



When and How to Introduce Students to Free and Open-Source Software

Inaugural GVSU Technology Summit

Quinton Quagliano, M.S., C.S.P.

Psychology Department

Table of Contents

1	Introduction	2
1.1	Follow Along	2
1.2	Disclosures and Disclaimers	2
1.3	Learning Objectives	2
1.4	Motivation	2
1.5	Purpose	3
2	Vocabulary of Software Availability and Pricing	3
3	Examples of Integration	3
4	Connection to Tech Literacy	3
5	Advantages and Challenges	4
5.1	References	5

1 Introduction

1.1 Follow Along

- See the link to presentation handout on Tech Summit Website!:[www4.gvsu.edu/gvtech-summit]

1.2 Disclosures and Disclaimers

- I have no disclosures or conflicts-of-interests related to this presentation
- I am not a software engineer, computer scientist, or other technology-oriented professional by training - but am an enthusiast and hobbyist

1.3 Learning Objectives

- Appreciate *why* we should pay attention to how software is published and priced (Sections: [Motivation](#) and [Purpose](#))
- Understand the vocabulary used to describe pricing models and source code availability in software (Section: [Vocabulary of Software Availability and Pricing](#))
- Appreciate both the advantages and disadvantages of adopting open-source alternatives into instruction (Section: [Advantages and Challenges](#))
- Consider the ways in which exposure to different tools and methods produces more technology-literate students (Section: [Connection to Tech Literacy](#))
- Learn about some methods to bring more diverse software to students (Section: [Examples of Integration](#))

1.4 Motivation

- The four (hyperbolic) “Evil” Es of software
 - Software is everywhere - it’s always all around us
 - Software is essential - it’s a common requirement of navigating the world
 - Software is elaborate - but it doesn’t look it!
 - Software is expensive - and keeps getting more so!
 - I want my students able to navigate these “Evil Es” during and after college, and not feel lost
-

1.5 Purpose

- Follow the liberal arts mission of creating well-(tech)-rounded students
 - Expose students to more alternative tools
 - Help students see the similarities, differences, and quirks of each tool
 - Ensure that when students encounter new software they can adapt easier
- Support projects and software that are free in a time of increasing prices
 - Push back against reliance upon subscription-based tools
 - Show students how to build their portfolio and skill set without incurring additional financial burden
- This is **not** an attempt to insist upon **only** using open-source software

2 Vocabulary of Software Availability and Pricing

! Important

There's a lot of different ways to describe software price models, so be discerning when looking at options!

3 Examples of Integration

! Important

There are several routes to integration of open-source tools, but all focus on flexibility and agency!

4 Connection to Tech Literacy

! Important

There are several routes to integration of open-source tools, but all focus on flexibility and agency!

5 Advantages and Challenges

Important

Introducing open-source tools brings benefits and drawbacks, but the good outweighs the bad!

5.1 References

- Allaire, J., & Dervieux, C. (2025). *Quarto: R interface to quarto markdown publishing system*. <https://github.com/quarto-dev/quarto-r>
- R Core Team. (2025). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. <https://www.R-project.org/>
- Xie, Y. (2014). Knitr: A comprehensive tool for reproducible research in R. In V. Stodden, F. Leisch, & R. D. Peng (Eds.), *Implementing reproducible computational research*. Chapman; Hall/CRC.
- Xie, Y. (2015). *Dynamic documents with R and knitr* (2nd ed.). Chapman; Hall/CRC. <https://yihui.org/knitr/>
- Xie, Y. (2025). *Knitr: A general-purpose package for dynamic report generation in r*. <https://yihui.org/knitr/>
-