THE SELECTION OF A LANGUAGE FOR INTRODUCTORY PROGRAMMING COURSES: AN EXPLORATORY SURVEY

Thomas A. Ottaway, College of Business Administration, Idaho State University, Box 8020, Pocatello, ID, 83209, 208-282-2601, ottathom@isu.edu
Kevin R. Parker, College of Business Administration, Idaho State University, Box 8020, Pocatello, ID, 83209, 208-282-2601, parkerkr@isu.edu
Joseph T. Chao, Department of Computer Science, Bowling Green State University, Bowling Green, OH, 43403, 419-372-2364, jchao@cs.bgsu.edu

ABSTRACT

The selection of a programming language for an introductory programming course often consists of faculty evaluation, discussion, and consensus. As the number of faculty, students, and language options grows, this process is likely to become increasingly unwieldy. Developing and documenting an exhaustive set of selection criteria, and an approach for the application of these criteria, will allow the process of language selection to be more easily repeated in the future. This paper presents a comprehensive set of criteria and a survey instrument based on the criteria. The objective of the survey is to ascertain the relative importance, or lack thereof, for each of the criteria identified.

INTRODUCTION

The selection of a programming language for an introductory programming course often consists of faculty evaluation, discussion, and consensus. As the number of faculty, students, and language options grows, this process is likely to become increasingly unwieldy. In addition, it lacks structure and cannot be easily replicated. The selection process will, in all likelihood, need to be repeated every two to three years. Providing a structured approach to the selection of a programming language would yield a more thorough evaluation of the options available and a more easily justified selection. Developing and documenting an exhaustive set of selection criteria, and an approach for the application of these criteria, will allow the process of language selection to be more easily repeated in the future. This paper presents a comprehensive set of criteria and a survey instrument based on the criteria. The objective of the survey is to ascertain the relative importance, or lack thereof, for each of the criteria identified.

LITERATURE REVIEW

The programming language used in teaching an introductory programming course can have a great impact on how the course is taught as well as its effectiveness. Language selection has long been a difficult and unstructured process. Fewer issues in the world of software development are as strategic, political, and contentious as the choice of programming language [1]. Over the years various languages have been viewed as contenders for the primary programming languages in IS and CS programs, as seen in Wile's [3] timeline of the succession of programming languages (and language types) throughout their evolution. Often programming language selection is no more systematic than a series of faculty meetings focusing on informal language assessment, debate, and eventual consensus. With the diversity of high-level programming languages available, selecting the "right" language for a computing-focused curriculum or course can be a perplexing process [2]. For many reasons, such as the manner in which students approach problems or scarce computing resources, the selection of a programming language has

ramifications throughout the curriculum [2]. As the number of faculty, students, and language options grows, language selection is likely to become increasingly complex. Furthermore, the selection process currently lacks structure and thus cannot be easily replicated. A structured approach to the selection of a programming language will enable a more thorough evaluation of the available options and a more easily validated selection.

Several factors must be considered when selecting a programming language, and while different curricula place greater emphasis on different factors all must be considered. We seek to develop a comprehensive set of selection criteria and a process for the application of these criteria to evaluate programming languages to be used in programming classes. The selection criteria must take into account the programming features of each language under consideration, the appropriateness of each of these features for beginning (and perhaps advanced) programming courses, the present and future industry acceptance of each language, the availability and quality of textbooks, the costs associated with adopting each language, the infrastructure and support implications of each language, and the impact of the decision on the tactical and strategic direction of the department and curriculum.

SURVEY INSTRUMENT DEVELOPMENT

The research being presented in this paper is exploratory in that we seek to better understand the criteria that are being used by faculty when selecting a programming language for an introductory programming course. Table 1, below, provides a list of the criteria under consideration.

Methodology or Paradigm
Real or Customized
Reasonable Financial Cost
Availability of Student/Academic Version
Availability of Textbooks
Stage in Life Cycle
System Requirements of Student/Academic/Full Version
Operating System Dependence
Proprietary/Open Source
Academic Acceptance
Industry Acceptance
Marketability (Regional and National)
Development Environment
Ease of Learning Fundamental Concepts
Supports Target Application Domain
Scripting or Full-Featured Language
Teaching Approach Support
Debugging Facilities
Support of Web Development
Coding Safety
Advanced Features for Subsequent Programming Courses
Availability of Support
Qualified Instructors and Staff
Anticipated Programming Experience Level for Incoming Students

Table 1. Selection Criteria

[*Please note that each of the selection criteria presented in Table 1 is supported by an extensive literature review, available in the full paper. Please contact the lead author of you wish to obtain a copy.*]

Having developed a detailed and comprehensive set of criteria, a survey instrument has been developed to measure the degree to which the respondent feels each of these criteria is important in the programming language selection process. For example:

> *Industry Acceptance: Industry acceptance refers to the market penetration of a particular language in industry, i.e., the use of a language in business and industry.*

For each criterion the respondent is asked to provide a response on a 5-point Likert scale with responses, numbered 1 through 5, corresponding to *Unimportant*, *Of Little Importance*, *Moderately Important*, *Important*, and *Very Important*. Given that each of the criteria is supported by the literature, the responses are purposely skewed towards positive side (i.e. no Very Unimportant is provided).

CONCLUSIONS AND DISCUSSION

The survey presented in this paper represents the first part of a two part study. A second study is being developed that will attempt to determine the degree to which specific programming languages (Java, Visual Basic.NET, C++, etc.) meet the criteria developed in this research. The target audience consists of faculty that have specified their primary teaching interest as introductory programming computer programming. This approach should increase the validity of the responses. Results and analysis will be available by the time of the conference.

REFERENCES

[1] Jensen, C. (2004) 'Choosing a language for .NET development', Borland Developer Network, http://bdn.borland.com/article/0,1410,31849,00.html

[2] Tharp, A.L. (1982) 'Selecting the "right" programming language', Proceedings of the 13th SIGCSE Technical Symposium on Computer Science Education, Indianapolis, Indiana, pp.151-155.

[3] Wile, D.S. (2002) 'Programming Languages', in J.J. Marciniak (Ed.), Encyclopedia of Software Engineering (2nd ed). Hoboken, NJ: John Wiley& Sons, pp. 1010-1023.