IT Process Automation & Private Cloud Management

iwave™
SOFTWARE

Modern Java Development

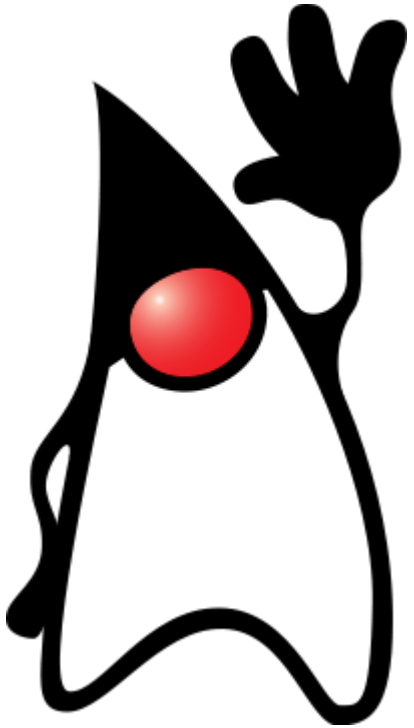
Chris Dail
http://chrisdail.com
Twitter: @chrisdail

What is Java?

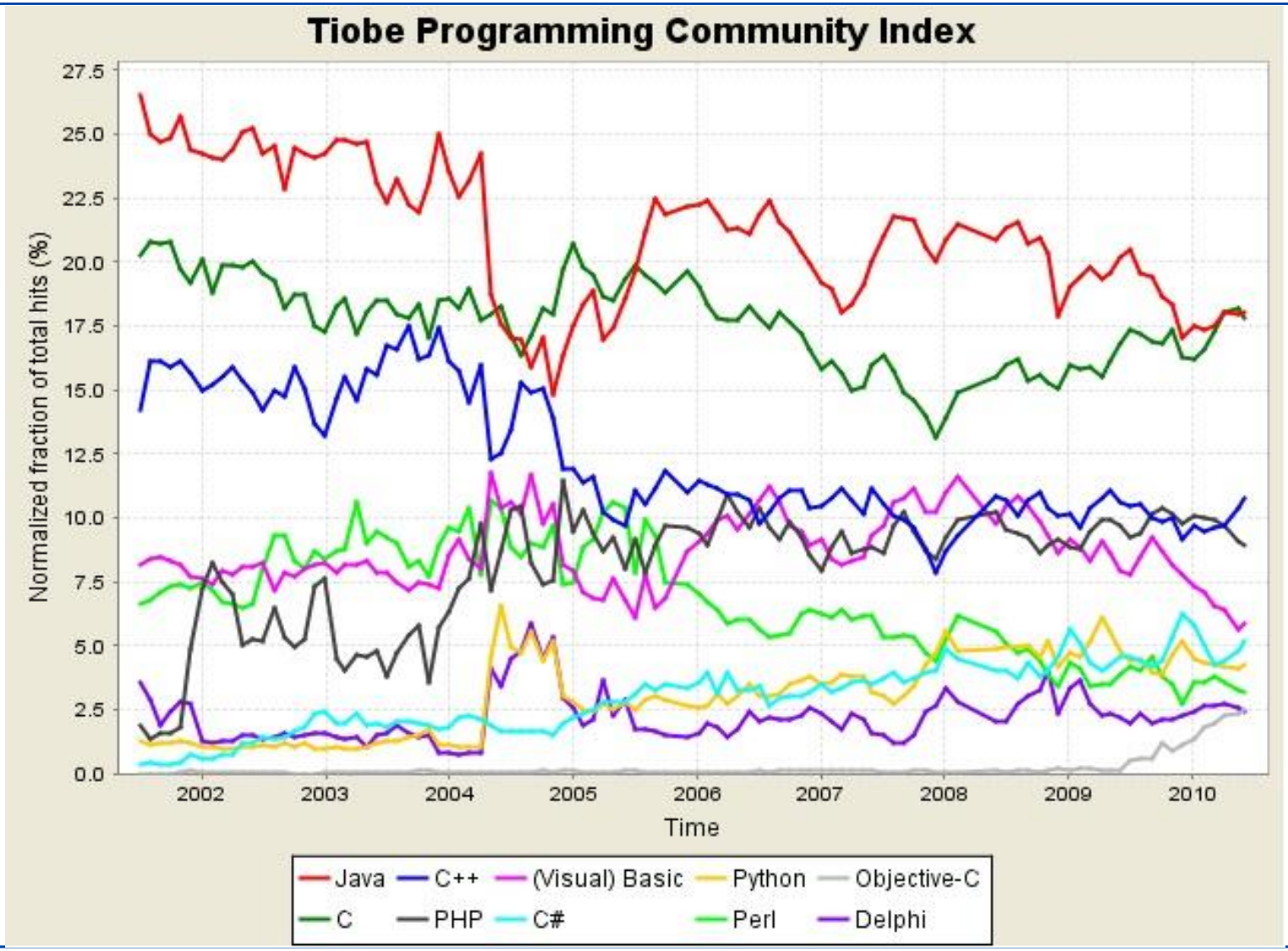
- **Programming Language**
  - Object Oriented
  - syntax derrived from C and C++
  - Java applications compile to bytecode
  - bytecode can be run on any Java Virtual Machine regardless of the platform
  - "write once, run anywhere"
- **Typing**
  - Statically typed
  - Strongly typed
  - type-safe
- **Open Source**
  - GNU Public License - May 2007

Why Use Java Today?

- **Enterprise Development**
  - De facto standard
- **Cross Platform**
- **Open Source Community**
  - Java Community Process
  - Availability of 3rd party libraries
  - There's a library for that
- **Well established and stable**
  - First release in 1995
  - Most widely used programming language today
    - http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html



Why Use Java Today?



Java is not just Java™

- **Java SE (Standard Edition)**
  - Core Java Platform
- **Java EE (Enterprise Edition)**
  - Web Container (Servlets, JSP)
  - EJB
  - Persistence - JPA
  - Web Services - JAX-WS, JAX-RS
- **Java ME (Micro Edition)**
- **Open Source**
  - Apache Foundation
  - Eclipse Foundation
  - SpringSource
  - JBoss

Getting Started - Environment

- **Download the JDK**
- **Choosing an IDE**
  - NetBeans (Open Source, GPL, http://netbeans.org/)
  - Eclipse (Open Source, EPL, http://eclipse.org)
  - IntelliJ IDEA (Community Edition, Commercial, http://www.jetbrains.com/idea/)
- **Plugins**
  - Version Control (Subversion, Git, Mercurial)
  - Building (Maven)
- **Building**
  - Javac - The Java Compiler
  - Apache Ant (http://ant.apache.org/)
  - Apache Maven (http://maven.apache.org/)
  - Buildr (http://buildr.apache.org/), Gradle (http://www.gradle.org/)
- **Dependency Management**
  - Apache Maven and Apache Ivy (http://ant.apache.org/ivy/)
    - Maven Central Repository - http://repo1.maven.org/maven2

Java Developer's Toolbox

- **Java API**
  - JavaDoc - http://java.sun.com/javase/6/docs/api/
  - Core language, collections, IO, net, JDBC, Concurrency, Cryptography
  - Swing, Web Services (JAX-WS), XML Binding (JAX-B) and handling (DOM, SAX, XSLT)
- **Web**
  - EE Specification - Servlets and JSP
  - **Containers** - Apache Tomcat (http://tomcat.apache.org/), Jetty (http://jetty.codehaus.org/jetty/)
  - **Frameworks** - Spring, Wicket, Tapestry, Struts2
  - REST - JAX-RS Specification (Jersey, Restlet, Apache CXF)
- **Persistence (JPA)**
  - Object relational Mapping
  - Hibernate, EclipseLink
- **Dependency Injection**
  - Google Guice, Spring Framework

Getting Started

- **Hello World Application**
  - Everything is a class
  - main method is the entry point
  - HelloWorld.java

```
package com.chrisdail.monctondevcon;

/**
 * This is a comment.
 *
 * @author Chris Dail
 */
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

Java 'Files'

- **.java**
  - Java Source code
- **.class**
  - Java Bytecode (1 .java is compiled to 1 .class)
- **.jar**
  - Archive of many .class files in a single bundle.
  - Mechanism for distributing an application or library
  - Literally a .zip file
- **Manifest.MF**
  - Jar file manifest
  - Describes a library or component
- **.war**
  - Web application archive
  - Similar format to a .jar file but with a web.xml descriptor

The Java Virtual Machine

- **Java Compiler**
  - Compiles `.java` files into Java bytecode
- **Execution environment for compiled Java bytecode**
  - Features JIT compilation and garbage collection
  - Executes `.class` and `.jar` files
- **Source does not need to be Java**
- **Languages ported to the JVM**
  - Rhino (JavaScript), JRuby, Jython
- **Languages designed for the JVM**
  - Groovy, Clojure, Scala

Future of Java

- **On the downtrend**
- **Stagnant core language and syntax**
  - Difficult to change due to market share
  - Slow to change - Reflective of big companies (Oracle and IBM)
- **Java 7**
  - Delays due to JCP politics around moving to Open Source
  - Oracle acquisition of Sun
- **The Java VM**
  - Will likely outlive the Java language itself

Building an Application

- **Demonstration of building a Java application from scratch**
- Using the Eclipse IDE
- Using Maven for dependency management
- Using an open source library (twitter4j)