revit. Next iterations will be written in Python and will have their own GUI, they will still depend on grasshopper as there is a large number of tools already available on this platform.

Efforts have already been made to reach out and engage the communities behind previous and similar projects [8, 9, 10]. The process of modifying and validating the existing tools will require the involvement of multiple communities working together. The authors welcome the involvement of everybody.

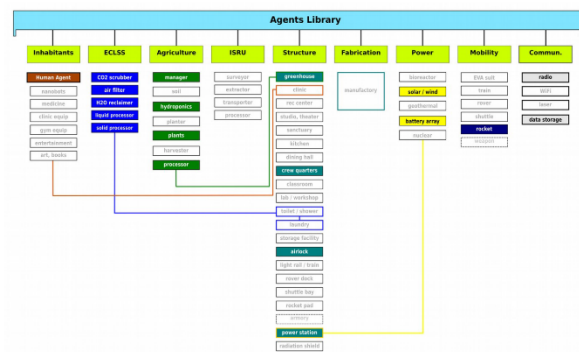
Please refer to the websites that the authors use for communication. The website used are Github, Researchgate, Academia and others. You will have access to the updated material, tutorials and a list of people and communities (including the authors) you can contact to help take part in this project.

The authors hope that this software will help to better connect the world of CAD and extreme environment projects.

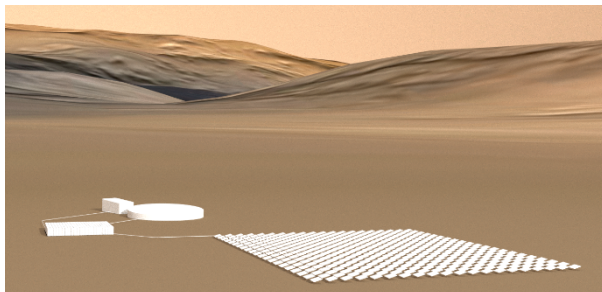
## Appendix



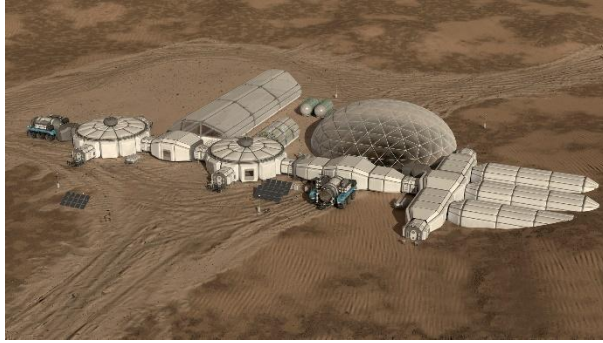
Integration of the PST within Revit using Rhino.Inside.Revit



The list of agents in the SIMOC project



The correct amount of PV panels, battery packs, and volume of pressurized environments for four crews and RECLSS according to one of the SIMOC simulation



Possible Mars base, presented by the SIMOC website. This base was designed by Bryan Versteeg Copyright © SIMOC

### Acknowledgments

The SIMOC project team (Kai Staats, Project Lead, Iurii Milovanov, Lead Server Developer, Ezio Melotti, Lead Front-end Developer, Sheri Klug Boonstra, Associate lead, Don Boonstra, Educational Developer, Bryan Versteeg, Space Habitat Architect & 3D Artist

SICSA

The AIAA Space Architecture Committee on Standards

The Grasshopper community

The Ladybug community

The Openstudio community

The Energy+ community

All the people that dedicated their time and energy to provide the community with new tools to improve their workflow and their design process

Our families and our colleagues in the industry

### References

- [1 ] Accredited Programs in Space Architecture (repository of training programs)  
<http://spacearchitect.org/how-to-become-a-space-architect/>
- [2] Inês Caetano, Luís Santos, António Leita, Computational design in architecture: Defining parametric, generative, and algorithmic design, *Frontiers of Architectural Research* (2020) 9, 287e300
- [3] WELCOME TO SIMOC (starting portal for SIMOC users), <https://ngs.simoc.space/>, (accessed 15.08.21).
- [4] Kai Staats, Project Lead, Iurii Milovanov, Lead Server Developer, Don Boonstra, Educational Developer, A SIMOC Technical Document, ASU, School of Earth & Space Exploration, Interplanetary Initiative Pilot Project
- [5] Álvaro Vicente-Retortillo, Mark t. Lemmon, Germán m. Martínez, Francisco Valero, Luis Vázquez, m<sup>a</sup> Luisa Martín, Seasonal and interannual variability of solar radiation at Spirit, Opportunity and Curiosity landing sites, *Física de la Tierra* 126 Vol. 28 (2016) 111-127
- [6] D.Viúdez-Moreiras, J.Gómez-Elviraa, C.E.Newman, S.Navarroa, M.Marina, J.Torresa, M.de la Torre-Juárez, the MSL team, Gale surface wind characterization based on the Mars Science Laboratory REMS dataset. Part I: Wind retrieval and Gale's wind speeds and directions, *Icarus* Volume 319, February 2019, Pages 909-925
- [7] Amber Thomas homepage for code, <https://github.com/the-pudding/data/tree/master/mars-weather>, (last accessed 15.08.21).
- [8] Atmosphere relative humidity Gale crater,  
[https://www.nasa.gov/mission\\_pages/msl/multimedia/pia16915.html](https://www.nasa.gov/mission_pages/msl/multimedia/pia16915.html), (last accessed 21.08.21).

- [9] Jody M. Luna, Sustainable Land Development Using Permaculture, Research Anthology on Measuring and Achieving Sustainable Development Goals (2022)
- [10] National Aeronautics and Space Administration, HUMAN INTEGRATION DESIGN HANDBOOK, NASA/SP-2010-3407/REV1
- [11] Marc M. Cohen, Testing the Celentano Curve: An Empirical Survey of Predictions for Human Spacecraft Pressurized Volume, SAE TECHNICALPAPER SERIES 2008-01-2027