University of Dayton
eCommons

Computer Science Faculty Publications | Department of Computer Science

10-2018

Developing a Contemporary Operating Systems Course

Saverio Perugini
University of Dayton, sperugini1@udayton.edu

David J. Wright
University of Dayton, dwright1@udayton.edu

Follow this and additional works at: https://ecommons.udayton.edu/cps_fac_pub

Part of the Graphics and Human Computer Interfaces Commons, and the Other Computer Sciences Commons

DEVELOPING A CONTEMPORARY OPERATING SYSTEMS COURSE*

TUTORIAL PRESENTATION

Saverio Perugini[†] *and David J. Wright*[‡]
[†]Department of Computer Science
[‡]Learning Teaching Center
University of Dayton
300 College Park
Dayton, Ohio 45469–2160
(937) 229–4079
{saverio,dwright1}@udayton.edu

The objective of this tutorial is to foster innovation in the teaching of operating systems (OS) at the undergraduate level as part of a three-year NSF-funded IUSE (Improving Undergraduate STEM Education) project titled "Engaged Student Learning: Reconceptualizing and Evaluating a Core Computer Science Course for Active Learning and STEM Student Success" (2017–2020).

PROJECT DETAILS

The IUSE project develops a new model for an OS course that plays a central role in the curriculum of computer science undergraduate degree programs. This new OS course resolves significant issues of misalignment found between existing computer science courses on the topic of operating systems and employee professional skills and knowledge requirements.

The *goals* of this IUSE project are: 1) design a contemporary model of OS curriculum and related pedagogy that involves three

---

progressive modules: a) mobile OS and Internet of Things, b) concurrent programming and synchronization, and c) cloud computing and big data processing; 2) conduct a rigorous research plan to evaluate the effect of the model on a variety of important factors including student learning, retention, growth, and job placement, as well as faculty and STEM/CS education research community engagement; and 3) build a community of practice among computer science faculty at multiple institutions that adopt/adapt the model, or elements thereof, for their own programs and students.

This project produces the course model itself (including module-based content, laboratory lesson plans, and culminating, active-learning project designs) and a set of guidelines and trade-offs, based on the results of our experiments evaluating the model, indicating the degree of improvement in a variety of metrics (e.g., student learning, retention) to be expected from adoption of the model, thus making the model transferable and replicable.

TUTORIAL SUMMARY

This tutorial is the first of three workshops—one per year of the three-year project—intended to promote the use of our model, disseminate the results of the project, and foster community engagement.

In our preliminary discussions with potential external faculty members, they appreciate the ability to plug-and-play with the modular content and see significant value in making use of the content in their teaching activities, especially since developing real-world lab and project plans is quite time consuming. Thus, participation in the project is in-part incentivised through immediate access to the shared module content (i.e., the laboratory manual). For purposes of model transferability and adoptability, and to sustain continued community engagement, we intend to share, through a plug-and-play, crowd sourcing fashion, all the materials for each individual module through an *open-access, federated/curated digital*

repository. The items collected in the repository will be shared and accessed through our socially-engaging *project portal* for computer science educators (`https://sites.google.com/a/udayton.edu/operatingsystems/`). Thus, participating faculty will also have the ability to contribute laboratories to the manual in BitBucket. We provide stipends to tutorial participants who both adopt the model in Spring 2019 and collect pre- and post-evaluative data.

We are also establishing an *advisory group* of computer science faculty members for this project for an external perspective on the model and its adoption. Thus, another goal of this tutorial is to expand the participation of regional faculty in the project advisory group.

TUTORIAL SESSION AND ACTIVITIES

- Introduce attendees to the course model, the plug-and-play content modules therein, and the active-learning exercises in the laboratory manual.

- Introduce attendees to the equipment (e.g., Raspberry Pi) used in the learning environment and active-learning exercises.

- Share our experience in teaching the OS course using this model.

- Invite adoption and evaluation of the course model.

- Invite tutorial attendees to be members of the project advisory group.

ACKNOWLEDGMENTS