

# What's New in Computers

## The Java Internet Programming Language

*T S Mohan*

The Java programming language is poised to change the way Internet is currently being used. In this article, we explore Java and find out why it has become popular with programmers all over the world.

### Introduction

Computers and telecommunications have enabled our progress into the *information society*. The Internet serves as the *information superhighway* that every nation can connect to. It essentially consists of powerful computers with a variety of capabilities, resources and information networked via high-speed and large bandwidth communication channels. These computers are also known as *servers*. Connected to these servers are a large number of desktop systems using which many computer literate participants interact. They can search and obtain the latest information on any topic. In addition, they can share their unique experiences and information, by adding them to the Internet. Information over the Internet is going multimedia; that is, it will include text, graphics, images and sound.

The Internet is a rapidly evolving network of computers, where chunks of related information are available across the world. This information is always evolving, thus making all stored versions obsolete within a short period. It was a challenge to create information structures that looked like a simple document to a user on a computer system, but were actually spread across multiple systems in different parts of the world. This distributed document enabled authors to build-in references to remote documents. The creators of these remote documents could always update them whenever necessary and these updates were always available at any access of the original document.



T S Mohan has been working on distributed programming languages and environments for many years. Of late he has been hacking with Java and HotJava. When he doesn't hack, he hitchhikes around Bangalore. He is currently a Senior Scientific Officer in the Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore.

The Internet is a rapidly evolving network of computers, where chunks of related information are available across the world.



### The Global Information Society

When the comet Shoemaker-Levy broke up into pieces and crashed into Jupiter, the images sent back by the spacecraft Galileo were immediately made available over the Internet by NASA. Unable to meet the access demand by users – astronomers, scientists and lay public – all over the world within hours of release, NASA scientists had to replicate this information in other systems that were geographically spread out. This helped to bring the data transfer load on the computer

networks under control. Thus people all over the world had direct and immediate access to the images sent by Galileo even as the NASA scientists themselves were using it as part of their scientific study. Never before had this kind of information been distributed or used and the results of the efforts by different people studying the rare astronomical phenomenon been shared so easily. The global information society has arrived. The network is the computer.

Initially developed at the CERN labs in Europe and distributed freely over the Internet, the first *world wide web browser* was essentially a *hypertext* document interpreter package that helped follow hypertext links or references to other documents residing in different sites. With this came the distributed hypertext model of transparently structuring information on computer systems, along with a new information transfer protocol for the Internet called *hypertext transmission protocol* or http. A language was developed to express documents in the hypertext mode. This was called the *hypertext markup language* or html. People quickly realized that sending html based documents over the Internet was the ideal way to publish and annotate scientific research and reference material.

People quickly realized that sending html based documents over the Internet was the ideal way to publish and annotate scientific research and reference material.

Augmenting the web browsers with windows based graphical user interface (as in Mosaic and Netscape) made it popular. Currently, hypertext documents comprise text, graphics, sound and images. When sophisticated web documents with features that used non-standard extensions to the html were made, most web browsers around the world ignored them because the underlying computer systems were incapable of supporting them. Examples include documents with very advanced graphics made using graphics workstations supporting more than a million



color palettes, video clippings made with video capture and display package, as well as digitized sound made using audio recording and playing package. In addition, the transmission of such documents over the computer network hogged a large bandwidth. Such documents are never seen in their complete form on most computer systems. In addition, documents were also extended with commands to execute programs needed by the browser. However, this feature quickly ran into trouble. The code plugged into the document by the author was specific to the local processor and operating system and therefore could not be executed in the browser's system. This was perhaps because of a different processor architecture or operating system version or the runtime libraries. In addition, there was the hazard of potentially executing malafide pieces of code called computer viruses embedded within the documents, causing problems to the recipient system and spreading it around. Thus the Internet has thrown up a number of technological challenges like using the net for secure, private and fast commerce as well as supporting video and 3D graphics. Yes, the web has now become the all-pervasive computer! And the *web page* is no more a passive entity.

Given this scenario of the Internet, there was an implicit need to come up with a programming language to enable authors to incorporate program application fragments or *applets* into the web document. It had to be sophisticated, system independent and secure so that the author concentrates on the application and is not concerned with its execution.

### The Programming Language Java

The *Java* programming language was designed to cater to the above mentioned requirements, overcome a number of problems in modern programming practice and enable the fast development of distributed programs and applets. It is a simple, object-oriented language that has been designed to be portable, robust, secure and efficient. Its compiler, runtime system and class libraries have been targetted for most processor architectures and

#### What is Hypertext?

All books and articles are linear documents where the order of information presentation is fixed by the author or editor. This article and the journal are typical examples. Generally the title page is followed by a contents page and then by the articles. Finally we have the references and the index. While browsing through a book we cannot obtain information in any other order of presentation. However, the computer permits one to create documents in the system wherein multiple presentation orders can be designed. The person browsing through it can choose a suitable order. Thus the organization of information becomes non-linear and multi-dimensional. Such texts have *hyperlinks* or cross-references between different parts. Hence they are called *hypertexts*.

### Browsing through a Hypertext Document on the Web

Assume that you have a magic wand and table and are interested in learning how a computer works. You ask for a computer book and wave your wand; and lo and behold, there is the book on computers on the table opened at the page which explains how it works. You read a paragraph in between and see a reference to a computer journal paper. You want it and with a wave of your wand, the referenced document, opened at the topic of interest appears stacked over the book on your table. You start reading this paper and come across another reference, now to an encyclopedia. You wish for it and with a wave of your wand, the third document appears, stacked over the other two and opened at the topic of your interest. You start reading it. You come across a reference to a figure explaining the electronic circuits comprising the CPU. You ask for it and you get a sheet with the figure in all colours and precision. You finish noting the

intricacies of the circuit diagram and wish to get back to the earlier document. With a wave of your wand, the detailed figure disappears and you are left with the encyclopedia you were reading. You find it becoming irrelevant, so with a wave of your wand you get back to the journal paper. You begin to get more curious and seek clarifications. You wish you could ask the authors; you draft a letter and with a wave of your wand, send it off. Not wanting to get into further details of the paper, you wish to get back to the initial computer book you started with. Before doing that you wish to make a note (or put a bookmark) at the journal paper. Again with a wave of the wand, you get it conveniently recorded. Finally, you observe that the journal paper disappears and you are left at the right place in the first book that you started reading on how the computer works. A *web* user has a similar experience where the magic wand is replaced by the mouse and the magic table is your computer monitor.

operating systems. The class libraries are rich and powerful, enabling the quick and effective development of distributed web applications.

Java is a simple, object-oriented language that has been designed to be portable, robust, secure and efficient.

Java is a derivative of C/C++: simple, familiar syntax and with fewer complex features. It does not have many of the poorly understood, confusing and rarely used features of C++. There are no pointers in Java — only tightly bounded arrays. This eliminates the possibility of overwriting memory and corrupting data unwittingly. Pointers have been the source of raw power in C and C++ programs as well as the primary feature that helped introduce bugs into almost all programs. Java is sophisticated enough to help programmers express complex ideas easily, using



arrays and has an efficient *automatic garbage collector* based memory management scheme. In addition, unlike C or C++, Java does not support *structures, enums* or *functions*. Every programmable object is an instance of a 'class' in object-oriented Java. This class definition permits both *static* and *dynamic inheritance* and therefore full reuse of code. A *class* is a template that characterizes an object's internal representation as well as its behaviour. The internal representation subsumes the role of a structure in C or C++. The behaviour of an object is essentially captured in terms of *member-functions* or *methods* in the class definition. In addition, an object template, that is a class, can also be defined in terms of other class definitions or properties or behaviour that can be inherited dynamically (or statically) if needed. Java is truly object-oriented in that many class definitions can be inherited dynamically. This combined with the inheritance of programs across the network from remote systems enables applets to be inherited during runtime. However Java does not support multiple inheritance whose semantics and usage has been quite controversial in other languages. Following the principles of structured programming, there are no *go to* statements, no automatic type-casting or operator overloading. Java is so versatile that one of the big programs first written using it, was its own interpreter and compiler.

The most interesting contribution of Java has been in its runtime system: support for automatic memory management and support for *multi-threading*. The Java garbage collector keeps track of all objects generated, automatically freeing the memory used by objects that have no further use and are not referred to by other existing objects. This contributes enormously to making the code robust. In addition, the support for multi-threading enables the efficient execution of programs that potentially have multiple threads of control. Thus the garbage-collector is efficiently run as a background process overcoming the biggest drawback of garbage collector based languages. Multi-threading is supported via inheritable thread class libraries. It results in better interactive response and near real-time features. This enables Java to inter-

### Why is Java called Java ?

Java designer Jim Gosling couldn't decide on a better name than that of an oak tree growing outside his office window and called it 'Oak'. However it ran into trademark problems for Sun Microsystems. After a bit of a search he called it Java, slang for the best coffee that was imported into California from Indonesia decades ago. Initially Oak was aimed at programming a heterogeneous network of electronic home appliances. This defined Java's characteristics as a small, reliable, real-time system.

Java programs are portable because they can be executed on any system without change.



**What is a URL ?**

A typical website address for a hypertext document consists of four parts. For example, this article is available over the internet at: `http://serc.iisc.ernet.in/~mohan/java-art.html`. The first part ('http:') is the protocol to be followed in accessing the document. The second part ('serc.iisc.ernet.in') gives the domain address of the system which has this document. The third part ('~mohan') indicates that the document is in the home directory of the user with a login 'mohan', while the last part gives the filename of the document ('java-art.html'). Such addresses are also called uniform resource locators (URLs) in the web terminology. Java applications can open and access documents and objects across the network using these URLs with the same ease as accessing a local file.

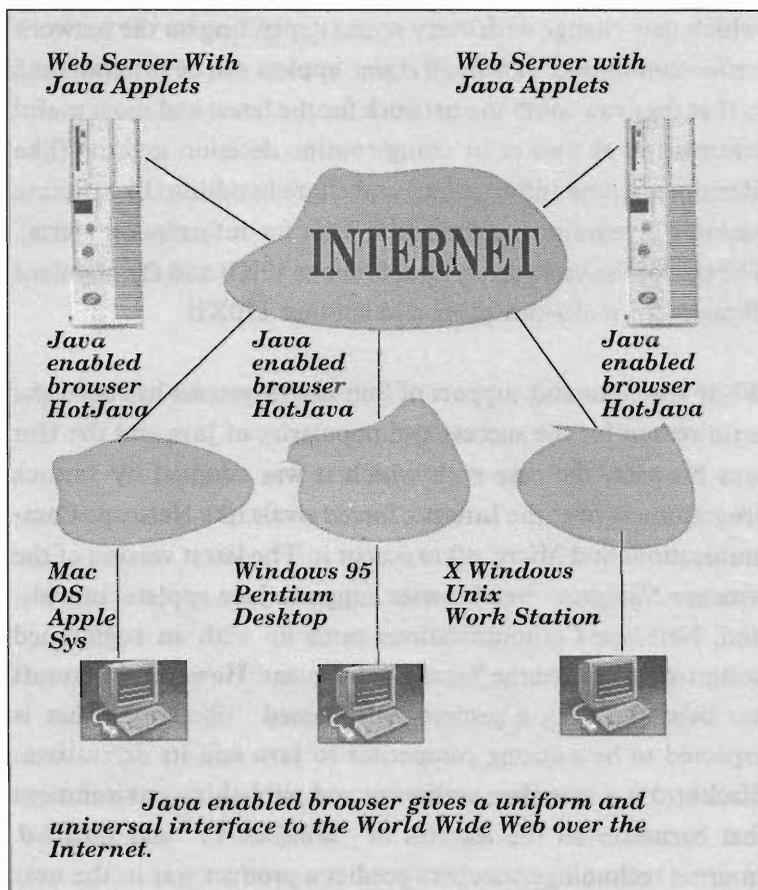
face and support many features of modern operating systems and network protocols.

Java programs are portable because they can be executed on any system without change. The Java compiler transforms the program sources into the instructions of an abstract processor called the *Java virtual machine*. Thus the compiled Java code is architecture neutral. The *bytecode instruction interpreter* specific to each processor type first verifies the consistency of the compiled program and then efficiently converts this code into the native processor instructions before executing it. Java programs are robust because explicit memory manipulations by the programmer are prevented: memory addresses cannot be *dereferenced* nor can *pointer arithmetic* be used to access object contents. Array bounds are checked so that array indices are never out-of-bounds. Java programs are secure: distributed applications have to exhibit the highest levels of security concerns. A *bytecode verifier* in the Java interpreter ensures that the compiled code is strictly language compliant, thus trapping all malafide modifications, more so the computer viruses parading as legal code. In addition, by making the Java interpreter determine the memory layout of all objects at runtime, all possible means of inferring memory contents at compile time and appropriately accessing them at runtime are eliminated — these are the potent security holes in most executables. Using conventional languages to come up with such distributed applications as rigorous as Java has been difficult. It is no wonder that Java is popular with Internet applet programmers.

**Java And Internet**

The Java programming language enabled the web document authors to deliver small application programs to anyone browsing the pages of the html documents. The page became alive because it could create game scoreboards, execute animated cartoons, audio files and video clippings. In addition, it changed the way Internet and world wide web worked by allowing archi-





### Making an Active Web Document

An active Java or html document is essentially a text file with html commands for accessing links, displaying various text strings in different fonts and colours as well as graphical objects. This is along with support for interactive execution of a variety of application program fragments. This makes the document come alive with video and sound while browsing. Thus a book on Assamese folk music can have music programs or applets embedded in the document. When you wish to hear a representative piece, you can execute and hear it on your system. Writing such an applet involves digitizing the musical piece, storing it compactly while retaining the original fidelity, using appropriate class libraries to interface with the sound subsystem hardware and interfacing with the browser's interactive command interface.

recturally neutral compiled code to be dynamically loaded from anywhere in the network of heterogenous systems and executed transparently.

One of the popular web browsers is the *Hot Java* entirely written in Java. This browser brings out the best of Java and Internet programming. It incorporates the Java runtime system and thus enables one to execute Java applets embedded within html documents. The potential applications which can be written using Java are enormous: Secure commercial transactions spanning multiple nations and continents are easy, more so those using electronic cash. Based on it we can develop applications whose usage can be metered across the network — the latest version being available from the nearest easily accessible host

(which may change with every access depending on the network traffic conditions). *Intelligent Agent* applets can be programmed so that they can comb the network for the latest and most useful information as well as in doing routine decision making (like filtering out junk information) and chore handling (like routine backups or monitoring particular sites for information alerts). The size of the basic interpreter is about 40KB and the standard libraries for multi-threading add another 170KB.

In the near future, a sizeable part of our professional activity will be over the Internet, and our success depends on our ability to adapt to and use the Internet technologies.

While the clout and support of Sun Microsystems has been the main reason for the success and popularity of Java and the Hot Java browser, the ease with which it was adopted by various programmers over the Internet forced rivals like Netscape Communications and Microsoft to accept it. The latest version of the *Netscape Navigator* web browser supports Java applets. In addition, Netscape Communications came up with an augmented design of Java called the *Javascript* language. However, Microsoft has been pursuing a project code named *Blackbird* that is expected to be a strong competitor to Java and its derivatives. Blackbird is a complete authoring and publishing environment that harnesses all the features of *Windows 95* and *OLE2.0*. Internet technology watchers predict a product war in the near future wherein Blackbird applications over the Internet, more so in the world of business and financial transactions, are expected to dominate and perhaps almost kill competitive products, something like what *Windows 95* and *Microsoft Office* have done to their contemporaries today. But one outcome is certain — the Web has become the computer — and costly desktop systems will soon be replaced by a simple computer connected to the Internet, costing less than Rs 20,000 and capable of executing any program dynamically loaded from anywhere in the world. In this case, companies will sell computing capabilities (viz., number crunching, visualization, data mines, information warehouses) over the Internet cable even as our electricity boards supply us with power, albeit more reliably. This means that in the near future, a sizeable part of our professional activity will be over the Internet, and our success depends on our ability to adapt to and use the



Internet technologies. If the necessary infrastructure gets in place soon, India will have caught up with the rest of the world and will have arrived in the *global information society*.

### How to Learn More about Java

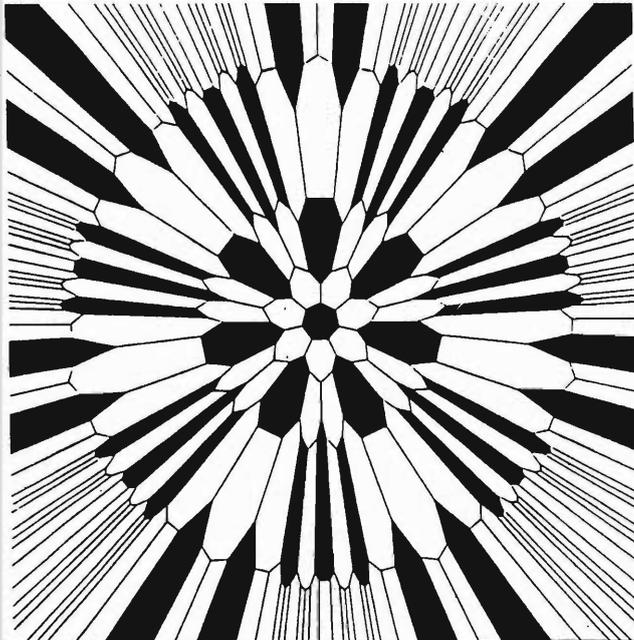
The best source for learning more about Java is the Internet which has many Java application servers: Access the website: <http://java.sun.com> and you can get all documents and the software (including a HotJava browser, a Java compiler, class libraries etc) for free. In addition you can join electronic mailing lists and newsgroups to interact with others around the world who work on Java. The book *Java!* by Tim Ritchey, New Riders Publishing, Indiana, 1995 is one of the first few books on the language.

#### Suggested Reading

Tim Ritchey. *Java!*. New Riders Publishing, Indiana, USA. 1995.

#### Address for correspondence

T S Mohan  
Supercomputer Education  
and Research Centre  
Indian Institute of Science  
Bangalore 560 012, India.



Tessellation of convex heptagons

