

title: Assignment Four tags:

Tommy Arnzen

Due Date: Wednesday, November 15th, 2017

Assignment: Homework Four and Five

Class: Software Engineering

Table	Table	Table
One	Two	Three

Section One - Hackers and Painters.

Chapter 13 - Revenge of the Nerds

Story

The chapter “Hackers and Painters” displayed an important concept in software engineering: The selection of a programming language can alleviate or add to the burden of a software project. An example can be seen in two organizations: organization A and organization B. Both organizations are developing a similar software system. Organization A decides to use a lesser known programming language (Lisp) to complete their project. They have determined that although this programming language is not as widely known as languages such as Java and C++ they have developers that can use the language to develop and maintain the system. Organization B decides to implement the software system using Java. The decision to use Java was based on the idea that Java is a popular language that meets the requirements of the software system’s development. Management at organization B also believes that if the need presents itself they will be able to quickly replace developers on this project since Java is a popular language. After a period of time both organizations complete their systems. Organization A’s program is written in fewer lines, but organization B’s program was able to implement more libraries and will be better able to be maintained over time.

The purpose of this story is to show that the selection of a programming language depends upon the organization that is developing the software system and the requirements of the software system’s development. In the case of organization A it was decided that a lesser known programming language would be used to reduce the size of the program and enable the software system to be completed earlier than it would have been in another language. In the case of organization B it was decided that Java would be used because it has more libraries, can be more easily maintained over time, and developers can be more easily replaced.

Different programming languages can be used to solve similar problems. In order for a programming language to be selected an organization must take into account multiple factors before making a decision.

Arguments

- I agree that the selection of a programming language for a project should depend on the elements that the project is trying to address. The selection of a programming language should depend on the needs of the system.
- I agree that programming languages should not be selected based on how well they are known. It is important to be able to find programmers to write applications in the programming language selected, but this consideration should not be the only basis for a language selection.
- I believe that programming languages should be selected that can lead to the development of systems that can be maintained over time. I also believe that programming languages should be selected that are commonly known or easily teachable to new developers.
- I agree that programming languages should be selected that can help organizations avoid the mythical man-month that can result in the inability to add developers to long projects.

Takeaways

- Do not use popular programming languages only because they are considered to be industry standards.
- Programming languages are not all equivalent.
- Many organizations use popular programming languages such as C++ and Java because they are considered to be industry standards, and developers can be hired if the organization loses developers.
- If programming languages were all the same then there would never be a change in industry standards. There would be no need to develop new programming languages.
- Java, Perl, Python, and Ruby are all similar to Lisp.
- The more powerful a programming language the shorter its programs will be compared to other programming languages.
- Programming languages that result in longer programs can take longer to develop, which can lead to problems in production.
- Languages vary in power.

Chapter 14 - The Dream Language

Story

Person X has developed programming language Y. Language Y is a new mobile development language that was created to remove the burden of linking dynamic programming languages to XML user interfaces. Person X places the source code for the new language online where a small number of individuals find it. These individuals download the source code and begin writing test applications using the language's new user interface interaction features. As the individuals begin to see the practicality of the new language compared to other options currently available they begin to show it to others. Soon there is a small group of users developing sample applications in the new language to test its functionality. Eventually person X begins to receive feedback, which is used to improve the language's main features, and enhance the reference material.

Eventually language Y's users begin to show their organizations the benefits that the new language can provide over their current methods. As the language begins to be used in commercial settings its benefits begin to be seen by more users. Over time the language becomes the standard programming language for a specific mobile operating system. Since the language is free, open sourced, and has a detailed collection of reference material it is continually used by developers of mobile applications.

Arguments

- I agree with the author that programming languages differ from software systems in that organizations will be less likely to use them if they are not free.
- I agree with the author that a popular programming language should be the scripting language of a system, but I also believe that the system must also be popular. In addition, I believe that as a programming language becomes more popular systems will add support for them for enhancement.
- I agree that a language should have some form of reference material. Reference material should not be frequently changed, and should be updated with new versions of the language.
- I also agree that a programming language should have examples of its functionality that are located in the same area as its reference material.

Takeaways

- Programming languages become popular as hackers start to use them.
- The popularity of programming languages also comes from their use in maintaining legacy software systems.
- Programming languages should be developed to serve a purpose.
- "A language has to be popular to be good." - Programming languages change over time as feedback is received from users.
- "For a programming language to become popular twenty programmers must decide to use the language on their own."
- Three requirements for a popular programming language - Free, offers a reference book, and is the main scripting language for a system.
- Programming languages should be designed to develop throw-away applications.
- Dream languages are open sourced and open design.