JAVA™ on Trial

The case of the People vs Java

by

Daniel Charlu

(Synopsis of the Skit)

Background:

Java is a fairly new object-oriented programming language and an execution environment. It has been brought to trial for fraud and misrepresentation of its benefits! The District Attorney as the chief prosecutor the people has accused Java of ten (10) counts of fraud and misrepresentation which are as follows:

Performance of Java applications (or a lack of)
Integration with existing legacy applications
Support for accessing Database information
Ease of portability of Java applications
Support for creating distributed applications
Easier code development compared to other Object-Oriented (OO) languages
Integrated Development Environments (IDE) for Java
Enabling end-users to compose of applications from software components
Availability of end-user third-party Java applications
Increase in Application development productivity

The defendant is represented by his counsel and is also backed by various customer witnesses.

The Trial: The court's falls into a hush as the judge enters the room. The judge is known for his unconventional style and no-nonsense attitude. The judge quickly starts the proceedings of by reading the ten charges on which Java stands accused. The judge then asks Java if he pleads innocent or guilty. Java says "Not Guilty". The judge then asks the prosecution to bring their first witness...

In the course of events the court starts to understand the benefits that Java brings to the people. The defense systematically argues against each charge...

1. Performance - It turns out that the witness has been using an early Java release. Java creates a new paradigm where the development and deployment of applications can be done in different types of platforms. The key to this is the Java Virtual Machine (JVM) which runs on execution machine. The JVM provides the environment for running the Java code. Earlier versions of the JVM interpreted the code line by line and so execution quite slow compared to natively compiled program. However, now the JVM also provides a Just-in-Time (JIT)

compiler which takes the code and dynamically compiles it to machine code. This increases the performance in some cases up to 50 times over interpreted code. In addition, it also becomes apparent that some people are not interested in portability but only in performance. To enable this, the court learns that now there are native Java compilers in the works which can provide native performance as good as existing C++ compilers.

2. Integration with legacy applications - The defense brings forward testimonials from customers who are integrating existing applications with Java, using the Java Native Interface API (Application Programming Interface) which is provided with Java Release 1.1 These customers are integrating new Java applications with existing C/C++ programs.

3. Accessing Database information - It also becomes apparent that access to Database information is now possible using the Java Database Connectivity (JDBC™) API . SQL databases such as Oracle, Informix and Sybase are now accessible by Java applications.

4. Portability of Java applications - The defense explains to the court the 100% Pure Java initiative. A programmer who writes 100% Pure Java code is ensured that his/her applications can run over a wide variety of platforms supporting 100% Pure Java without any changes .

5. Creating Distributed applications - The notion of Remote Method Invocation (RMI) API is presented to the court. This allows Java applications to access remote systems and invoke methods i.e. functions and/or resources on those machines. In addition Java also provides the basic networking support for managing connections. Thus enabling the creation of distributed Java applications.

6. Easier code development - The defense again explains to the court the features that Java provides compared to C++; such as built-in garbage collection, no pointers, networking support and 16-bit Unicode characters for internationalization. It is pointed out that in development efforts using other OO languages are in many cases, significantly increased, just in debugging garbage collection code, and inherent memory leaks.

7. Integrated Development Environment (IDE) - There are now various IDE packages for different types of platforms. For example the "HP edition of Java WorkShop by Sun" is available on HP-UX. In addition the same IDE is also available on Windows NT. This allows one to develop an application on a Windows based PC, and deploy it on an HP-UX system. In this particular case the HP-UX Java WorkShop can used to fine tune or debug the Java application.

8. Software Components - Java also provides a component model called JavaBeans™. So now a end-user can use software components from multiple sources and put them together. The resulting application provides the functionality that the end-user requires.

9. Third-party applications - With Java now offering a rich set of features, a number of application providers such as Oracle, Baan, Lawson Software and Corel, are either adding new features using Java to existing products, or they are rewriting new applications using

Java. Furthermore, a number of vendors have Java plans in the works. While Java matures and improves even further there will be a plethora of applications.

10. Increase in Productivity - It is quite evident that there is a temporary drop in productivity when people move from procedural languages, such as C or Pascal, to OO languages such as C++ or Java. However, in a few months of using Java, productivity quickly improves and climbs dramatically. The reasons being the rich API set, the networking and garbage collection support, to name a few.

Having defended all the 10 charges, the defense counsel requests the judge to acquit Java of all charges. The judge then asks the people present to decide for themselves.

What is YOUR decision ?