**Sampler**

This component automatically generates a list of parametric design vectors, called a “design map”, based on user-defined design variable properties. It outputs this design map as a nested list and Grasshopper, which can be used directly or plugged into other DSE tools, and saves it as a .csv file for documentation and interfacing with outside software. Sampler works on double-click, which will both output a Design Map and write a new file to your directory.

**Inputs:**

**Variables (Vars) –** Takes in any number of sliders that are used as design variables for a given project. Sampler automatically reads the bounds of the sliders to set the limits of the design space being explored.

**Number of Samples (N) –** The number of samples to be generated. In the ‘grid’ design mode, rather than automatically truncating the list of samples at the edges or adjusting their spacing, the Design Map contains a list of samples that fully covers the design space in each dimension. This results in a number of samples that is higher than the input N, but is divisible by **Vars** . The user can manually trip this Design Map as needed.

**Type (Type) –** Sets the sampling technique: random, grid, or Latin Hypercube(citation).

**Seed (S)** – Allows the user to refer back to a previously generated Design Map, even if random as involved. If the seed is set to ‘0’, Random or LHC will generate a new Design Map each time it is clicked. Every other integer will reference back to the same Design Map, even if it is randomly generated.

**Filename (F)** – The name of the Design Map file that will be written as a .csv.

**Directory (Dir) –** The location on your computer where the Design Map will be saved. The directory MUST end with a “\” – otherwise, the file will be written one level higher than the intended directory.

**Outputs:**

**Design Map (DM) –** A nested list of design vectors that span the design space being explored, based on the selected sampling technique.

**Capture**

This component is a general iterator, which allows the user to automatically generate many different design options and record an image, the performance, and/or other properties of each design. Capture works on double-click, which will cause the component to cycle through each design option and recording the results.

**Inputs:**

**Variables (Vars) –** Takes in any number of sliders that are used as design variables for a given project. Sampler automatically reads the bounds of the sliders to set the limits of the design space being explored. The sliders should be identical to those used to generate the Design Map.

**Objectives (Obs) –** Reads in a list of the numerical values from performance evaluations generated by the script. Any performance-based measurement (or objective function), generated from either an outside plug-in or a collection of native Grasshopper components, can be recorded. Depending on Mode, Capture can function with either zero, one, or multiple objective functions in this input.

**Design Map (DM) –** A nested list of design vectors to be evaluated, coming either directly from Sampler or from another previously generated list of design options.

**Mode (M) –** Sets the operating mode: screenshots, eval, or both. Screenshot mode records the image of the current Rhino view and saves it as a .png in the directory.

**Properties (P)** – Similar to objectives, reads in a list of numerical secondary design properties to be recorded while Capture is cycling through the different design options. However, these properties are not necessarily objective function evaluations and will be kept separate from the objectives when both are output by Capture.

**.CSV Filename (.csv F)** – The prefix name of the Design Map + Objectives and Properties files that will be written as a .csv to the directory.

**.CSV Directory (.csv Dir) –** The location on your computer where the Design Map + Objectives and Properties files will be saved. The directory MUST end with a “\” – otherwise, the file will be written one level higher than the intended directory.

**Screenshot Filename (SS F)** – The prefix name of the screenshot files that will be saved as a .png in the directory.

**Screenshot Directory (SS Dir) –** The location on your computer where the screenshots will be saved. The directory MUST end with a “\” – otherwise, the file will be written one level higher than the intended directory.

**Outputs:**

**Design Map + Objectives (DM+O) –** A nested list of design vectors and their objective values that span the design space being explored. Standard format for each design is the design vector followed by as many objective values as are recorded.

**Properties (Props) -** A nested list of the secondary design properties recorded while Capture is cycling.

**Reader**

**Inputs:**

**Directory + Filename (Dir + F) -** Takes in the full address on your computer where the desired file is located, including the extension. The format takes in each row and converts it to a nested list, following the standard format for Design Maps. However, this component could read in any file saved in a similar row/column structure.

**Separator (S) –** Sets the character used to separate the data in the .csv. By default it is a space, which is how Design Maps are written, but for other data it could be “,”.

**Outputs:**

**Data (D) -** A nested list of the data contained in the .csv, following the standard format of design variables followed by objective values in the case of DM+O files.

**Writer**

**Inputs:**

**Directory + Filename (Dir + F) -** Takes in the full address on your computer where the file should be written (no extension is necessary).

**Data (D) –** Takes in a nested list of the data that will be converted to a row x column .csv file.

**MOO**

This component automatically generates a list of parametric design vectors, called a “design map”, based on user-defined design variable properties. It outputs this design map as a nested list and Grasshopper, which can be used directly or plugged into other DSE tools, and saves it as a .csv file for documentation and interfacing with outside software. Sampler works on double-click, which will both output a Design Map and write a new file to your directory.

**Inputs:**

**Variables (Vars) –** Takes in any number of sliders that are used as design variables for a given project. Sampler automatically reads the bounds of the sliders to set the limits of the design space being explored.

**Number of Samples (N) –** The number of samples to be generated. In the ‘grid’ design mode, rather than automatically truncating the list of samples at the edges or adjusting their spacing, the Design Map contains a list of samples that fully covers the design space in each dimension. This results in a number of samples that is higher than the input N, but is divisible by **Vars** . The user can manually trip this Design Map as needed.

**Type (Type) –** Sets the sampling technique: random, grid, or Latin Hypercube(citation).

**Seed (S)** – Allows the user to refer back to a previously generated Design Map, even if random as involved. If the seed is set to ‘0’, Random or LHC will generate a new Design Map each time it is clicked. Every other integer will reference back to the same Design Map, even if it is randomly generated.

**Filename (F)** – The name of the Design Map file that will be written as a .csv.

**Directory (Dir) –** The location on your computer where the Design Map will be saved. The directory MUST end with a “\” – otherwise, the file will be written one level higher than the intended directory.

**Outputs:**

**Design Map (DM) –** A nested list of design vectors that span the design space being explored, based on the selected sampling technique.