

The Epic Survival and Resurgence of Java: Chronicled

INTRODUCTION

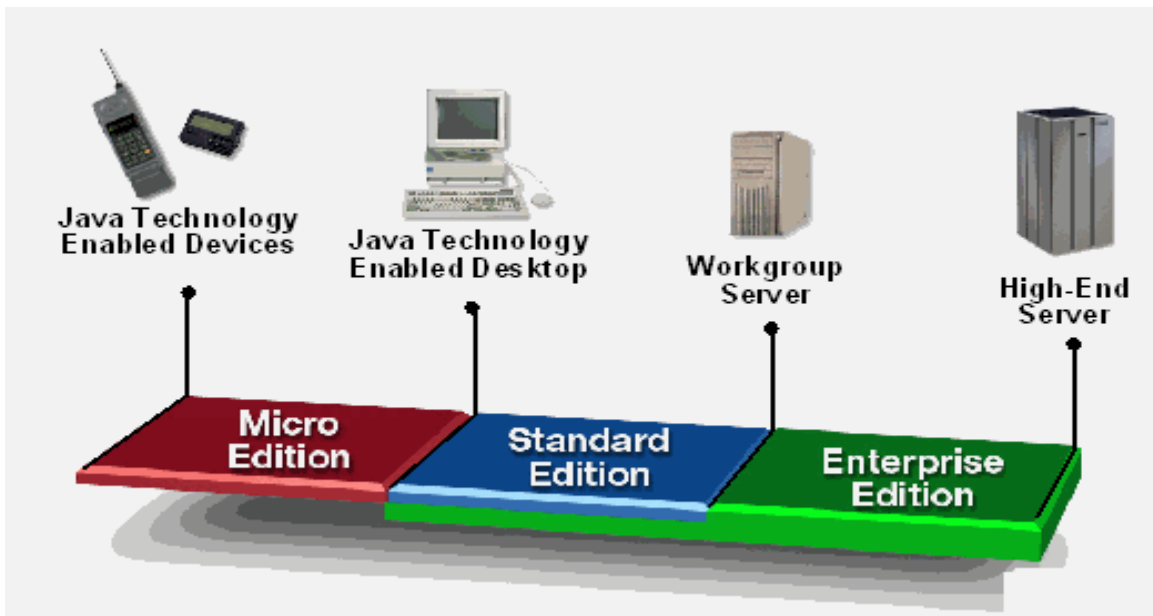


The evolution of programming languages was the bulwark on which the extensive usage of computers manifested itself with humongous proportions. Development and design of fourth and fifth generation programming languages with advanced features to minimize and ease the conceptualization process and aid the creation of cutting-edge applications and products have deemed the increasing need and demand of state-of-the-art enterprise level applications, necessary and inevitable.

The years that have gone by, have seen many a programming language buckle at its knees and surrender owing to its inability to adapt to the swift changes in the development techniques and methodologies. The lone warrior standing amidst this perpetual degeneration is the Java programming language. Such is the interesting tale of its rise, fall, detraction and redemption that its in-vogue period can be a metaphor for the mystical phoenix which rises from its own ashes.

THE INCEPTION

The birth of Java as a programming language is related as an internal project at Sun Microsystems back in 1990. The years there on saw major improvements with 1995 seeing the internet embracing Java as an inherent component, 1996 witnessing the introduction of the Java Development Kit (JDK) and the subsequent versions. The dawn of the new millennia offered no hurdles, instead saw an exuberant emergence of Java features with J2SE and J2ME (Micro edition) versions taking the market by storm. The number of classes readily available within Java swelled in mind-boggling proportions entrancing developers worldwide with this potent mix of development features that manufactured many an innovative application on a global scale.



With major features of portability and the inherent advantage of empowering developers to harness the potential of the language from their desktops has made Java a favourite programming language within the ever-increasing developer community. With such an exemplary background available as a sounding board for the conception of novel applications and services, Java lives up to the idealistic proposition of an entity growing from a fledgling into a highly capable force to reckon with. James Gosling, credited with conceiving Java, envisioned a perfect solution to the increasing demands of having a revolutionary programming language that had the capability to evolve and adapt to the changing times. The state-of-the-art features that Java brings to the table include platform independence, object orientation, rich standard library, applet interface, familiar C++ like syntax and garbage collection.



Platform independence implies that Java compilers produce “byte code” instructions for the Java Virtual Machine (JVM) rather than the usual native object code for individual platforms. Writing a simple byte code interpreter to simulate a

JVM, will make a Java code work on a particular platform which effectively means that the same compiled byte code will run unmodified on any platform supporting Java. Being a pure object oriented programming language entitles Java to treat everything in a program as an object and create a hierarchical tree where everything descends from a root object class. The feature that sets java apart and has since been imbibed by many a programming language, is the standard library containing hundreds of classes and methods segregate into 6 major functional areas. Language support classes providing features like strings, arrays, threads and exception handling; utility classes, Input/Output classes to read and write data of many types to and from a variety of sources; networking classes; abstract window toolkit for creating platform-independent GUI applications; Applet class for designing Java programs to be downloaded and run on browsers provide a vastly augmented environment for application designing. Garbage collection entails that the dynamically allocated memory is automatically freed without any manual intervention from the programmer thus easing the process of writing programs less prone to errors.

All these factors contributed to Java gaining popularity and becoming commonplace in the developer community jargon.

THE DEPLORATION

Every Hero has his detractors. So has Java and such was the case a few years ago that Java was on the verge of extinction, of being declared as obsolete and in danger of being shunned by the developer community. There were reasons, however big or small, behind this opinion and stance. For one, Java being an interpreted language was inherently slow and with the then contemporary computing systems, it proved to be a major hindrance. There are many a factors which cause Java program execution to be slow. All objects within Java are allocated on the heap, right from small objects like iterators etc... to large objects usually having identity semantics. While larger objects do not pose much of an issue, smaller objects can have a big-time impact on the performance. In the absence of templates, Java had to resort to "casts" which increased the code by a significant amount. Java entailed increase memory usage owing to its garbage collection, heap object allocation and larger size of objects. High-level optimizations were unavailable in Java due to the absence of the meta-programming feature.

Another issue with Java that put developers off was the difference in the default look and feel of GUI applications written in Java. Also Java was considered to be predominantly a single-paradigm language which hindered the implementation of the procedural paradigm in the earlier versions of Java. Such was the case that the buzz around Java seemed to fizzle out and the numerous advantages that were enumerated during its inception were totally ignored.



The much touted programming language and its undisputed and unchallenged reign was threatened in a big way so much so that the developments for the language went into a shell and newer languages reared their heads with gusto.

THE RESURRECTION

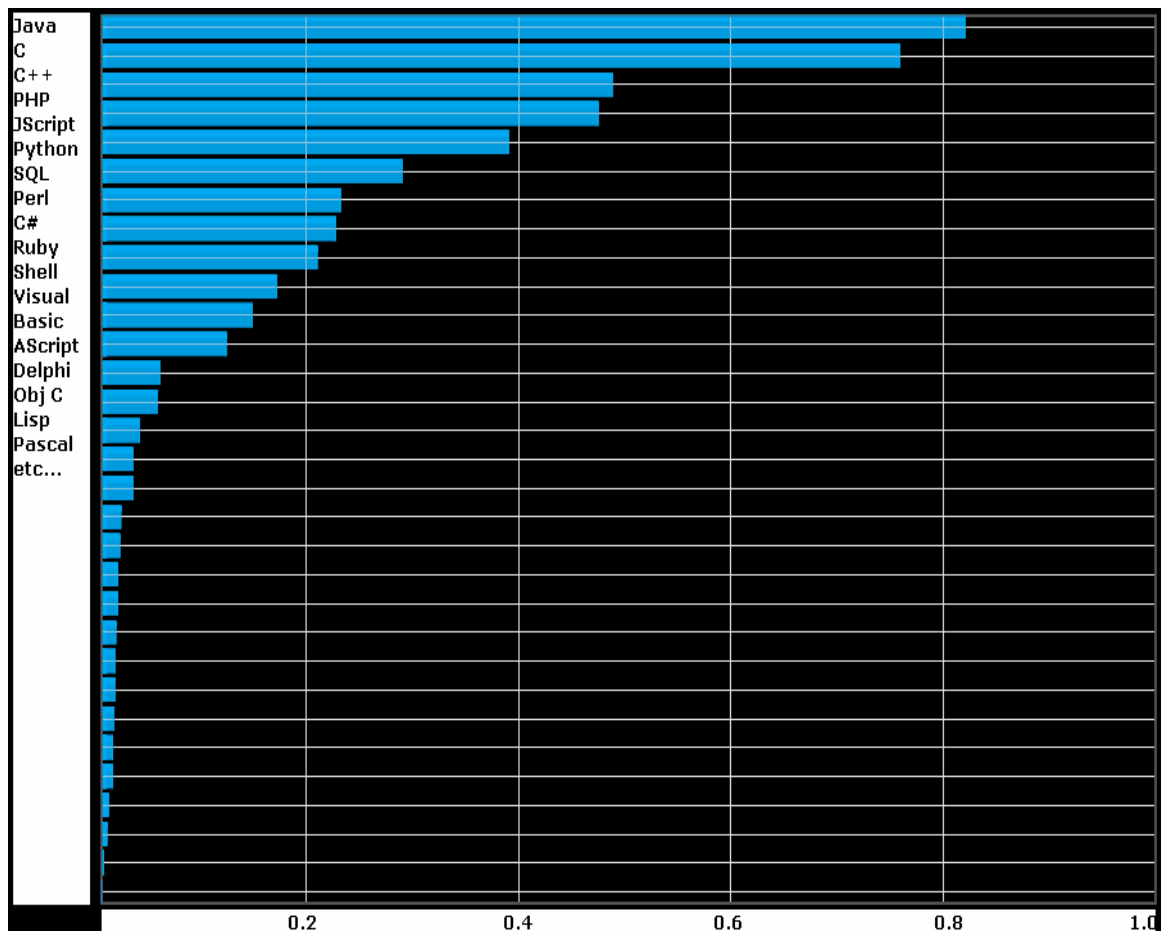
The biggest advantage that Java had by its side was that even though on the global scale the language development had stopped, on the sidelines away from the hue and cry; augmentations and features continued to be added to it. The result was as expected! The resurgence of Java to its strength was with such vitality that it left many a cadavers of later languages in its wake. With numerous features and extended capabilities Java resumed its unabridged march to glory spreading its reach across applications catering to all the domains and verticals. The dawn of the new millennium also ushered in technological advances of the most astonishing kind with the development of faster and more capable machines, wireless access to information, increased mobility and novel innovations in the telecommunication sector.



The virtual machine concept that Java brought along with it was revolutionary. It added more feathers to the already decorated cap that adorned the java mantle. Security capabilities like never before were implemented which attempted to verify all programming before it ran for malicious activity. Even serious errors within a Java program are confined to the virtual machine's sandbox. The Java Virtual Machine (JVM) empowered the program to be written and compiled only once, which would then run on a wide variety of systems without modification. Modern day cell phones and other wireless and embedded devices include a Java VM by default. The Java programming language provides reliability by emphasizing on the early checking for possible errors by providing state-of-the-art compilers that diagnose and detect errors earlier than most other languages. Multithreading, a novel concept

that is mandatory in visual and network programming is smoothly integrated into Java. In comparison, other programming languages have to call system-specific procedures to enable multithreading. Distributed computing is another feature that distinguishes Java by endowing it with an inherent networking capability while allows several computers within a network to work together.

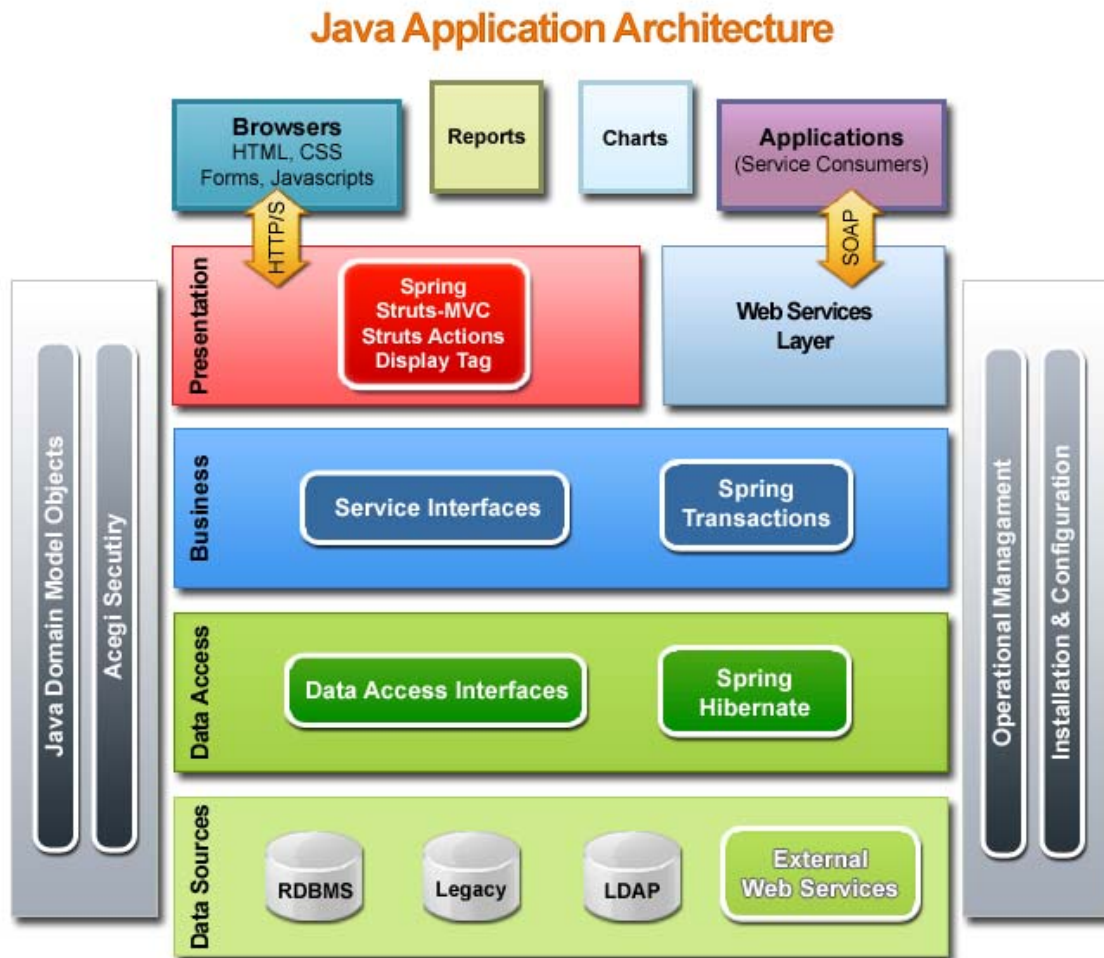
The recent advances in the Java programming language and the subsequent takeover of Sun Microsystems, the owner of java, by Oracle can only be seen as a pedestal from which the development of the language will take a giant leap and evolve into an entity with much wider and far reaching scope and effects. If figures are to be considered, even with the advent of numerous other object oriented languages, Java still rules the roost. Based on various criteria and in-depth research, figures have been brought to light which show that Java towers head and shoulders above other languages and is still a favored choice among developers.



Java, today, has found its way into every possible domain where automation, networking and secure application building is a necessity. Java has been leveraged to a great extent due to E-commerce as portals providing secure online transaction capabilities have sprung up maximizing the scope for virtual transactions to occur. Every online transaction money right from garnering funds, making insurance

payments, shopping online, buying gifts, transferring funds and every other imaginable task can be conceived into an application using Java.

If the forecasts and predictions are to go by, then the latest edition of Java i.e.: Java 7 slated to hit the market by late 2010 will raise the benchmarks to an altogether higher level. The features that are said to be included in the yet to be unveiled release are said to be cutting-edge and novel.



Java 7, codenamed Dolphin, is said to include capabilities like support for dynamic languages with a Multi Language Virtual Machine in the pipeline. A completely revamped and new library for parallel computing on multi-core processors is also incorporated into Java 7. A replacement of the existing garbage collector with a G1 garbage collector to increase speeds is also said to have been included. Strings in switch, better and concise calls to constructors with type parameters or multi-catch in exceptions are also some features that are being tested for inclusion.

With the existing tremendous potential of Java set to be augmented by the newer releases, the future seems brighter than before with never seen-before application ideas and concepts dotting the horizon. With the developer community world over waiting with bated breath for the release of Java 7, the longest survivor among programming languages seems to be the best bet for application design and development. Such is the story of the survival and resurgence of Java that in spite of any hurdles that it might encounter in the future; it is sure to bounce back with a vengeance and continue to exist for a real long time.