CS Capstone Design Document

December 2, 2017

Molecules in 3D?! And in color!? That I can hold in my hand? No way!!!

Prepared for

Oregon State University College of Science Department of Biochemistry and Biophysics

Dr. Victor Hsu ______ Signature ______ Date

Prepared by

Group 63
The Polycules

Joshua Lioy ______ Signature ______ Date
Corynna Park ______ Signature ______ Date
Jackson Wells ______ Signature ______ Date

Abstract

The Molecules in 3D project is designed to provide a platform for the Oregon State University (OSU) Department of Biochemistry and Biophysics (DBB) to print 3D models of molecular objects in multiple colors. This design document covers in detail the specifications of the project as well as general information regarding the purpose of the project. The design views that are covered are 3D printers, multifilament interfaces, computing platforms, filament type, 3D object file type, 3D slicing software, programming language, hardware communication, and conversion software.

Document Revisions

Date	Description	Revision	Editor
11/28/2017	Created the document and added formatting	0	Joshua Lioy
11/30/2017	Added to the Introduction and Design Overview	1	Joshua Lioy, Corynna Park, Jackson Wells
12/1/2017	Completed first draft and submitted	2	Joshua Lioy, Corynna Park, Jackson Wells

CONTENTS

1 Introduction

1.1 Scope

The purpose of this document is to describe the design components that will allow for the implementation of 3D polychromatic printing. A project requested by Oregon State University's (OSU) Department of Biochemistry and Biophysics (DBB), the design will be in conformance with the requirements described in the project's corresponding requirements document. The project will be designed to enhance 3D printing by adding a multicolor functionality while providing a robust workflow so that the process may be easily recreated.

1.2 Design Stakeholders

Design stake holders include the following members of the OSU DBB:

- Dr. Victor Hsu: Associate Professor

1.3 Context

The Molecules in 3D project will create a customized workflow that allows instructors in the OSU DBB to take 3D models of molecular objects and print polychromatic representations of said models using 3D printing hardware. The project exists to allow instructors to enhance their teaching and create better learning experiences for students.

2 Definitions

- **Fused Deposition Modeling (FDM):** This is an additive manufacturing technology that is used for modeling and prototyping. This is done by taking a plastic or metal filament and creating a material by adding successive layers of material.[1].
- **Additive Manufacturing (AM):** This is the process of make a physical object from a 3D digital model.[2]

3 Design Overview

3.1 System Components

3.1.1 3D Printers

Our choice of 3D printer is based on a number of criteria. We need to be able to create objects that are large enough for the faculty and the students in the OSU DBB to handle and to provide an accurate representation of the molecular objects that are being discussed in relevant classes. The choice of our printer also influences the material types that we are able to use to create the 3D objects. As described later in the document we have decided to use PLA as our primary material type. With the ZMorph printer we also have the ability to print in ABS as well as a few other materials if so desired. The main reason for choosing an FDM printer like the ZMorph is for the ease of use that it affords. One person is able to select and feed the filament type that they want to use, set up the print and run it off. For the most part the rest of the printing process is handled by the printer.

For our project we will not be making any changes to the printer hardware other than potential changes to the interface between the printer, the computer, and the multifilament interface. Our plan is for the printer to have the most efficient connection to the rest of the hardware components in our workflow whether that be via USB or a network connection.

Based on our selection of multifilament interface that will be detailed below, we will only be utilizing on of the two filament feeds on the ZMorph printer. However, for the printer to function properly, there must be filament present in both feeders. The Dual Pro head on the ZMorph relies on a specific amount of pressure for successful extrusion. There will be a filament of an arbitrary color feeding into one side and the other feeder will take the mixed filament from the multifilament interface. Other considerations such as printer maintenance and materials costs are outside the scope of our project and will be covered by the OSU DBB.

3.1.2 Multifilament Interfaces

The keystone technology for our project is the multifilament interface. In our case we have chosen to use the Mosaic Manufacturing Palette+. At the most basic level the multifilament interface will allow us to take up to four different colors of filament and print objects that would otherwise only be available in one color. The multifilament interface takes the various filaments and splices them together into a new filament that is then fed into the 3D printer as a single filament. The printer will handle printing no different from before while the multifilament interface is responsible for handling changes in color as the printing process progresses.[3]

In the design of our project, the multifilament interface will sit as an intermediary piece of hardware between the computing platform and the 3D printer. Filament will be fed into the multifilament interface via a rack containing multiple spools of filament. We also intend to an efficient way for a user to change out one or more of the filaments are any time between prints. In our design we have also considered the future possibility of using the multifilament interface to print objects in multiple materials. One example of this would be to print support structure for an object with a water-soluble filament while the object itself could be comprised of up to three different colors.

As the multifilament interface is the main hardware component of our project we will be spending a significant amount of time ensuring the relationship between the multifilament interface and printer and the computing platform and the multifilament interface meets the criteria that was put forth in our requirements document. One criteria will be that the multifilament interface is able to consistently produce a similar quality of print each time an object is made. This is important for us as users may want to run off multiple prints of the same object since the objects created using our workflow will be used for educational purposes. Another consideration is that the time it takes to print polychromatic objects will likely be greater than monochromatic prints. While we do not have a specific target for the speed of the printing process it is still important to take this into account in future design decisions.

3.1.3 Computing Platforms

While not as crucial to the overall success of the project as the two hardware components, the choice of computing platform will influence several design decisions. Based on the current computing platform that we have available to us as well as the considerations that discussed in the technology review we have decided to initially focus on the MacOS platform. One major influencer in this decision was the availability of the software that we are planning to use (this is covered in further detail later in the document). As with some of the other technologies in our project, the computing platform is heavily influenced by our choice of multifilament interface. We must have the ability to enable support for the Palette+ proprietary software if necessary as well as providing the ability for future changes to the selection of software that may interface with the multifilament interface.

Another deciding factor in our choice of computing platform is the flexibility it affords us with some of the Python libraries that we are planning to utilize. We are looking for our computing platform to have strong support for scripting languages. Part of our overarching design is to interweave the varying kinds of programs together using Python scripts. This will allow us to create a cohesive workflow that grants the user a tool for quick and effective polychromatic 3D printing. While Windows does have support for Python, the implementation in MacOS is more streamlined.

Finally, when considering the impact of the computing platform on or design decisions, it is important to address the existing system that is in place at the OSU DBB. Currently the MacOS platform is being used for the printing of monochromatic objects. This makes it a logical choice to adapt our workflow design to incorporate MacOS as our primary computing platform. As we are intending to design our workflow to be adaptable to other computing platforms, as one of our stretch goals we may add a version of or workflow in Windows. Ensuring that the workflow is sufficiently consistent on MacOS will allow us to accomplish the stretch goal time permitting.

3.1.4 Filament Type

3D printing filament is the thermoplastic feedstock which is used 3D printers that utilize fused deposition modeling. The 3D objects are created by melting filament and feeding it through an extruder, building the object up layer by layer, following the patterns and design of the object. Since the filament is what composes the object, the type used directly affects the physical properties of the object that is created. Currently, there are many different materials that are able to be used in 3D printing, each with unique characteristics and specific requirements. Depending on the filament used, 3D printed objects vary in strength, flexibility, durability, and appearance.

Throughout the Molecules in 3D project, the type of 3D filament that will be used is polylactic acid, or PLA. PLA is a the most commonly used material in 3D printing. The material itself is biodegradable and derived from renewable resources; thus, PLA is more environmentally friendly than most plastic materials. Its attributes are as follows: PLA is of medium strength, has low flexibility, and medium durability. PLA is relatively easy to print and requires a print temperature range between 180-220 degrees Celsius. When using PLA, a heated bed is not needed unlike other filament types where heated beds are needed to improve print quality and prevent warping that deforms the model.[4] A notable quality about PLA is its biocompatibility with a human body, meaning it does not produce a toxic or immunological response when exposed to the body or bodily fluids.[5]

This project was purposed for OSU's DBB with the hopes that anyone with and interest and basic knowledge in 3D printing will be able to benefit from it. Since this project intends to be useful to a audience that may vary in time and money resources, PLA was chosen as the filament type because of its accessibility and affordability. PLA is also available to be purchased in different colors, which is necessary for the project's polychromatic requirements. The material is durable, environmentally friendly, and safe to use, all factors that will result in a successful project outcome.

3.1.5 3D Object File Type

Before an object can be printed in 3D, its information must first be encoded and stored in a file. This file is then passed through slicing software to be translated into instructions on how to produce a 3D object that the 3D printer can understand. Depending on the file type, this information is saved as either plain text or binary data and represents physical details about an object such as is shape, coloring, texture, shadings, etc. [6] There are many different file types exported by CAD (Computer-Aided Drawing) software to be used in 3D printing, and these types vary in the amount of information they store.

The file type that will be used for the Molecules in 3D project is 3MF, or the 3D Manufacturing Format. 3MF is a new 3D printing format that is XML-based (Extensible Markup Language), human-readable, and allows design applications to export complete information on an object to a wide variety of applications, platforms, services, and printers. [7] While this file type may be considered more complex than other file types because of the significant amount of data that it can store, data is given through clear and concise specifications which makes the information easy to comprehend. 3MF files are able to fully describe a model, including object geometry, appearance, and more, while narrowing the scope of the data to avoid adding unnecessary detail that may make the file more difficult to process. Since 3MF is a newer file format type, not every software supports it as of yet. However, there is an open source library as well as several Microsoft implemented Windows 10 APIs which allow 3MF files to be read, written, and validated. During its development, a main objective was interoperability, or the ability of computer systems or software to exchange and make use of information between each other. This concern arose because some file types are only able to be used by certain software, restricting their usability across platforms. As a result, 3MF was developed as a neutral format type, meaning that files of this type are able to be understood by any software that supports them. [7]

For this project, the file type used must be able to represent multiple colors as well as depict the appearance and geometry of an object. Because of the interoperability, multicolor capability, and depth of information that it can store, the 3MF file type was determined to be the best fit for this project. Another factor is that 3MF is supported by Ultimaker Cura, which is the slicing software that will be used throughout this project and is further discussed in the next section. It is important to note that since it is a relatively new file type, there may be a limited amounts of resources when it comes to documentation and trouble-shooting.

3.1.6 3D Slicing Software

In order for the 3D printer to understand and print an object, the file containing the data about the object must be translated. This translation is done through slicing software. Slicing software turns the information about an object's 3D attributes into G-code, a control language for machines. This code essentially creates instructions for the 3D printer to follow. [8] The slicing software impacts the quality and physical appearance of the print such as the height of the layers of filament, the thickness of the object's walls, and how hollow or filled the object is.

For this project, the slicing software that will be utilized is Ultimakers Cura. Cura is advertised as "the world's most advanced 3D printer software," suitable for both the novice and expert. Cura's special features, as listed on the Ultimaker website, include 200 customizable settings when optimizing 3D models, the ability to print multiple objects at once, each with their own settings, and downloadable plugins to "create seamless integration with leading design and engineering software." [9] The file formats that this slicing software supports are 3MF, the file type that will be used for this project, STL, and OBJ. Although designed to be used in conjunction with Ultimaker printers, Cura supports many other 3D printers and is customizable so that almost any 3D printer is able to be supported if it is not already. Compatible with Mac OS, Windows, and Linux, Cura is completely free and open-source. [9]

Cura was chosen for this project because it complies with the project requirements, specifically, it is free and open-source, and supports the 3MF file type. From an end product perspective, Cura is advanced enough that it is able to sufficiently communicate enough detail about the 3D object to result in a beyond satisfactory print, through its many adjustable settings and options. However, in terms of usability, Cura, being advertised as being suitable for all levels of 3D printer expertise, is relatively easy to learn how to use and has enough documentation and a support system that enables even the 3D printing novice to take print successfully, given that all other factors are implemented correctly.

3.1.7 Programming Language

The selection of a programming language to be used in our project is an important task. Each language was created with an intended purpose. Some languages are better fit for text parsing and data manipulation, while others are intended to be used as the backbone for a visual interface. The programming language chosen to be used will dictate how we solve the problem of producing polychromatic objects and what intermediary software tools will be put to use. Python is a widely known and respected high-level programming language that was created in 1991. [10] High-level languages enable users to complete computationally large tasks in just a few commands, handling the intermediate steps to a solution in the background. Python has extensive usage documentation, and an almost infinite list of packages. Packages add functionality to a Python script, allowing users to reach their intended solution in less time. Readability is a strong point of Python, an aspect focused on in development of the language. The Python interpretor forces code syntax to be formatted with tabs, ensuring quicker code comprehension from the user.

Python is intended to be utilized by our team for the development of our workflow. Each group member has working knowledge in Python, allowing for collaborative work on coding components the project. Python will be used to manage our workflow, and will handle the execution of the other software tools as needed. Workflow management involves the handling of user input, conversion of said input, sending outputs where they need to go and finally exporting the desired structure to the printer. Initial inputs will be identified by their file type, and sent to the appropriate software tool. For biological data files this software tool will be conversion software, while non-biological data will go straight to slicing software. Preceding the action of files being sent to their corresponding software tools, supplied files will be validated. Validation entails verification that the file's contents correspond to the its extension and are not malicious in nature.

3.1.8 Hardware Communication

This project involves multiple pieces of technology that must work together to produce three dimensional objects. One of the most important hardware conversations that must take place is communication between the workflow and the 3D printer, which involves the transmission of the processed structure file. The execution of this communication between devices can be completed in three ways, which are listed in the printer's user manual. [11]

In our project we plan to utilize two communication methods made available for use by the printer's manufacturer, Local Area Network (LAN) and direct USB connection. Both connections types will be used interchangeably by the workflow, selection depending upon the size of the file supplied by the user. The decision to use both methods of communication increases the versatility of the workflow, allowing for remote printing capabilities when the file is of appropriate size. [12] Using both methods also supplies the workflow with an alternative for communication; if one method is unable to communicate with the printer, the other is utilized. LAN communication between the printer and computer will require both devices to be connected to the same network. Our group will be able to make use of OSU's LAN, as it is publicly available to students and faculty. The computer involved in the workflow is already connected to the LAN, but the printer's connection still needs to be set up. LAN communication will require the development of a queuing system in the workflow, to manage cases where the printer receives another file for printing before the currently printing object is complete. The queuing system will not have a maximum size, progressively printing objects in the order that they arrived in the queue. Files supplied to the workflow will be validated preceding their placement into the printing queue, so users don't have to wait for their job to reach the top of the queue before discovering the file is invalid. When a structure file exceeds 500MB after processing, the USB connection method will be utilized. This is done so that users aren't held up by latencies in network interactions which, when left unattended, can eventually cause whole network slow downs. USB communication is executed using a A to B USB cable, which are very common and readily available.

3.1.9 Conversion Software

Biological structure files supplied to the workflow must be converted preceding the usage of slicing software, due to incompatible file types. PDB and CIF file types are most commonly used to store biological structure data, but our selected slicing software (and most others) cannot take these as input. Because of this, the workflow must include a conversion feature, which will be handled by an exterior software tool. The Python Molecular Viewer (PVM or PyMol) was written using Python, and makes use of many well-known Python packages. [13] PyMol is an open-source piece of software that is publicly available to students and faculty associated with accredited universities, after the obtainment of a usage license. The fact that PyMol is Python based makes it an excellent choice for the project's workflow, as it will integrate seamlessly into the existing Python code.

When biological structure files are received they will be validated and sent to PyMol, which will open a GUI for visualization of the structure. PyMol is a powerful tool, capable of many functions outside of file conversion. These functions will be available to the user as they interact with PyMol, allowing advanced users to fiddle with settings to process their structure files. During the execution of PyMol, the user has the option to color their supplied structure, according to molecule type or other structural features. This is where the workflow transitions from producing monochromatic prints to producing polychromatic prints. Coloring of non-biological structures will take place further on in the workflow. Once the user is finished interfacing with PyMol's GUI, the biological structure output will be written to the current working directory. PyMol is currently able to output structure files in VRML2 format (.wrl).[14] Our workflow aims to utilize the 3MF file type, which is not a possible output from PyMol, but many existing libraries can be used to produce this file type from PyMol's output.

3.2 System Operation

1) An object or biological structure is selected and procured for printing.
2) Command-line interface is opened on computer connected to the 3D printer.
3) The selected structure file is supplied to the Python workflow.
4) Workflow validates input file.
5) (If file is biological) Conversion software is launched by the workflow, user interfaces as needed. Output is stored in the working directory.
6) Slicing software is launched by the workflow, using either the output from conversion software or the non-biological input file. Output is written to the working directory.
7) Workflow sends slicing software output to the 3D printing software.
8) 3D printer receives printing information, and begins printing.

4 CONCLUSION

At the completion of this project, we will have produced a robust workflow capable of printing polychromatic three dimensional objects from biological structure data and non-biological structure data. Individuals associated with the DBB will be able to supply files they desire to have printed, process them, and then watch the objects print inside the 3D printer. Instances of the workflow will be able to be run outside of OSU's LAN, so users don't necessarily have to be on campus to print an object. We hope that the workflow we produce will be preferentially utilized to produce objects that improve student comprehension of concepts presented by instructors in the DBB.

REFERENCES

[1] E. Palermo, "Fused deposition modeling: Most common 3d printing method," 2013. [Online]. Available: https://www.livescience.com/39810-fused-deposition-modeling.html

[2] W. Associates, "What is additive manufacturing?" 2010. [Online]. Available: https://wohlersassociates.com/additive-manufacturing.html

[3] M. Manufacturing, "Palette+." [Online]. Available: https://www.mosaicmanufacturing.com/pages/technology

[4] J. L. Bouthillier. Heat beds in 3d printing - advantages and equipment. [Online]. Available: https://bootsindustries.com/heat-bed-3d-printing/

[5] 3d printer filament types overview. [Online]. Available: http://3dprintingfromscratch.com/common/3d-printer-filament-types-overview/

[6] D. Chakravorty. 8 most common 3d file formats simply explained. [Online]. Available: https://all3dp.com/3d-file-format-3d-files-3d-printer-3d-cad-vrml-stl-obj/

[7] 3mf - supporting innovation in 3d printing. [Online]. Available: https://3mf.io

[8] Jakk. What is slicing software, and what does it do? [Online]. Available: https://www.goprint3d.co.uk/blog/what-is-slicing-software-and-what-does-it-do/

[9] Ultimaker cura software. [Online]. Available: https://ultimaker.com/en/products/ultimaker-cura-software

[10] "About python." [Online]. Available: https://www.python.org/about/

[11] *Zmorph Personal Fabricator User Manual*, Zmorph.

[12] M. Leniak, "How to connect and manage your zmorph machine remotely," 2017. [Online]. Available: https://support.zmorph3d.com/hc/en-us/articles/208882325-How-to-Connect-and-Manage-your-ZMorph-machine-Remotely

[13] "Overview," 2017. [Online]. Available: http://mgltools.scripps.edu/packages/pmv

[14] "File formats," 2016. [Online]. Available: https://pymol.org/dokuwiki/doku.php?id=format