Java: Introduction and Overview

Originals of Slides and Source Code for Examples:
http://courses.coreservlets.com/Course-Materials/java.html

For live Java-related training,
see http://courses.coreservlets.com/
or email hall@coreservlets.com.

Taught by the author of *Core Servlets and JSP*, *More Servlets and JSP*, and this tutorial. Available at public venues, or customized versions can be held on-site at your organization.

- Courses developed and taught by Marty Hall
  - Java 7 or 8 programming, JSF 2, PrimeFaces, Ajax, jQuery, Android development, servlets/JSP, custom mix of topics
  - Courses available in any state or country. Maryland/DC area companies can also choose afternoon/evening courses.
- Courses developed and taught by coreservlets.com experts (edited by Marty)
  - Hadoop, Spring, Hibernate/JPA, GWT, HTML5, RESTful Web Services

Contact hall@coreservlets.com for details

Topics in This Section

- **Truths / Myths About Java**
  - Java is Web-enabled?
  - Java is safe?
  - Java is cross-platform?
  - Java is simple?
  - Java is powerful?
  - Java is popular?
- **Java versions and application areas**
  - Standard edition
  - Enterprise edition
  - Micro edition (and Android Edition)

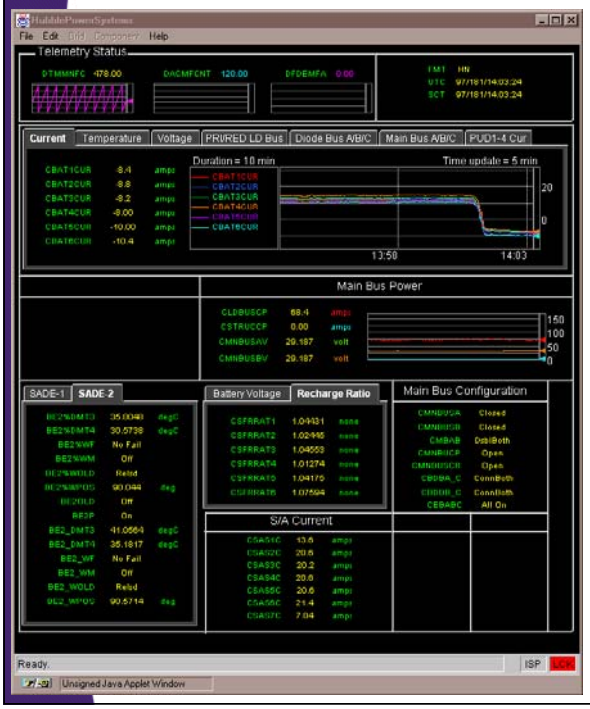
Overview of the Java Language

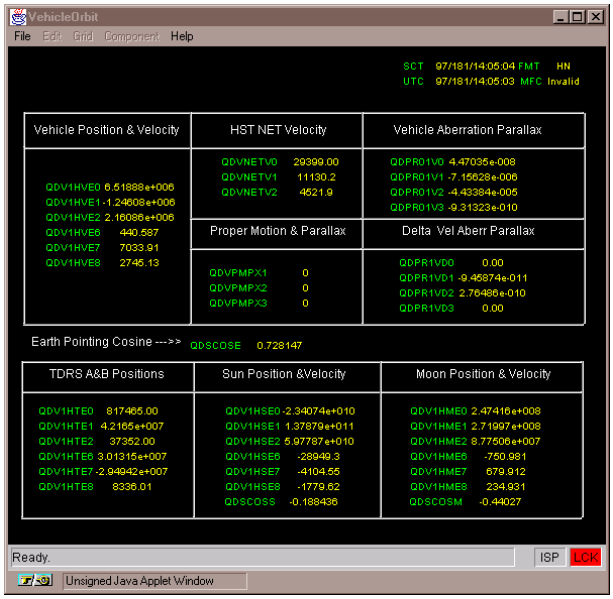
Java is Web-Enabled?

- **Truth: Web browsers can run Java "applets"**
  - The Web can be used for *software* delivery and *execution*, not just *document* delivery and *display*
  - No more installation or updates; just a bookmark
  - Large, complex applets best suited for intranets. Fits the APL model better than the WWW at large.
- **Truth: Java's network library is easy to use**
  - Ordinary mortals can do socket programming
  - Standard distributed object protocol and DBMS API

Hubble Space Telescope Monitoring:

"NASA Goddard's Most Successful SW Project *Ever*."

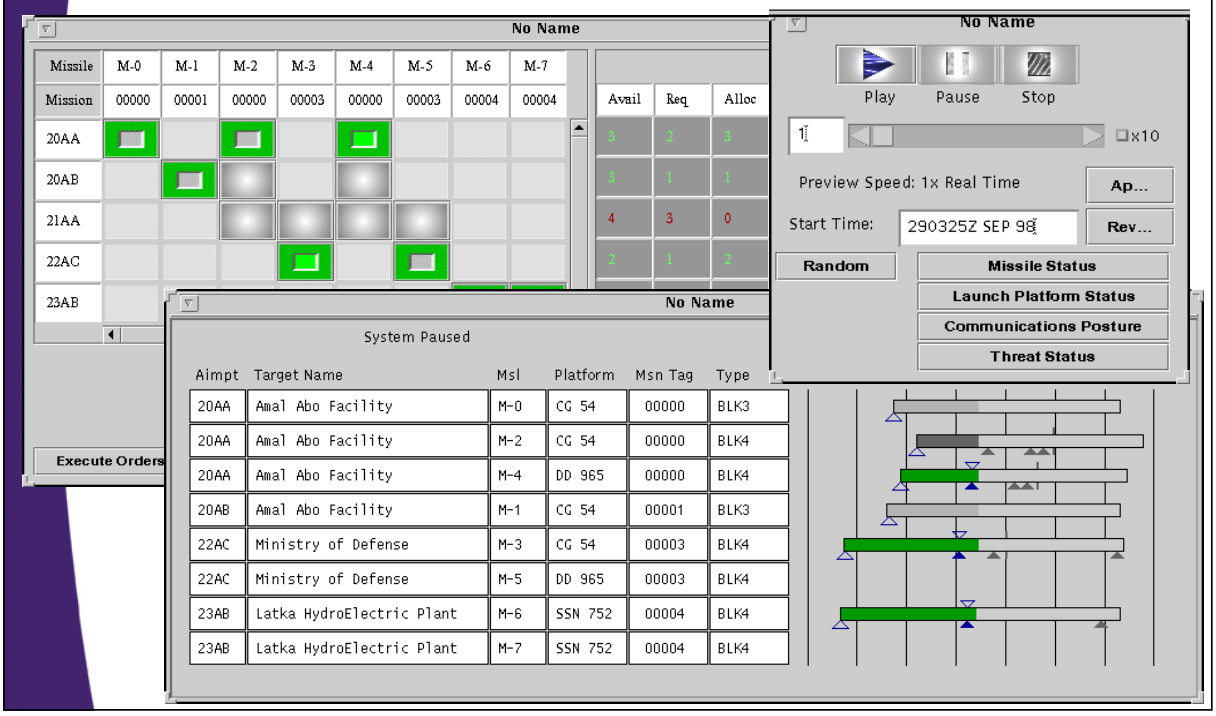




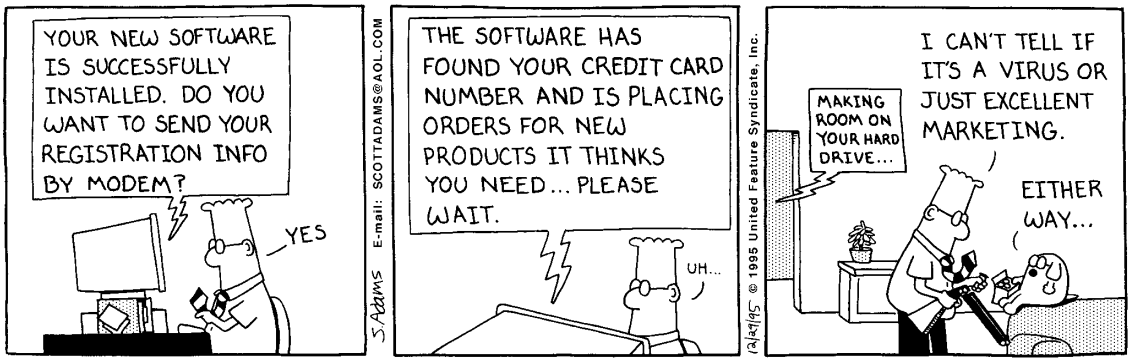
Java is Web-Enabled?

- **Myth: Java is *only* for the Web**
  - Java “applets” run in Web pages
  - Java “applications” run stand-alone
  - Current usage (roughly)
    - Client (applet): 5%
    - Desktop (application): 10%
    - Mobile (Android/Blackberry): 25%
    - Server (JSF/servlets/JSP/Hadoop): 60%

Tomahawk Strike Coordination Planner (APL/PPSD)



Java is Safe?



Dilbert copyright United Media. Used with permission.

- **JAVA: Just Another Virus Architecture?**

Java is Safe?

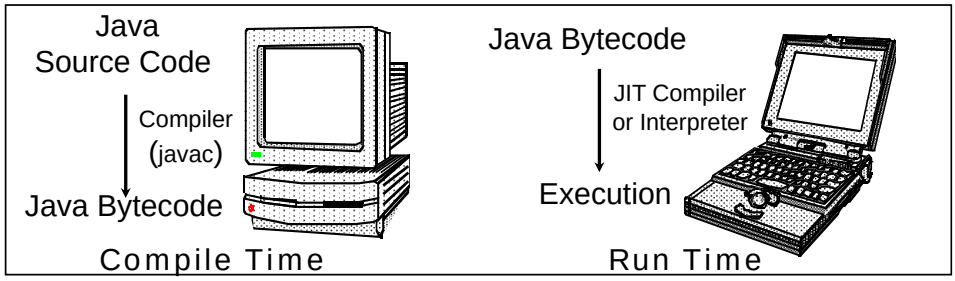
- **Truth: Restrictions on permissible operations can be enforced**
  - No "raw" memory manipulation (directly or indirectly).
    - Thus, it is easy to identify prohibited operations.
  - Applets, by default, prohibited from:
    - Reading from the local disk
    - Writing to the local disk
    - Executing local programs
    - Opening network connections other than to the HTTP server that the applet came from
    - Discovering private info about user (username, directories, OS patch level, applications installed, etc.).

Java is Safe?

- **Myth: Applets cannot harm your computer**
  - Denial of service
  - Browser misconfiguration
  - Implementation bugs
- **Myth: Java is too restricted to be useful**
  - Restrictions apply only to applets, not regular Java programs
  - Digital signatures support relaxed restrictions
- **Myth: Applets with digital signatures are no more or less safe than ActiveX**
  - Relaxed security in applets not “all or nothing” as in ActiveX

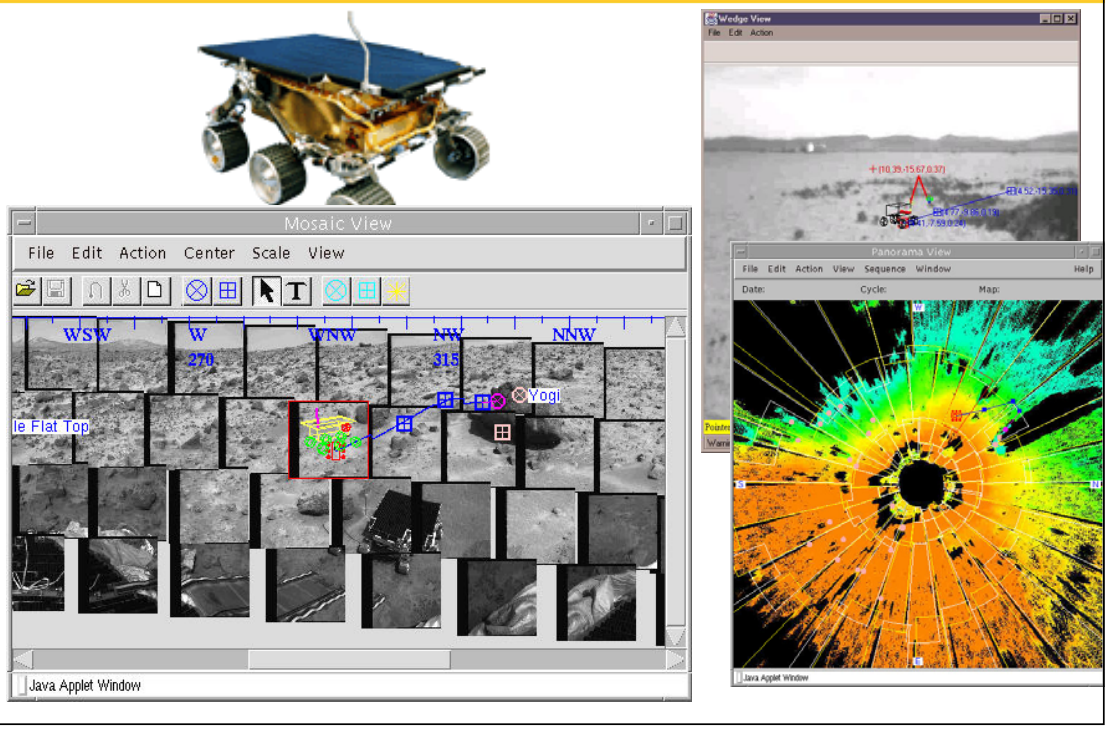
Java is Cross-Platform?

- **Truth: Java programs can compile to machine-independent bytecode**



- **Truth: All major operating systems have Java runtime environments**
  - Most bundle it (Linux, Solaris, MacOS, Windows XP)

Mars Rover Controller and Simulator



Java is Cross-Platform?

- **Myth: Safety and machine independence can be achieved with no performance penalty**
  - Current systems are about 20% slower than C++
  - Upcoming releases claim to lower or eliminate that gap
  - I expect the gap to stay at 10% or more
- **Myth: Java is interpreted**
  - Early releases were interpreted
  - Many major “Just in Time” (JIT) compilers

Java is Cross-Platform?

- **Myth: Write Once Run Anywhere**
  - Cross-platform code can be achieved, but you must test on all platforms you will deliver on.
    - Java apps can execute local code
    - The graphics library behaves slightly differently on different platforms
    - The behavior of the thread scheduler is only loosely defined

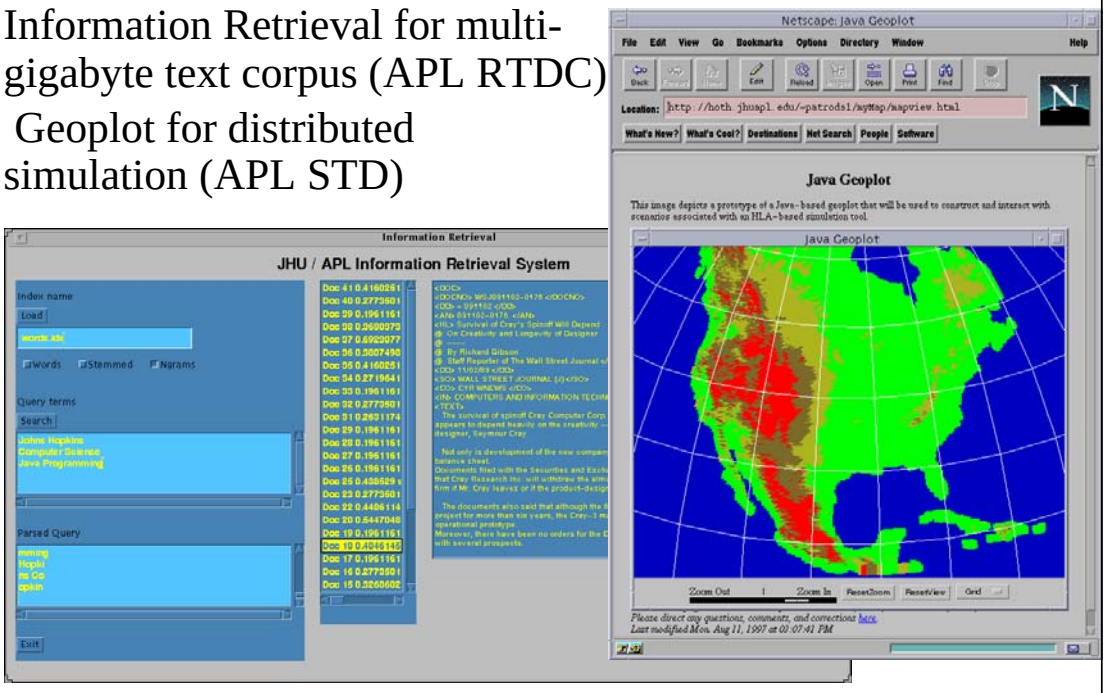
- **Myth: Java will kill Microsoft**
  - There is also no longer immediate danger of the reverse (Microsoft killing Java)
  - Microsoft wavered between trying to fight Java and joining it and making money by dominating the market. With .NET, they are back to fighting it again.

Java is Simple?

- **Truth: Java greatly simplifies several language features**
  - Java has automatic memory management
    - Does Windows and takes out the garbage
    - No dangling pointers. No memory leaks.
  - Java simplifies pointer handling
    - No explicit reference/dereference operations
  - No makefiles for simple applications
  - No header files
  - C++ syntax streamlined
  - C# is comparable to Java, at least as far as the core language goes.
    - For a comparison of Java and C# syntax/constructs, see http://www.harding.edu/fmccown/java1_5_csharp_comparison.html

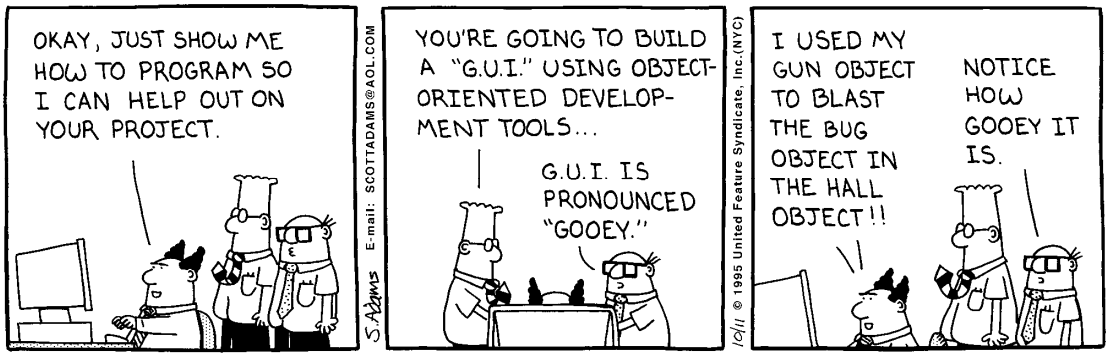
Rapid Application Development in Java

- Information Retrieval for multi-gigabyte text corpus (APL RTDC)
- Geoplot for distributed simulation (APL STD)



Java is Simple?

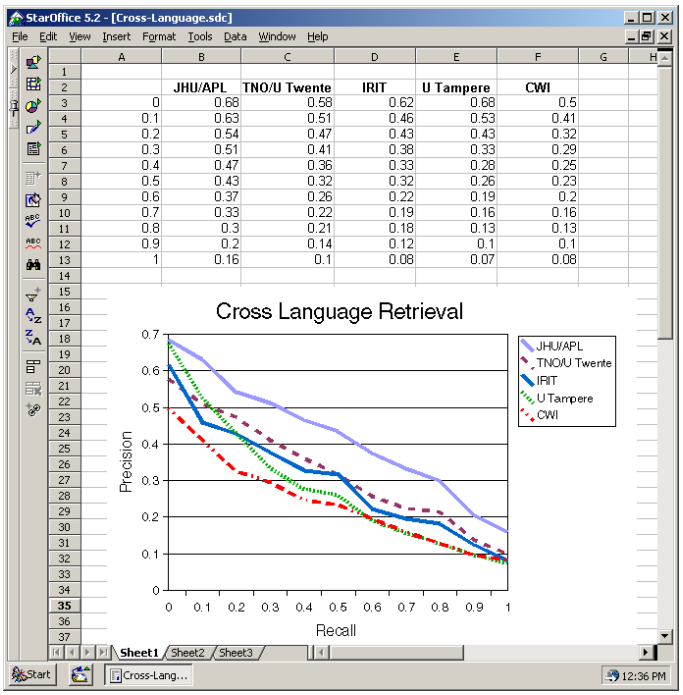
- **Myth: Java *programming* is simple**



Dilbert copyright United Media. Used with permission.

- Programming is always hard
  - Java is nothing like HTML; only a little bit like JavaScript
- Programmers typically push complexity envelope
  - Multithreaded and network programming

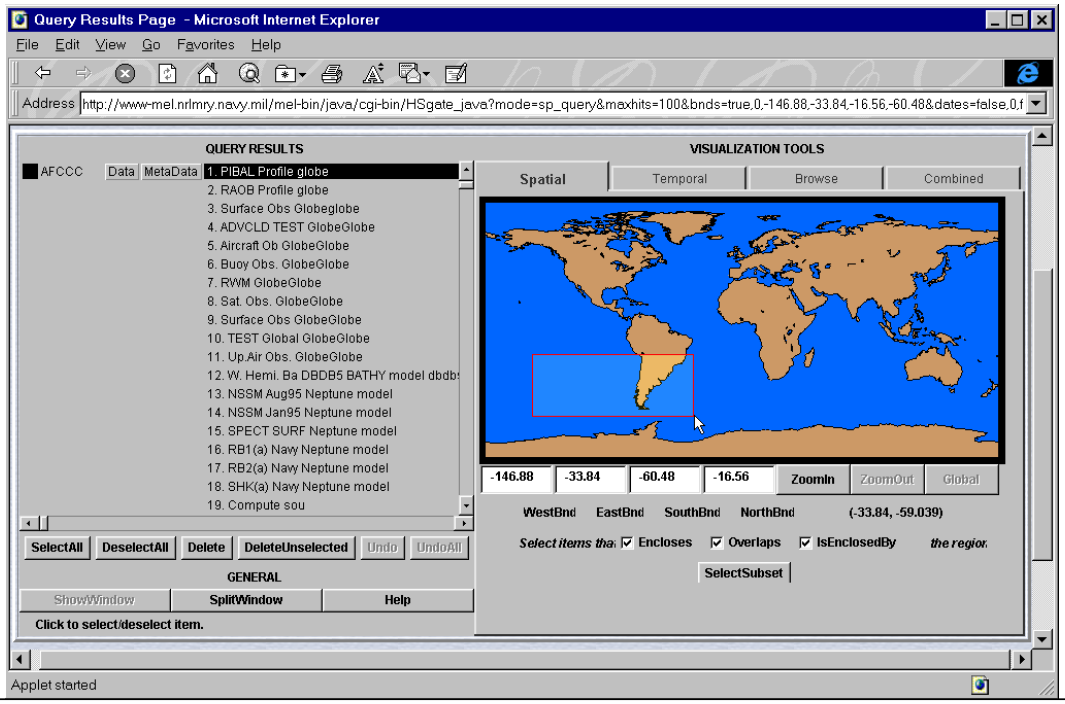
Star Office
MS Office Competitor Written in Java



Java is Powerful?

- **Truth: Java has a rich set of standard libraries**
  - Networking
  - Threads (lightweight processes)
  - Distributed objects
  - Database access
  - Graphics: GUI controls and drawing
  - Data structure library
  - Arbitrary precision integral and fixed-point arithmetic
  - Digital signatures
  - Serialization (transmitting/reassembling data structures)
  - File and stream compression
  - XML parsing
  - Web services

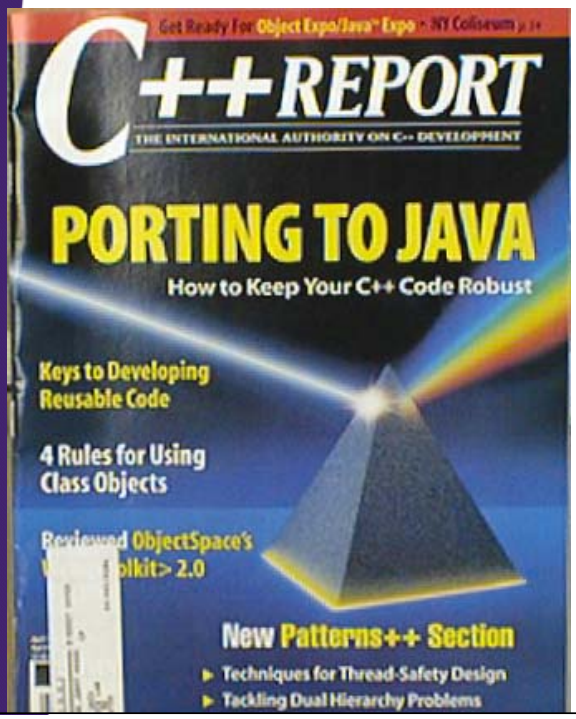
MEL - Master Environmental Library (DMSO)



Java is Powerful?

- **Myth: Java will increase programmer productivity for all applications by XXX%.**
- **Myth: Java will kill C++**
- **Myth: All software should be written in Java**
  - Unix utilities: C
  - Desktop utilities: Python, Perl
  - Small/medium Windows-only programs: Visual Basic
  - String parsing: Perl
  - High-performance, single-platform OO systems: C++
  - Air traffic control, aircraft flight software: Ada
  - Knowledge-based systems: Lisp/CLOS
  - High-performance number crunching: FORTRAN
  - Java also a good alternative for many of these

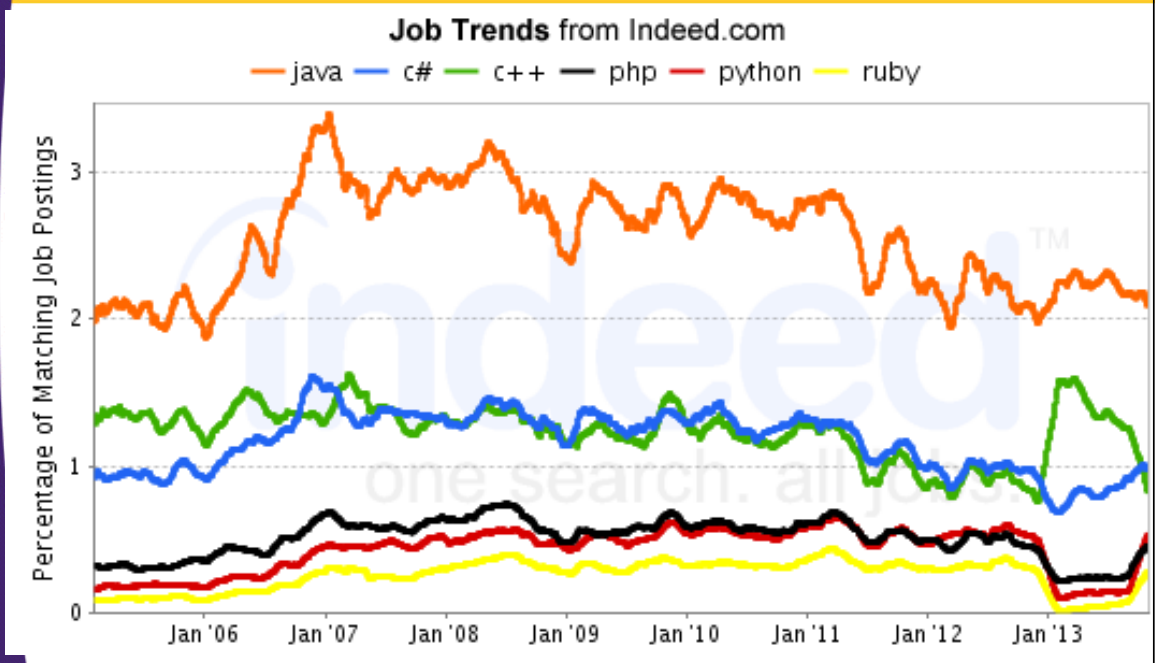
Java and C++



Although Java will certainly not kill off C++, Java and C++ do compete for some of the same territory.

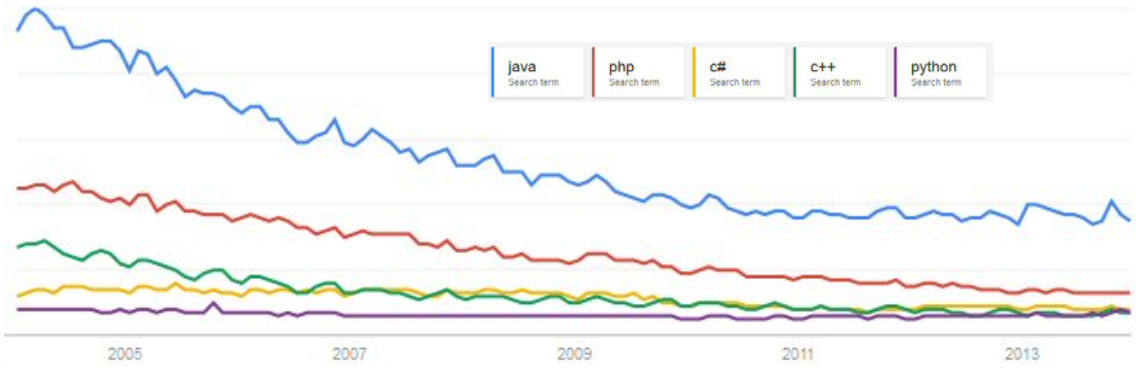
Hmm, does *The C++ Report* think that the way to keep your C++ code robust is to port it to Java?

Java is Popular?



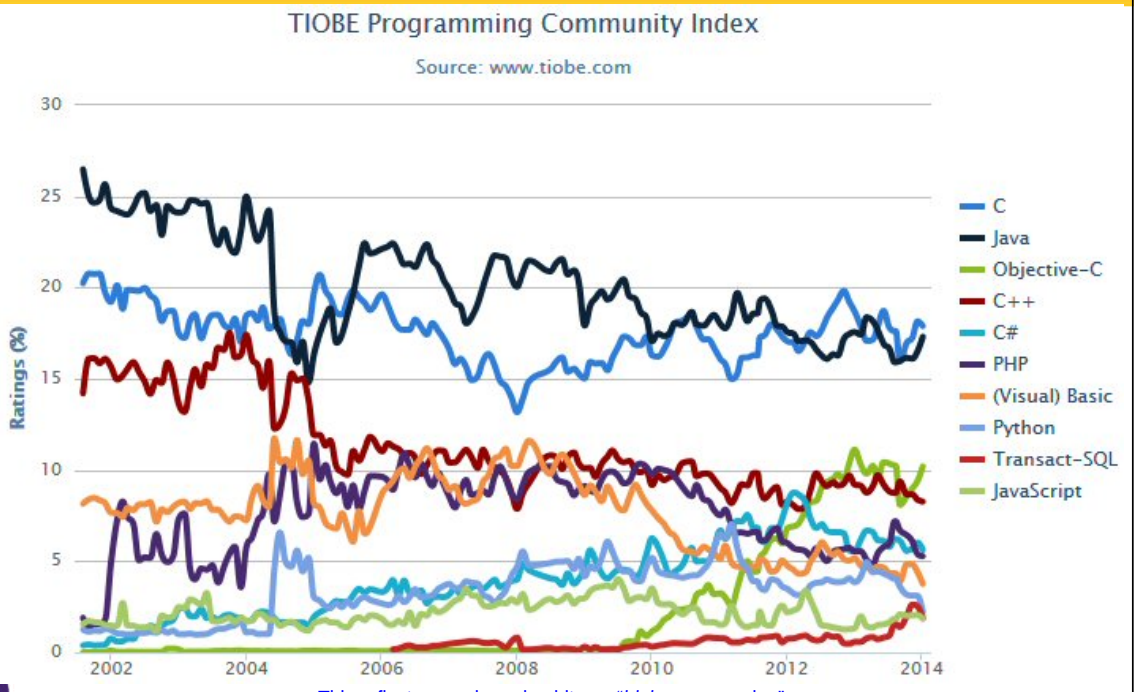
This reflects job postings that contain the keyword in the title or keywords. Since value is in percent, the specific ups and downs are not so relevant (perhaps there was a spike in teacher jobs then), but the relative values are instructive.

Java is Popular?



This reflects searches at Google

Java is Popular?



This reflects search engine hits on "*blah* programming".

Major Java Versions

Standard Edition

- **Java SE**
  - This is often what people mean when they say "Java" or "the Java programming language"
- **Applications**
  - Desktop programming
  - Applets
  - Java WebStart
  - Java FX
  - Base on which to build Web apps that are not full Java EE
- **Famous examples**
  - Limewire
  - Eclipse, NetBeans, IntelliJ IDEA
  - Yahoo games
  - Ant and ANTLR
  - GWT (Google Web Toolkit) and Laszlo

Enterprise Edition

- **Java EE (formerly "J2EE")**
  - This is Java running on app servers
- **Applications**
  - Servlets, JSP, JSF, Struts, EJB, Spring, Hibernate
- **Famous examples**
  - Google home page, gmail, Google Maps, Google Docs
  - Ebay and PayPal
  - walmart.com, kmart.com, target.com, kohls.com, macys.com, homedepot.com, ikea.com, llbean.com
  - travelocity.com, orbitz.com, hotwire.com, hotels.com
  - Baltimore Orioles, Washington Nationals, Washington Redskins

Micro Edition (or Java SE for Phones)

- **Java ME**
  - This is Java running on small devices
- **Applications**
  - Cell phone apps, embedded apps, printers, etc.
- **Famous examples**
  - Blackberry
  - Android
    - Really optimized Java SE, not Java ME
  - Amazon Kindle
  - All Blu-Ray DVD players
  - Sony Ericson phones
  - EA Mobile

Java + Kindle: Amazon's new wireless reading device

Wrap-Up

Summary

- **Java is a general purpose language**
  - Supports standalone apps, browser-based applets, server-side programs, cell phones, and more
  - It is by far the most widely used language in the world
- **Java has a number of good features**
  - But not better in every way than all other languages
  - Few of the technical features were new to Java
- **Reasons for using Java**
  - Combination of technical features, widespread use, available developers, tools, and libraries
  - But in most application areas, other languages are also viable alternatives

Questions?

JSF 2, PrimeFaces, Java 7 or 8, Ajax, jQuery, Hadoop, RESTful Web Services, Android, HTML5, Spring, Hibernate, Servlets, JSP, GWT, and other Java EE training. Also see the Java 8 tutorial and general Java programming tutorial.