

Originally published on <https://ultimaker.com/learn/education-blog> (Fall 2016)

Preparing to Teach with 3D Printing

Kristen R. Schreck
Saint Xavier University
Chicago, IL

After I successfully printed my very first 3D objects with the Ultimaker 2+ (a.k.a. the Ultibot and Ultibot, Jr.) and learned the mechanics of the process, I felt that I really needed to gain a more global understanding of 3D printing before the semester started – so that my students would, too.



(Credits: Ultibot and Coffin's Cube - courtesy of Ultimaker 2+ SD card; Wedge 2 (<http://www.thingiverse.com/thing:898069>) - by dennedesigns (<http://www.thingiverse.com/dennedesigns/about>); Pokemon Harmony badge (<http://www.thingiverse.com/thing:1670621/#files>) - by ClassyGoat (<http://www.thingiverse.com/ClassyGoat/about>); Lego block (<https://www.youmagine.com/designs/lego-brick-2x4>) - by Cajus Pruin (<https://www.youmagine.com/cajus-pruin/designs>); all other designs shown are by @mongemath (<https://twitter.com/mongemath>)).

In August, during the week the Ultimaker 2+ cast its long, beautiful shadow on my life, I participated in the workshop "Introduction to 3D Printing for Educators" facilitated by Megan Walsh, 4-H Youth Development Educator, at the University of Illinois Extension Campus (<http://web.extension.illinois.edu/gkw/>) in suburban Chicago. Megan did a wonderful job of introducing 3D printing and its uses in K – 12 and higher education. It was just what I needed to help set the scene for my Multivariable Calculus students this fall semester - for most, it will be their first experience with 3D printing.

Here's my framework for 3D printing based on what I have learned thus far, some resources I have come across in my own investigations, and a few of the new 3D prints I've made using these resources - all in preparation for getting my students started on their own 3D printing journey.

A Framework for 3D Printing

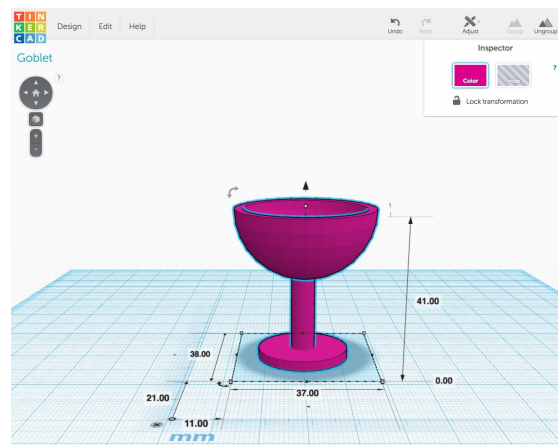
I quickly discovered that people in the 3D printing community love to share ideas and designs! The designs can be created from scratch or acquired through scanning or downloading.

Here are some of **ways I've learned to acquire designs**.

(1) Using **modeling** programs:

- Tinkercad (<https://www.tinkercad.com/>) seemed like a logical entry level place for me to start in creating my first original model since it is easy to use. A well-known Calculus II project came to mind, designing a goblet as a surface of revolution. In my Calculus II course we used the mathematics software, Maple (<http://www.maplesoft.com/>), to form a piecewise function and revolve it about an axis (vertical or horizontal) which generated a unique goblet form (to be rotated upright, as necessary). In that project, other specific criteria was given requiring this fancy cup to hold a specific volume of liquid - which made the project even more of a challenge.

My goblet model in Tinkercad could be created purely with geometric shapes! It was a new twist on an age-old problem.

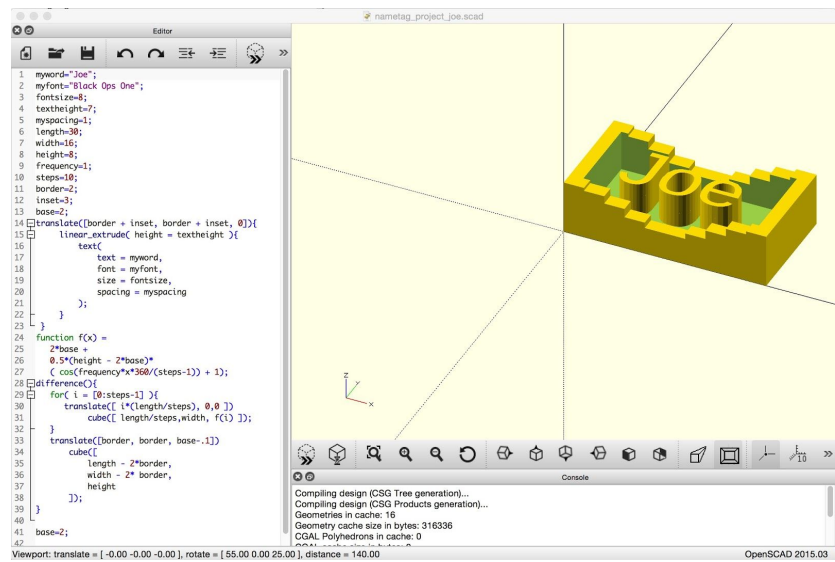
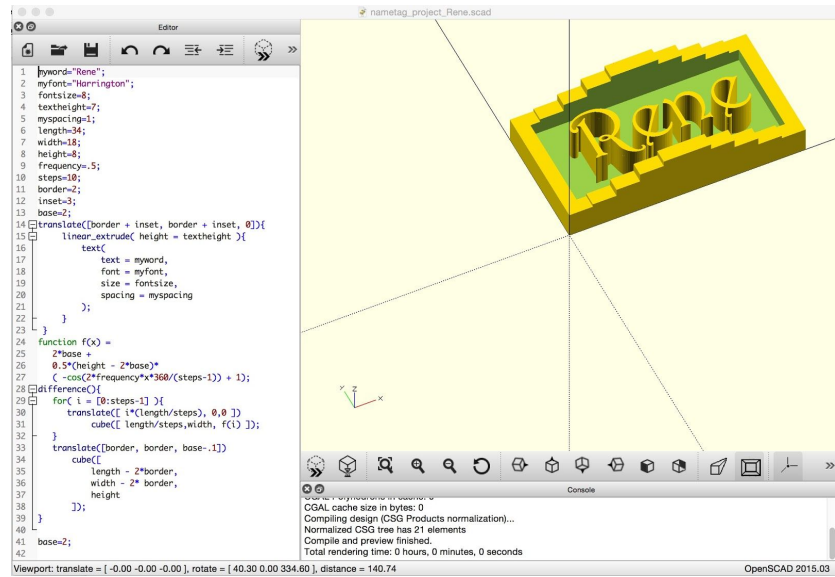


A few images of the goblet - hot off the 3D printer - are given below. The first photo shows the raft and a bit of roughness around the base of the goblet bowl (I did not use any supports). Then we see the final product (I think it's a beauty despite some rough edges).



- Another of my exploits with modeling programs involved OpenSCAD (<http://www.openscad.org/>). I was searching for nameplates to create for my students to give them during the first week of class. I came across a tutorial and editable program that I downloaded from Thingiverse (<http://www.thingiverse.com/>). The program is called Nameplate Generator with OpenSCAD by MakerBot Learning (<http://www.thingiverse.com/thing:876282>). This program forms wavy nameplates using transformations of trigonometric functions. (There will be more information about other places where to **download** files below.)

OpenSCAD's command-line format reminded me a lot of how procedures are written in Maple, so that felt familiar. I modified the code in OpenSCAD to personalize each nameplate by varying the font type and size, the type of trigonometric function, the amplitude and period of the function, and the length and height depending on the number of letters in each name. Examples of my modified OpenSCAD nameplate files are:



I'll have to maintain some suspense here - in my next post, which will be entirely about my students and their first projects, I will reveal the actual 3D prints of their nameplates.

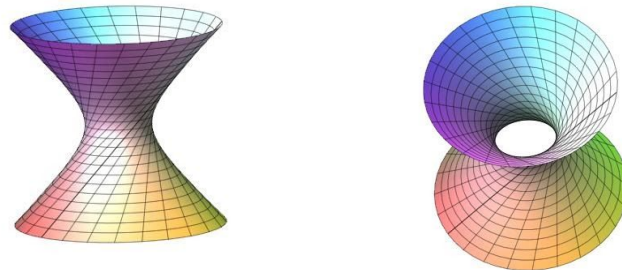
- These are just two of the modeling programs I have explored up to this point. The Savvy 3D Educator's Superlist on Ultimaker's Education website (<https://ultimaker.com/en/resources/education/the-savvy-3d-educator-s-superlist>) provides a great comprehensive list of popular 3D modeling software applications (<https://ultimaker.com/en/resources/21876-software-choices>). I look forward to investigating and reporting on some of the more advanced applications throughout this semester.

(2) Using **mathematics software**:

During my first week with the Ultimaker 2+, I was determined to figure out how to print 3D models that I had previously created in my own work using the mathematics software, Maple. I chose to print a hyperboloid of revolution. The Maple code for plotting this surface and the code required to create the STL file are shown below. Maple is directed to put the STL file in a specific folder of the current directory on my computer - so it's easy to find. As a side note, the number given in the bottom set of Maple code is the byte count of the STL file that is generated.

```
> restart :
with(plots) :
with(plottools) :
with(VectorCalculus) :

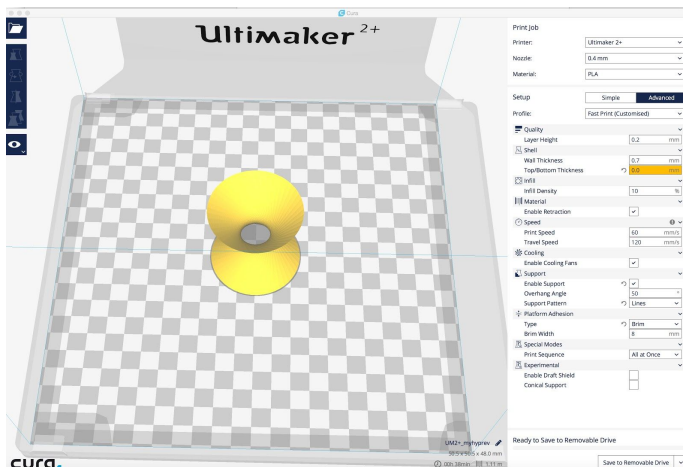
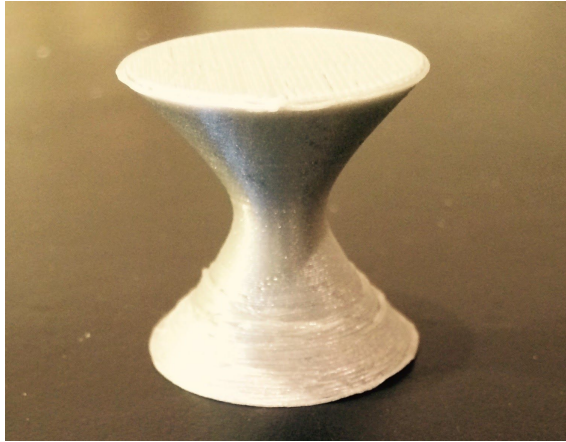
=
> hyperboloid := <cos(u) - v·sin(u), sin(u) + v·cos(u), v> :
plot3d(hyperboloid, u = 0 .. 2·Pi, v = -3 .. 3,
style = surfacewireframe, lightmodel = light4, scaling = constrained,
axes = none);
```

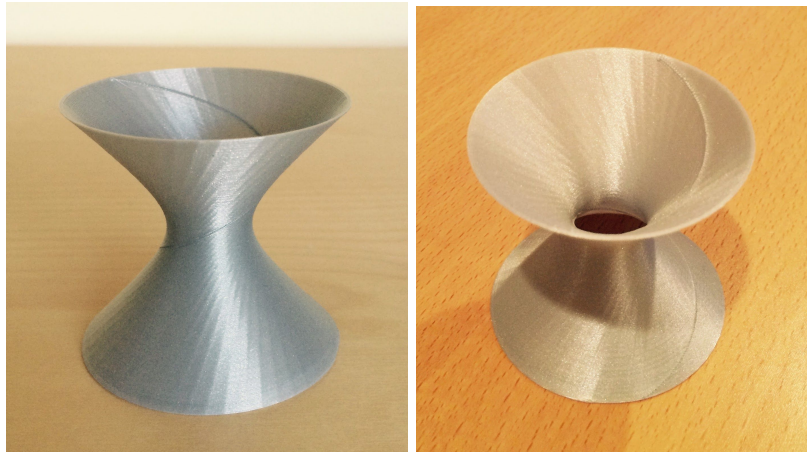


```
> hyperboloid := plot3d(<cos(u) - v·sin(u), sin(u) + v·cos(u), v>, u = 0 .. 2·Pi, v = -3 .. 3,
style = surfacewireframe, lightmodel = light4, scaling = constrained,
axes = none);
myfile := FileTools:-JoinPath([currentdir(), "myhyprev.stl"]);
plottools[exportplot](myfile, hyperboloid);
hyperboloid := PLOT3D(...)
myfile := "/Users/Kristen/Documents/3D with Maple 2016/myhyprev.stl"
230484
```

In Cura, although the hyperboloid of revolution is shown with the hole through its center, the final object was printed as a solid. This is why - my initial settings were Setup: Simple, Infill: Hollow, with a raft and no supports. When the layers were computed, the hyperboloid was still shown as a solid with a top and bottom layer.

The second time I printed the hyperboloid I figured out that I needed to adjust the Advanced Settings in Cura as follows: Top/Bottom Thickness = 0 mm with a Layer Height = 0.15 mm and Infill Density = 10%. This produced the hole through the surface and the hyperboloid looked just right!





An interesting development occurred when the brim started printing for the hyperboloid, a slight ridge of filament formed in a line across the brim. I did not think it would affect the surface. However, a trail was left on the inside of the surface - as if a bug crawled on it. I think I can work this into one of our class discussions on curvature of surfaces. It would begin something like . . . “Imagine yourself as a bug inching along the inside of a hyperboloid of revolution. . .”

(3) **3D scanning** applications:

I recently took photos of a small metal bust of Abraham Lincoln (my husband’s idea - he’s a Poli Sci guy). I was excited to scan it using 123D Catch (<http://www.123dapp.com/catch>) to see how the program worked.



Here are a few suggestions when using 123D Catch:

- Take many, many photos from several different angles and at varying distances from the object. (I took close to 100 and then narrowed it down to my best 70 photos.)
- Metal or shiny objects may be more difficult to capture. Objects with transparent parts will result in a failed capture (ask me, I know).
- It took over an hour to translate and finalize the photos.
- Patience is necessary in completing these tasks.
- All in all, it was a productive and educational first try.

The following Twitter link (<https://twitter.com/mongemath>) illustrates the results using this 3D scanning application.

NOTE TO LIZ: I wasn't sure if I should use the Twitter URL or the embedded link - so I have included both here. Please only the one you think would be best to link to for this post. Thank you!

(<https://twitter.com/mongemath/status/779518883589791745>)

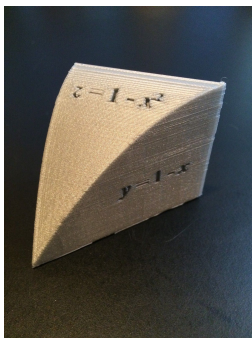
```
<blockquote class="twitter-tweet" data-lang="en"><p lang="en" dir="ltr">First try using 123D Catch - Honest Abe <a href="c"></a></p>&mdash; Kristen Schreck (@mongemath) <a href="https://twitter.com/mongemath/status/779518883589791745">September 24, 2016</a></blockquote><script async src="//platform.twitter.com/widgets.js" charset="utf-8"></script>
```

The wealth of sharing taking place in the 3D printing community is amazing and quite refreshing! In lieu of composing your own designs and models using modeling programs, mathematical software, or 3D scanning applications, you can readily download existing models artfully created by someone else.

(4) **Downloading** designs:

If you are perusing the web for designs to download there are many options such as: YouMagine (<https://www.youmagine.com/>), Thingiverse, GrabCAD (<https://grabcad.com/>), to name a few. Note to my students: It is important to credit the person who created the design and to be aware of any restrictions or licenses associated with the design.

I searched YouMagine and Thingiverse for Calculus models. I found a cool wedge model (<http://www.thingiverse.com/thing:898069>) to download from dennedesigns (<http://www.thingiverse.com/dennedesigns/about>). Imprinting the formula on each surface is really neat! Here's my 3D print - a really useful manipulative to bring to class:



3D printing this model has inspired me to develop my own designs for the problems we are working on in class related to the intersection of surfaces.

The Ultimaker Education website provides a wonderful collection of Where to Find Models (<https://ultimaker.com/en/resources/21990-where-to-find-models>) on the Internet.

(5) **3D Printer Hubs:**

If you've designed your own models and don't have access to a 3D printer, or if you would like to have your prototypes professionally printed using an industrial type 3D printer, there are printer hubs (services) to which you can upload your file and they will do the printing for you (for a fee). Examples of these types of 3D printing services are: Shapeways (<http://www.shapeways.com/>), 3D Hubs (<https://www.3dhubs.com/>), among many others.

Also, many local Libraries now have 3D printers. Libraries will often include free 3D printer training and only charge a nominal fee for printing (cents/ounce).

(6) **The History of 3D Printing:**

A framework for 3D printing would not be complete without delving a bit into its history.

In 1983, Charles "Chuck" Hull (<http://www.3dsystems.com/30-years-innovation>) invented STL which enables the data in CAD software to be translated to 3D printers. His first 3D printing machine was known as the "Rapid Prototyping" machine called the SLA-1. In 1986, Mr. Hull started his company 3D Systems (<http://www.3dsystems.com/>). In 2014, Mr. Hull was inducted into the National Inventors Hall of Fame (<http://invent.org/inductees/hull-charles/>). He is in the company of Thomas Edison, Wilbur Wright, Henry Ford, and Steve Jobs! The patents for the first 3D printers expired years ago which allowed for booming growth and open sourced technology to prevail in the industry - a true 3D printing revolution was born.

Reference: <https://3dprint.com/72171/first-3d-printer-chuck-hull/>

Whew! I think I'm finally getting a handle on my life with 3D printing. Here's to a fabulous 3D print-filled semester in Multivariable Calculus . . . let our excellent adventure begin! (I can't wait to introduce you to my students and their first 3D prints in my next post.)