Demonstrating FEDT: Supporting Characterization Experiments in Fabrication Research

Valkyrie Savage
vasa@di.ku.dk
Department of Computer Science, University of Copenhagen, Copenhagen, Denmark

Nóra Püsök
pusoknora@gmail.com
Department of Computer Science, University of Copenhagen, Copenhagen, Denmark

Harrison Goldstein
me@harrisingoldste.in
University of Maryland, College Park
College Park, MD, USA

Chandrakana Nandi
chandra@certora.com
Certora Inc., and University of Washington, USA

Jia Yi Ren
jiayi.ren@ucalgary.ca
University of Calgary, Calgary, Canada

Lora Oehlberg
lora.oehlberg@gmail.com
University of Calgary, Calgary, Canada

ABSTRACT
Recent fabrication research across HCI and graphics shows an incredible diversity of work, much of which features characterization experiments. However, performance and reporting of these experiments are wildly inconsistent, not only reducing transparency that reassures reviewers and readers of a project’s rigour but also challenging a technique’s replicability by future researchers. We propose building a domain-specific language (FEDT: Fabrication Experiment Design Tool) which can express a wide variety of such characterization experiments, and which can be extended to many different machines. This language is sufficiently expressive to describe many types of experiments (3D printing, lasercutting, post-processing), including ones which require human intervention in their steps. We replicate classic fabrication experiments in FEDT to demonstrate its flexibility and discuss the importance and applicability of domain-specific languages and tools to Open Science.

CCS CONCEPTS
• Human-centered computing → Interactive systems and tools.

KEYWORDS
fabrication, experiments, design tools, characterization, domain-specific languages

1 INTRODUCTION
Digital fabrication has become a booming area in Human-Computer Interaction (HCI) research, with a broad range of applications, tools, and techniques. A core part of evaluating this type of work is “characterization experiments,” where authors create fabricated artefacts under specific, varied conditions and subject them to controlled tests to characterize their behaviour. These experiments explore the generalizability and replicability of technical contributions: e.g., will it work with my lasercutter? What if I rotate it 90 degrees before 3D printing? Can users see a glowing effect in an office environment, or only in low light? Researchers fabricate required objects from digital files, using digitally-controlled machinery. Completed objects are subjected to characterization tests, from which emerge digital readings, which are recorded, analyzed with code, transformed into digital graphs, and included in digital conference paper submissions. The interesting HCI challenges, of course, are the interfaces between the human and these digital processes. Humans can contribute at several points: as a fabrication partner plugging gaps in machinery or infrastructure (e.g., to assemble ruffles [Signer et al. 2021]), as a comment on mechanization (e.g., by being the machine [Devendorf and Ryokai 2015]), or as a source of noise in the experiments themselves (e.g., providing unique capacitive signals [Schmitz et al. 2019]). Supporting and cataloging the community’s rich, diverse experimentation while harnessing its essential digital nature could boost open science and replicability [Echtler and Häußler 2018; Feger et al. 2019; Howell and Bateman 2023], and therefore the growth of this vibrant research area. In this spirit, we create FEDT: the Fabrication Experiment Design Tool.

2 RELATED WORK
Formal grammars capture commonalities between a wide variety of structures, like Vega does with graphical visualization [McNutt and Chugh 2021; Satyanarayanan et al. 2017, 2016], and Taxon does with fabrication machines [Tran O’Leary et al. 2021]. In this vein, FEDT captures the common structure of fabrication experiments. Beyond capturing structural commonalities, these grammars can aid in constructing structures; they are fully-fledged domain-specific
Figure 1: The general experimental flow FEDT supports.

languages (DSLs). In fabrication, O’Leary, et al., have published several programmatic interfaces to fabrication machines [Tran O’Leary et al. 2023; Twigg-Smith and Peek 2023], which inform our work conceptually. DSLs are also emerging as tools for experiments. Tea [Jun et al. 2019] and Tisane [Jun et al. 2022] support analysis, helping users choose and execute statistical tests. We interface with Tea, enabling us to focus on experiments’ overall structure.

3 HOW CAN WE DESCRIBE AND SUPPORT FABRICATION EXPERIMENTS?

With inspiration from some exemplar papers and our own practice in the fabrication research space, we describe a general, high-level experimental flow through experiments (see Figure 1). This begins with a user setting up an object’s geometry (CAD), material, and manufacture (CAM) parameters; then continues with creating toolpaths to be sent to the manufacturing machine; and then fabricating the physical object. After an object is manufactured, researchers may post-process it and/or have users interact with it. At some point, data about the object’s properties or behaviours are measured, and perhaps fed to statistical testing algorithms. The experiment is finally reported through text and, often, plots.

We explore scaffolding for independent (intentionally-varied) and dependent (measured) variables introduced at various points of the experimental flow. We also support reporting on the artefacts generated at different stages: geometry files (such as 2D .svg or 3D .stl), manufacture settings files, toolpaths, physical objects, experimental data collection files, and standardized report text.

FEDT is a language embedded in python, which we use in a Jupyter Notebook. It consists of library files that encode and execute particular actions; these are trivially extensible by e.g., passing flags to a commandline slicer to set a print temperature, overriding a lasercutter object to connect with a different lasercutter, or by writing custom functions that are passed as arguments into the library.

FEDT is built of three major components: variables, executors, and workflows. The user defines variables to describe what they will test. Executors are machines and functions that can execute actions in the experiment flow, along with their configurations. Workflows are collections of calls to executors to perform an experiment’s steps. A sample workflow for an existing published work (CircWood, [Ishii et al. 2022]) is shown in Figure 2.

4 DISCUSSION

4.1 Handling Randomness

Fabrication processes are subject to variation that may influence the quality and behaviour of fabricated objects: e.g., factors like atmospheric temperature and humidity affect 3D printing. Thus, it is difficult to know what to model in experiments.

4.2 Group Analysis of Experimental Programs

FEDT provides the opportunity to analyze collections of executed experiments, where parameters and machine settings tweaked and keywords or short-textual descriptions are stored in a linked manner. Search over collections could enable makers or researchers to gain inspiration on relevant settings for future experiments.

5 CONCLUSION

We presented a new type of design language for fabrication research characterization experiments: FEDT. We hope that this will make fabrication research more replicable and portable, in classic Open Science fashion.

ACKNOWLEDGMENTS

This work was partially supported by a Novo Nordisk Fonden Starting Grant under grant number NNF21OC0072716.
REFERENCES


