Priyanshu Dey

5/2/2023

STRE Extra Credit

Martin

A Comparison of Object-oriented Programing

Object-oriented programming is a widely used programming paradigm that focuses on

modeling real-world objects and their interactions in the software design process. C++ and Java

are popular object-oriented programming languages developed to simplify coding and improve program performance. However, the original developers of these languages, Bjarne Stroustrup

and a team at Sun Microsystems, respectively, had specific design criteria in mind when creating their languages. This paper compares the design criteria of object-oriented programming in C++ and Java as conceived by their original developers.

**Design Criteria of Object-Oriented Programming in C++**

Bjarne Stroustrup developed C++ in the 1980s as an extension of the C programming language. The primary design criteria of object-oriented programming in C++ were performance, efficiency, and compatibility with existing code. Stroustrup intended for C++ to be a fast and efficient language that could be used to write system-level software and applications that required high-performance computing. To achieve this goal, he incorporated features such as operator overloading, templates, and multiple inheritances, which allow for greater control over memory management and enable developers to write more concise and efficient code.Another design criterion of object-oriented programming in C++ is its compatibility with C. Stroustrup designed C++ to be a superset of C, meaning that any valid C code is also valid C++ code. This makes it easier for developers familiar with C to transition to C++ and use its object-oriented features. Design Criteria of Object-Oriented Programming in Java. Java was developed by a team at Sun Microsystems in the early 1990s to create a portable, secure, and easy-to-use language. Platform independence is one of the primary design criteria of object-oriented programming in Java. Java was designed to be a write-once-run-anywhere language, meaning that code is written on one platform that can be run on any other medium without modification. Java programs are compiled into bytecode that can be run on any platform with a Java Virtual Machine (JVM).

Another design criterion of object-oriented programming in Java is security. Java was designed to be a secure language that could be used to write applications for the Internet. To achieve this goal, and Java has a robust security model that prevents unauthorized access to system resources and protects against malicious code. Java What are the specific design criteria of object-oriented programming in Java as conceived by its original developers? It was also designed to be easy to use. The language was designed with simplicity in mind, and its syntax is similar to that of C++. Java eliminates the need for manual memory management, which can be a complex and error-prone process in languages like C++. Java garbage collector automatically manages memory allocation and deallocation, making it easier for developers to write error-free code.

Comparison and Contrast

C++ and Java share some similarities in their design criteria for object-oriented. Programming. For example, both languages were designed to be efficient and capable of high-performance computing. They also support inheritance and polymorphism, essential features of object-oriented programming.

However, some significant differences exist in the design criteria of object-oriented programming in C++ and Java. C++ was designed to be compatible with C, making it easier for developers familiar with C to transition to C++. On the other hand, Java was designed to be platform-independent, making it easier to write code that can be run on any platform with a JVM. C++ allows for greater control over memory management, while Java eliminates the need for manual memory management through a garbage collector. Finally, Java was designed with security in mind, making it a popular choice for writing applications for the Internet.

Conclusion

In summary, the original developers of C++ and Java designed their languages with different priorities in mind, resulting in varying approaches to object-oriented programming. C++ offers greater flexibility and control with its efficient, low-level programming capabilities and support for multiple inheritances, while Java prioritizes simplicity, security, and portability, with automatic memory management and emphasis on single inheritance. In conclusion, C++ and Java are designed with specific criteria for object-oriented programming. C++ is a fast, efficient, and compatible language providing greater memory management control. In contrast, Java is secure, easy to use, and platform independent. Despite their differences, both languages have become widely adopted and are used in various applications and industries. Ultimately, the choice between C++ and Java depends on the specific needs and goals of the development project, as well as the preferences and expertise of the developers involved.

Citations

Stroustrup, Bjarne. "The C++ Programming Language." Prentice Hall, 1985.

Stroustrup, Bjarne. "The Design and Evolution of C++." Addison-Wesley, 1994.

Gosling, James, et al. "The Java Language Specification." Addison-Wesley, 1996.

Gosling, James. "The Java Story." Oracle,2013,

[www.oracle.com/java/technologies/javastory.html](http://www.oracle.com/java/technologies/javastory.html).

Kruglinski, David. "Inside the C++ Object Model." Addison-Wesley, 1996.

Stroustrup, Bjarne. "Why C++ is Not Just an Object-Oriented Programming Language."

Communications of the ACM, vol. 36, no. 10, 1993, pp. 134-148.

Oracle. "Java SE Overview." Oracle, 2023,

[www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html](http://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html).