Organize It!
User Manual

Enabling users to create custom dream closets that optimize their space since 2024.

Rajvi Patil, Vita Sipayung, and Julia Beitel

USER MANUAL INCLUDES:

- HIGH LEVEL OVERVIEW
- OUR PLUG-INS (6 COMPONENTS) EXPLAINED
- QUICKSTART GUIDE (Parts 1, 2, and 3)
- DETAILED FUNCTIONALITY
- ADVANCED USAGE & EXAMPLE
- RESULTS FROM USING OUR TOOL
- KNOWN BUGS & LIMITATIONS
- DOWNLOADING CUSTOM TEXTURES
- CUSTOM MODEL COLORS
- WHAT TO DO NEXT (Parts 1 and 2)

HIGH LEVEL OVERVIEW

Organize It! is a DIY tool that enables users create customisable closets and helps users maximize space and build their dream closet. (Note: You do not need licensing access for any of our tools or those used in our example file.)

Organize It is divided into 7 parts visually in one Grasshopper file (not a packaged component), where the user first needs to input the overall dimension of their closet, complete an inventory of their items, and then change the colour and texture with 3D rendering capabilities. Please utilize our example file when first getting started and refer to the quickstart guide for tips.

The user is advised to use the perspective window and set the 'rendered' viewport viewing option to get the most out of the visual appearance of the cupboard. Please align the perspective such that the x-z axis is the base plane. And enjoy playing around with the inventory number, output and texture to finalize from various options!

We, the Organize It! Team, hope that you can make the most out of this tool.

Happy organizing!

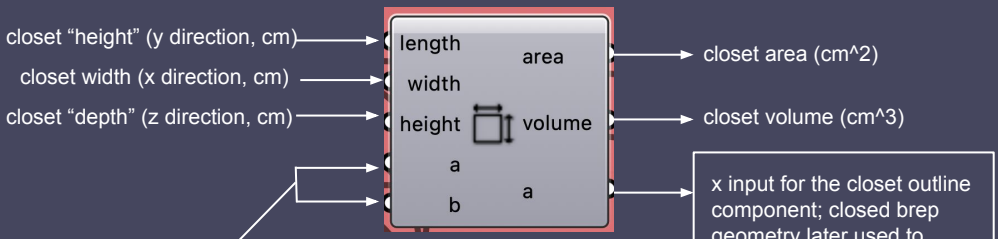
OUR PLUG-INS EXPLAINED

CLOSET: CLOSET DIMENSIONS

The closet dimensions component allows users to input the length, width, and depth (unit: centimeters) of their closet space **using number sliders**, outputting the area and volume of that closet space for visualization, as well as closed brep geometry used to later generate the 2D outline of the user's closet space.

These component input values may be initially confusing to the user, as the closet render is located in the x-z plane, whereas we are used to using a x-y plane to measure. Please refer to the documentation below to reference when you are measuring and inputting your closet space values.

Please place number sliders into each input of the component and note that it may be helpful to visualize the area and volume outputs by connecting panels to these values (as seen in our example file) so that you may reference these values later on.



We initially placed these values in our code as placeholders for the number of shelves included in the closet render (a) and the base height (b) of each shelf. Please place a number slider to each input a and b. We recommend placing a value of 10 (as referenced in our example file), but these inputs can be any whole number.

x input for the closet outline component; closed brep geometry later used to create a 2D outline of the user's closet space

(refer to the closet outline component documentation as shown below here)

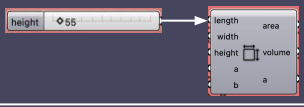
OUR PLUG-INS EXPLAINED

CLOSET: CLOSET AREA AND VOLUME

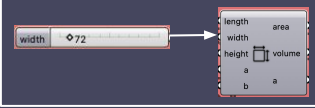
The closet area and volume component is fairly simple as the user should be reusing the previously defined values for the closet height, width, and length in the closet dimensions component.

This component generates the area and volume (again) in order to be repurposed for the inventory comparison component, with the ultimate goal of providing the user with suggestions for decluttering their closet (but this is just the first step in doing so ☺).

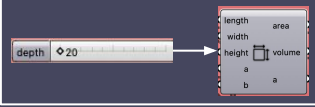
Connect the number slider previously used to define your closet height in the closet dimensions component. This number slider should have two connections now.

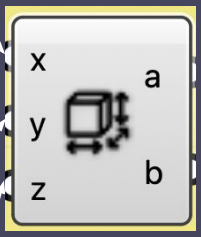


Connect the number slider previously used to define your closet width in the closet dimensions component. This number slider should have two connections now.



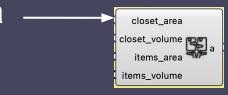
Connect the number slider previously used to define your closet depth in the closet dimensions component. This number slider should have two connections now.





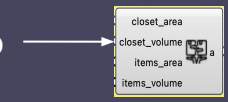
This output generates the closet area (cm^2).

This output (a) is the input of closet_area in the inventory comparison component.



This output generates the closet volume (cm^3).

This output (b) is the input of closet_volume in the inventory comparison component.



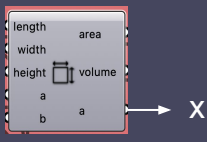
OUR PLUG-INS EXPLAINED

CLOSET: CLOSET OUTLINE

The closet outline component creates a 2D outline of the user's closet area. This outline (polyline geometry) is used to represent the closet's total dimensions in the 2D plane so that later it can be generated into a 3D model.

In the sake of full transparency we wanted to draw attention to the fact that in our example file, you will see the output of this component (a) connected to a grasshopper move component. This was used by the Organize It! Team while working on developing this plug-in. Please note that the move component is not necessary if you are generating the closet model generating tools yourself and without the aid of the example file.

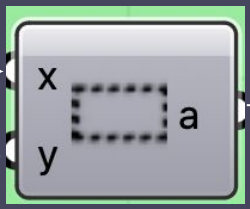
The a output of the closet dimensions component.



This is an empty input parameter.

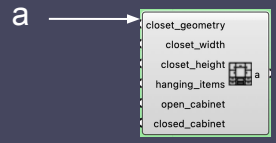
Please leave this input empty and do not connect any values to it.

Please see the bugs and limitations section for further explanation.



This output generates the polyline geometry that represents the closet area in a 2D plane.

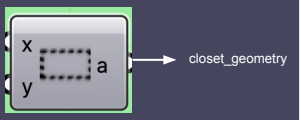
This output (a) should be connected to the closet_geometry input of the closet 3D component.



OUR PLUG-INS EXPLAINED

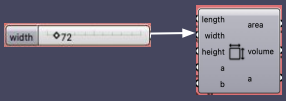
CLOSET: CLOSET 3D

The a output of the closet area and volume component.

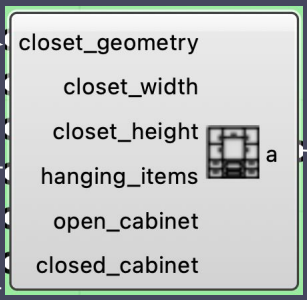


The closet 3D component creates a 2D model of the user's closet design (with polyline geometry) that includes hanging racks and open and closed cabinet spaces.

Connect the number slider previously used to define your closet width in the closet dimensions component. This number slider should have three connections now.



Connect the number slider previously used to define your closet height in the closet dimensions component. This number slider should have three connections now.



This output represents the polyline curves that make up the 2D plane of the total closet geometry with shelving and hanging areas included.

Next steps: (2D -> 3D)
As seen in the example file, connect this output (a) to an extrusion component with the depth value (in unit z) equal to the depth of your closet.

Connect the number sliders previously used to define your closet inventory. For each item type, connect that number slider to an addition component in grasshopper (one for items you store in an open cabinet, another for items you store in a closet cabinet, and lastly one for the items you hang in your closet).

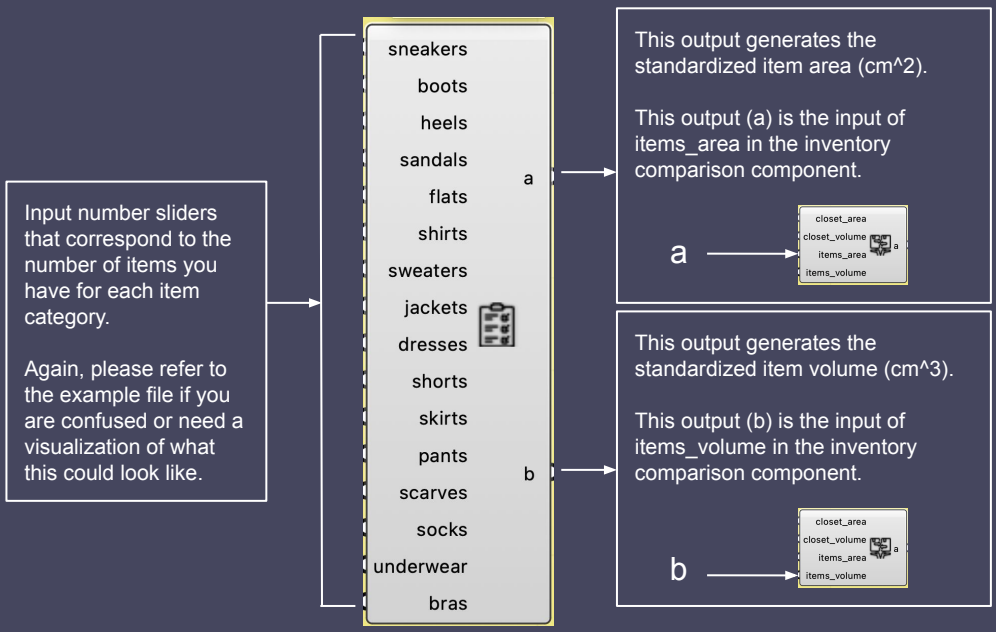
Then, connect each addition component that corresponds to the inputs in the closet 3D component. For example, the open cabinet items will connect to the open_cabinet input for this component. If you are confused, please refer to our example file to see how this should look in your own grasshopper file. Our example file standardizes what items may fit in the different areas of your closet, but feel free to add item types to the locations where you see them fitting best.

OUR PLUG-INS EXPLAINED

INVENTORY: CLOSET ITEMS

The closet items component is where the user will be able to input number sliders for all of the items in their closet. This component is used to generate the total area and volume of the user's closet inventory so that we may provide the user with suggestions for decluttering their closet (by comparing the closet area & volume to the inventory area and volume (which is standardized for simplicity).

Please refer to the example file for a visualization of how this could look in your grasshopper file.

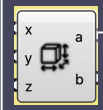


OUR PLUG-INS EXPLAINED

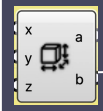
INVENTORY: INVENTORY COMPARISON

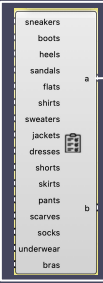
The inventory comparison component is used to compare users inventory of closet items to the dimensions of their closet space. We provide decluttering suggestions based on different outcomes: the closet dimensions are equal to the item dimensions (providing a perfect fit), the closet dimensions are less than the item dimensions (meaning the user is suggested to declutter their item inventory), and the closet dimensions are greater than the item dimensions (meaning the user will be left with extra space for future use).

Connect the a output of the closet area & volume component. This represents the value for the closet area.

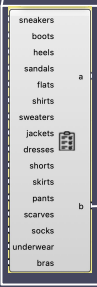


Connect the b output of the closet area & volume component. This represents the value for the closet volume.

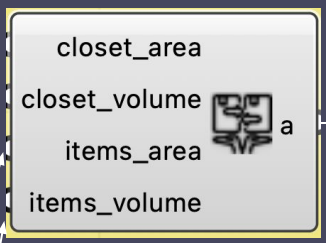




Connect the a output of the closet items component. This represents the area of the user's item inventory.



Connect the b output of the closet items component. This represents the volume of the user's item inventory.



This output provides the user with a text output of suggestions for decluttering.

Please connect a panel to this output (a) so that you may easily see the suggestions we have provided.

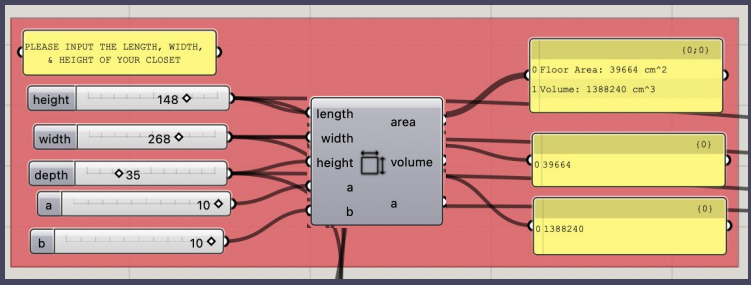
Please also note that these are merely suggestions and your items may fit in your closet even if the decluttering suggestion tells you otherwise.

QUICKSTART (part 1)!

USING THE EXAMPLE FILE

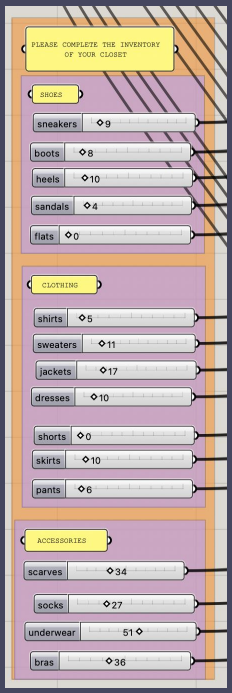
1 Please input the height, width, and depth of your closet (RED group).

Height →
Width →
Depth →



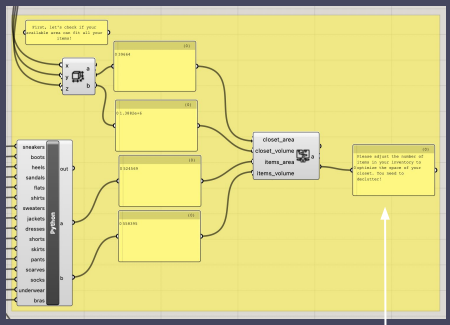
2 Complete the inventory of the closet, that is you just need to adjust the number slider for each item (ORANGE group).

Inventory is divided into 3 main categories shoes, clothing and accessories and multiple sub-categories (BLUE group).



QUICKSTART (part 2)!
USING THE EXAMPLE FILE

3 Then, let's check if your available area can fit all your items (YELLOW group)!



If not, you should most likely declutter your closet by removing some items so that you can maximize the space. See if your space fits by checking the leftmost panel in this group.

4 Download textures in Rhino (to know more look at the downloading custom textures section).

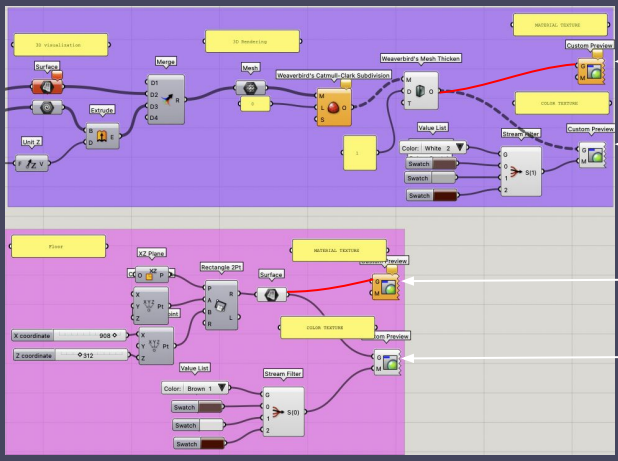
Example textures include but are not limited to:

QUICKSTART (part 3)!

USING THE EXAMPLE FILE

5 Zoom into the 3D rendering components (PURPLE and PINK groups). Note that the purple group is for the closet texture and the pink group is for the floor texture.

- Here, you can right click on the materials in the custom preview and select material or use the colour toggle to switch between different colours.

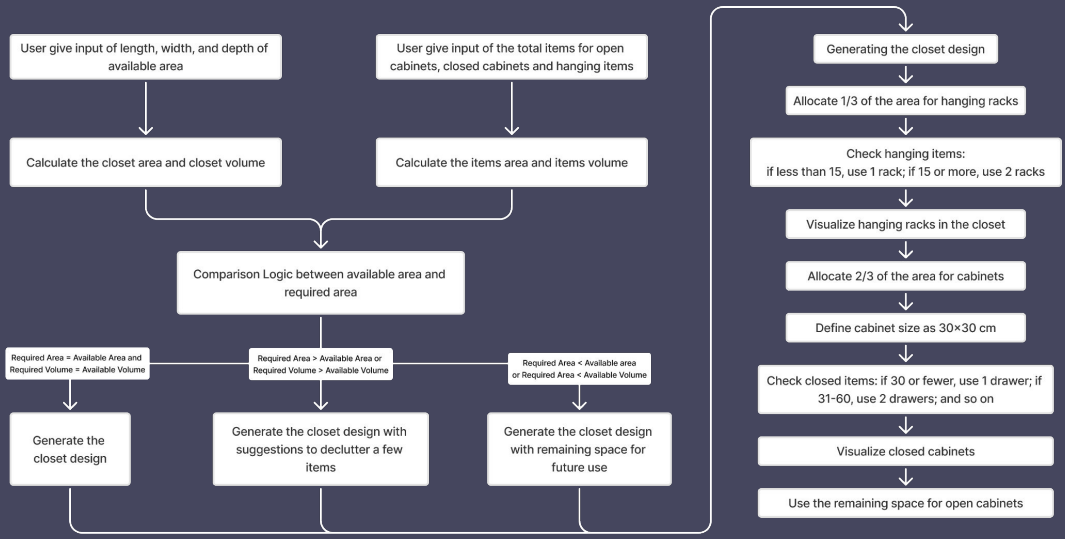


★ **Do not forget** to connect the colour custom preview to the output geometry, that is connect O→G for the closet and the Srf component → G for the floor.

- In the given screenshots, you can see that there is a connector for both pink and purple groups for the custom color texture. If you want to choose the material texture, you must connect the below preview component (notated with a red line above).

DETAILED FUNCTIONALITY (part 1)

OUR WORKFLOW EXPLAINED



1st Step: Closet vs. Items Comparison

As a user, you will provide the length, width, and depth of your available area, along with the total number of items you have.
Our tool will calculate the following based on your input:

- Closet area and closet volume: Total available space (cm²/cm³).
- Items area and items volume: Required space for your items (cm²/cm³).

You will receive a message either confirming the closet design, suggesting decluttering, or noting any extra space available.

2nd Step: Closet Design

Provide the closet geometry (boundary polyline), width, height, and item counts for hanging, open, and closed storage. Our tool will:

- Validate the geometry as a closed rectangle.
- Allocate space:
  - **Hanging Racks:** Use 1/3 width; add a second rack if items > 15.
  - **Cabinets:** Use 2/3 width; prioritize closed cabinets (30 items per drawer) and assign leftover space to open cabinets.

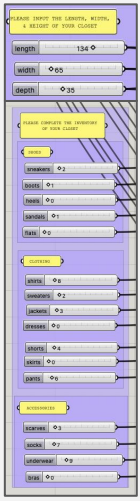
You will receive a visualized design with geometries for racks and cabinets, optimized for your items.

RESULTS FROM USING OUR TOOL

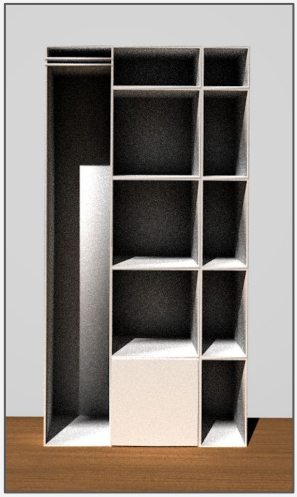
User 1: Rick M.

- Just moved into a tiny apartment; living a minimalist lifestyle
- Saving $$ to buy a car

Inputs:



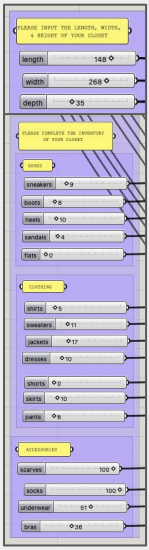
Output:



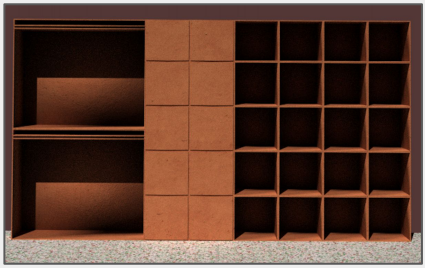
User 2: Claire Z.

- Lives in a luxury apartment unit, but struggles to organize her things
- Wants to save $$ on closet design so she can buy more scarves

Inputs:



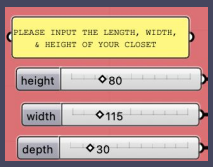
Outputs:



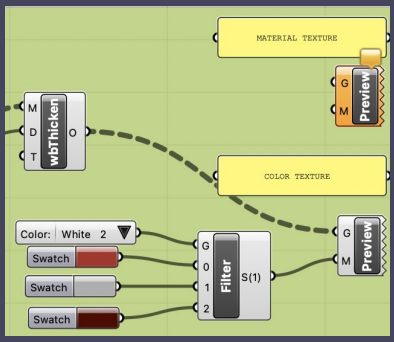
ADVANCED USAGE & EXAMPLE

To view our example file, you do not need to have the custom texture library downloaded as we have used only the color swatch texture option in this render. Below you will find the example closet dimensions and inputs as well as the chosen closet and floor textures:

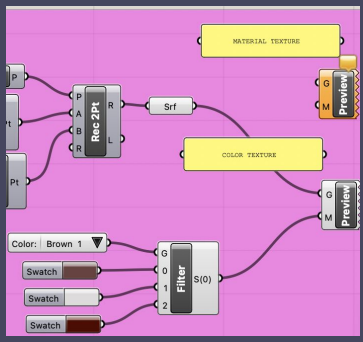
Example dimensions:



Example closet texture:

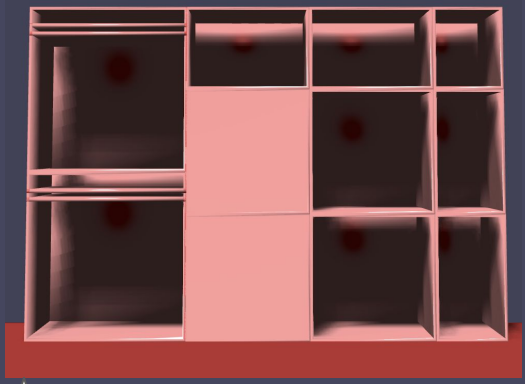


Example floor texture:



Example inventory:

Example closet model output:



Given the example inputs, here is the output model closet.

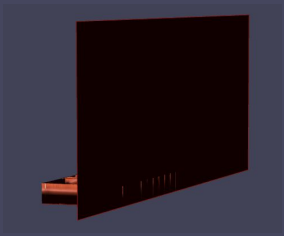
The two hanging racks account for the large number of hanging items and the closed cabinet spaces account for the large number of items to be stored in closet cabinet drawers.

KNOWN BUGS & LIMITATIONS

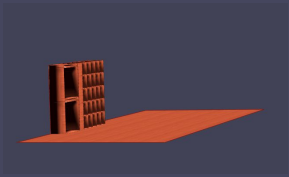
1. User has to manually set the texture and material thickness for the cupboard (See downloading custom textures page).

2. The thickness tool used is a component from the grasshopper plug-in weaverbird and is not included in the Organize It! Software. This allows for the output model to have greater thickness so that the edges of the closet are smoothes out.

3. The cupboard and floor orientation is along x-z plane, so it can be challenging to navigate through the viewport.



Here is an angled view of the 3D render to show how the floor output may cover the closet model from non-optimal perspectives.



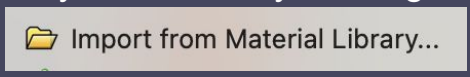
Once rotated, the model is easily viewed and able to be edited to the user's desire!

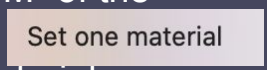
4. In the closet dimensions component, the a and b inputs are parameters that must be placed as number sliders, but have no end value in our code. We found this as a unnecessary addition after completing the final version of our code, initially using them as placeholder values for the number of shelves in the output closet and the base height of each self.
    a. The same can be said for the closet area & volume component: This specific component is ultimately redundant, as users can input the closet area and volume (in the inventory comparison component) with the area and volume outputs of the closet dimension component. We decided to leave this component in our plug-in and as an addition to our example file as it allows for simple area and volume generation from number slider inputs, which is not currently available in Grasshopper.

5. In the closet outline component the y input is an empty parameter that should not have anything connected to it. This should've been removed from the component before we packaged it. We apologize for any confusion or inconvenience.

6. The closet inventory component is limited by the item categories we have set. If the user has other items in their closet that are not accounted for in that component's inputs, they are not included in the final model of their closet (the items are also standardized in size for the closet area and item area comparison for decluttering suggestions, which limits users if they have differently sized items in their closet inventory).

DOWNLOADING CUSTOM TEXTURES

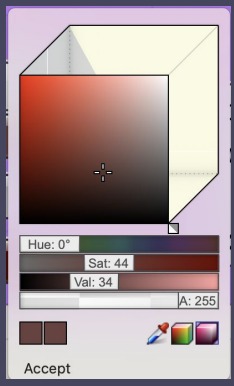
1. To be able to set the texture, first download the Rhino textures, by typing in the command box ‘Download library textures’ in your rhino window.
2. Once downloaded, head to the materials panel by clicking the settings icon  on the right side of your rhino window.
3. You will know you have successfully reached the materials panel when you see this icon →  ghted at the top of the left side of your rhino window.
4. Add new materials for your model by clicking this plus icon and selecting:  

5. By selecting “import from material library..” you will be brought to the downloaded material texture library where you can select from a number of different texture options. Once you have selected a material texture, it will appear in your materials panel (within rhino).
6. From here, you can toggle between any selected material texture from inside the Organize It! Grasshopper file by right clicking on the “M” of the material texture component and clicking:  and then selecting your chosen saved material.

 Please save a Rhino file along with your grasshopper file to retain the saved textures for multiple uses.

CUSTOM MODEL COLORS

1. To be able to set different colors to the model, users can modify the 3 color options given in the (PURPLE and PINK groups) by clicking on the color swatch they want to change.
2. After clicking on the color swatch, users will see a pop-up window that looks like this

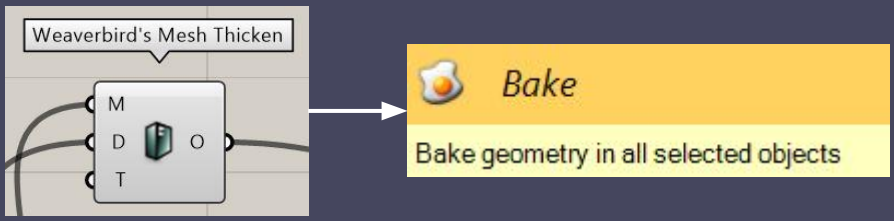


3. Here, users can modify the color of their model to any option in the color spectrum.
4. After selecting the desired color, users should click this “accept” button and will be able to cover their model closet with their new custom color.

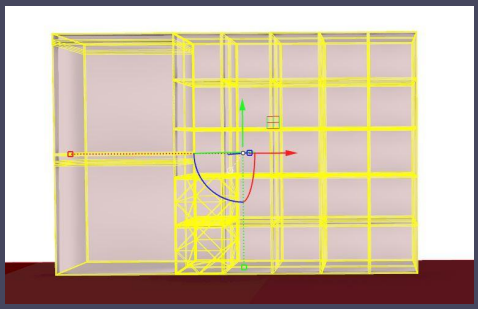
★ If confused on how to apply colors to the model, please refer to the quickstart guide (part 3) for instructions.

WHAT TO DO NEXT (part 1):

Once you have decided the final version of your cupboard design, you can bake the outcome, to do so right click on Weaverbirds's Mesh Thicken (Plug-in found on Food4Rhino).



Following the above steps will create an object in rhino which is directly editable, it will look like the following:

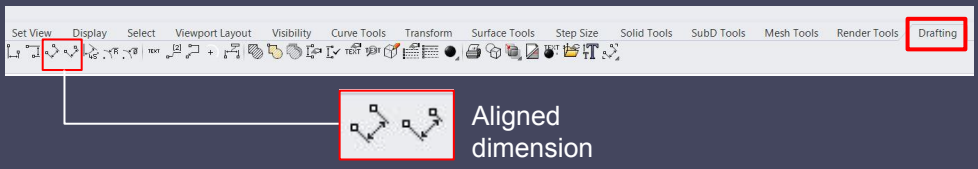


The object has been selected in rhino modelling space that is the reason why you can see a yellow outline. The red, green and blue arrows that you see are the representation of the axes of the 3D modeling space, thus selecting and dragging them will help you move the cupboard.

Now that you have a baked version you can play around with it and measure the dimensions to actually start building the cupboard.

WHAT TO DO NEXT (part 2):

To measure dimensions, and to have a layout of cupboard , you can use tools from drafting panel in the rhino interface. For this tool, the output dimensions are in centimeters(cm) set by default but you can alway change units.



Getting familiar with rhino toolbar will help you navigate easily without getting lost in a wide range of tool set, so let's keep things focused on getting the cupboard dimensions.

Select Drafting→ Aligned dimension

Pick 2 end points of the cupboard shelves to get a clear dimension.

Now that you have your desired closet with dimensions, the Organize It! Team wishes you luck in making your closet dreams come true.