

CSC293 CS Improves 3D Printed Manufacturing Introduction

Sreepathi Pai

Aug 30, 2023

URCS

Outline

Software Development Tools

3D Printed Manufacturing

Technology

Administrivia

Outline

Software Development Tools

3D Printed Manufacturing

Technology

Administrivia

Basic Tools

Developing software can be a very simple process with limited tools

- Input: Source Code (typed is using an *text editor*)
- Output: Binary (generated using a *compiler*)
- Debugger

Sophistication

- Integrated Development Environment (IDEs)
 - Frontend for editor + compiler + debugger
- Version Control
 - Track changes
 - Prototype features
 - Coordinate development
- Issue Tracker
 - Report and track bugs
- Continuous Integration
 - Test software after every change
- Documentation Generators

The scale of software systems

- Google, around 2 billion lines of code
- Windows Kernel, around 50 million lines of code
- Linux, around 28 million lines of code
 - 4,000+ developers in 2020
 - 1,730 organizations since 2007

Google Is 2 Billion Lines of Code—And It's All in One Place

Linux in 2020

Linux Kernel History Report 2020

Next generation?

- Code writing assistance
 - LLMs
 - Other techniques

Outline

Software Development Tools

3D Printed Manufacturing

Technology

Administrivia

3D Printing

- Sometimes called *additive manufacturing*, the creation of physical objects by laying down materials in layers
 - Fused Deposition Modeling (FDM)
 - Stereolithography (SLA)
 - Selective Laser Sintering (SLS)
 - Multi Jet Fusion (MJF)

3D Modeling (Solid Modeling)

- Create a digital representation of a 3D object
- One method: Constructive Solid Geometry (CSG)
 - Primitive objects (Lines, Cubes, Spheres, etc.)
 - Boolean operations (Union, Subtraction, Intersection, etc.)
 - OpenSCAD

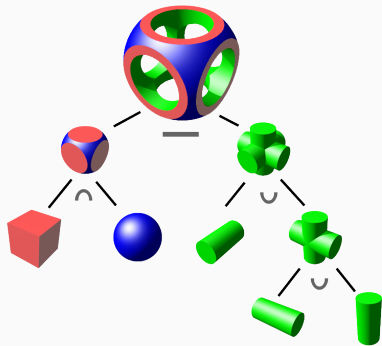


Figure by User:Zottie - Own work, CC BY-SA 3.0,
<https://commons.wikimedia.org/w/index.php?curid=263170>

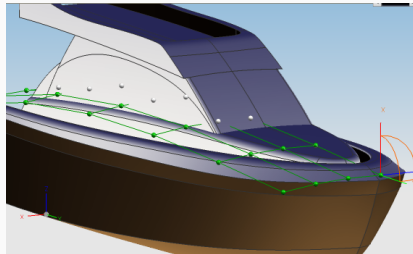
B-reps

- Another method: Boundary Representations (B-reps)
 - Faces, Edges, Vertices
 - Boolean operations + Others
 - Most Computer Aided Design (CAD) packages
 - Check out cadquery, a Python 3D modelling library
- Many others!

The history of solid modeling has strong Rochester connections, check out the Production Automation Project.

Figure by Freeformer, CC BY-SA 3.0,

<https://commons.wikimedia.org/w/index.php?curid=3854978>



Model to Mesh

- A 3D model represents a continuous object
- A *mesh* is a discretization of the object into (usually) triangles
- Common file formats: STL, 3MF
- Most CAD tools that are used to design can also produce meshes

STL-file

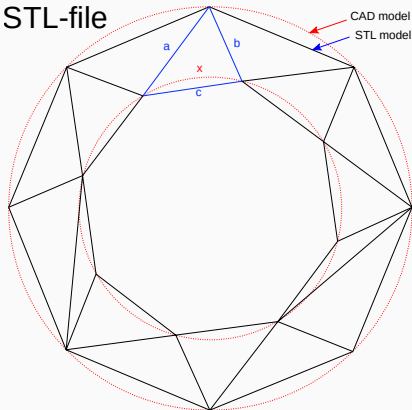


Figure by User:LaurensvanLieshout - File:STL-file.jpg, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=34722631>

Mesh to G-Code

- A mesh needs to be “sliced” into layers, and commands to move the print head for each layer need to be generated
 - The program is called a slicer
- Akin to the compilation step for programs
- End result is G(eometry)-Code, a set of commands originally intended for robots
 - Also used by automated CNC machines

G-code to Printed Object

- A printer "firmware" then converts G-code commands to motion plans
- Then, generates actual signals to drive motors, heat/cool plastic, etc.
- Popular open source firmware: Marlin, Klipper

3D Printed Manufacturing

- The widespread availability of 3D printers allows the creation of physical objects that can be completely defined digitally
 - “Download an object”
- But usually also incorporate other off-the-shelf components
 - Fasteners (nuts, bolts, gears, etc.)
 - Extrusions
 - Electronics

Examples

- FLATBURN: the open source city scanner
 - <https://github.com/MIT-Senseable-City-Lab/OSCS>
- Gondola PlottyBot
- PUMA Opensource Research-Grade Microscope
 - <https://github.com/TadPath/PUMA>
- Stealthburner 3D printer head
 - <https://github.com/VoronDesign/Voron-Stealthburner/tree/main/STLs>
- Voron 3D printers

Artifacts

- CAD files
- STL files
- Bill of Materials (BOM)
- Assembly Manuals

Goals for the Projects

To develop technologies that improve the process of creating these 3D printing manufactured objects.

Outline

Software Development Tools

3D Printed Manufacturing

Technology

Administrivia

Science, Engineering, Technology?

- Science: body of knowledge
- Engineering: "application" of knowledge
 - Under this definition, can paradoxically precede science
- Technology?

Examples of Technology

- Fire
- Wheel
- Printing press
- Electronic, Digital Computing
- Internet

Technology definition in this course

In this course, we will treat *technology* as synonymous with *tool*, with the understanding that technologies differ from tools primarily in that:

- enables us to do things we couldn't do before
- improves some metric (time, cost, labour, precision, accuracy, etc.) at least 10x on existing tasks

Humans and Technology

- Note: *not* societal implications of technology
- How do we, as individuals, use technology and/or are affected by it?
 - From perspective of developers of technology

Outline

Software Development Tools

3D Printed Manufacturing

Technology

Administrivia

What we will do in this course

- Read about technology development, esp. computer-science related
 - How do we identify the need for tools based on computing?
 - What impact do computing technologies have?
- Software development “in-the-large”
 - Apply everything you have learnt in your degree so far
 - Gain experience in all software development-related tasks!
- All activities above in the context of 3D Printed Manufacturing
- What we will *not* do: original research

People

- Instructor: Dr. Sreepathi Pai
 - E-mail: sree@cs.rochester.edu
 - Office: Wegmans Hall 3409
 - Office Hours: TBD
- TAs:
 - There are no TAs in this course.

Places

- Class: Hylan 203
 - M,W 900–1015
- Lab: Gavett 312
 - F 940–1055
- Course Website
 - <https://cs.rochester.edu/~sree/courses/csc-293/fall-2023/>
- Blackboard
 - Announcements, Discussions
- GitHub
 - Project development

References

- This course requires a lot of reading!
 - Material will be released weekly
 - Some online, some in Library

Grading

This is a project-based course with 100% of the grade from activities related to the project.

Activities include:

- Reading
- Writing
- Coding/Programming
- Presentations, Demonstrations

Academic Honesty

- Unless explicitly *disallowed*:
 - you may show your code to other students
 - you may use external code (with credit)
 - you may jointly code with fellow students
- You may discuss, brainstorm, etc. with your fellow students
- If in doubt, ask the instructor
- You are permitted (and expected) to make your assignments on GitHub (or similar sites) public

All violations of academic honesty will be dealt with strictly as per UR's Academic Honesty Policy.

Course Goals

- Major: Learn to develop (computer-science-based) technology
- Minor: Learn 3D Printing