Introduction

What is it?
- A tool for software development
- Open source project
- Originally from IBM, seriously maintained

Features
- Extendable due to a Plug-in Architecture
- IDE
- Java compiler
- Java debugger
- Team/CVS management
- Many nice features that make programming a lot easier

Sopra 2003
Introduction

- Language-neutral
  - Permit unrestricted content types
  - HTML, Java, C, JSP, EJB, XML, GIF, ...

- Platform-neutral
  - MS Windows XP, 2000, NT, ME, 98SE
  - Linux® on Intel x86 - Motif, GTK
    - RedHat Linux 8.0 x86
    - SuSE Linux 8.1 x86
  - Sun Solaris 8 SPARC – Motif
  - Apple Mac OS® X 10.2 on PowerPC – Carbon
Java Perspective

- Java-centric view of files in Java projects
  - Java elements meaningful for Java programmers
Java Perspective

- Browse type hierarchies
  - "Up" hierarchy to supertypes
  - "Down" hierarchy to subtypes
Java Perspective

- Search for Java elements
  - Declarations or references
  - Including libraries and other projects

Sopra 2003
Hovering over identifier shows Javadoc spec

```java
package com.example.hw;

public class HelloWorld implements Cloneable {
    private static boolean DEBUG = true;

    public static void main(String[] args) {
        System.out.println("Hello world");
    }
}
```

The `System` class contains several useful class fields and methods. It cannot be instantiated. Among the facilities provided by the `System` class are standard input, standard output, and error output streams; access to externally defined "properties"; a means of loading files and libraries; and a utility method for quickly copying a portion of an array.
Method completion in Java editor

- List of plausible methods
- Doc for method

Sets the data of a node.

**Parameters:**
- **key** key of node for which to set data
- **data** new data value for node

**Throws:**
- `java.util.NoSuchElementException` the nodeKey does not exist in the receiver
- `java.lang.IllegalArgumentException` the receiver is immutable
Java Editor

- On-the-fly spell check catches errors early
Java Editor

- Code templates help with drudgery

```java
package com.example.hw;

public class HelloWorld {
    private static boolean DEBUG = true;

    public static void main(String[] args) {
        System.out.println("Hello world");
    }
}
```
Java Editor

- Java editor creates stub methods

Method stub insertion for anonymous inner types

Method stub insertion for inherited methods

Sopra 2003
Java Editor

- Java editor helps programmers write good Java code

Variable name suggestion

```java
public TestResult run() {
    TestResult res = run(result);
    return res;
}
/**
 * Runs the test
 */
public void run()
    result.run();
/**
 * Runs the bare test sequence.
 * @exception
 */
/**
 * Runs the bare test sequence.
 * @throws Exception
 * @throws RuntimeException
 */
public void runBare() throws Exception, RuntimeException {
    setUp();
    assertEquals(expected, actual);
    boolean expected, boolean actual;
```
Team Component

- Version and configuration management (VCM)
- Share resources with team via a repository
- Repository associated at project level
- Extension point for new types of repositories
- Repository provider API and framework
- Eclipse Platform includes CVS repository provider
- Available repository providers*
  - ChangeMan (Serena)
  - ClearCase (Rational)
  - CM Synergy (Telelogic)
  - PVCS (Merant)
  - Microsoft Visual Source Safe
  - AllFusion Harvest (CA)
  - Perforce
  - Source Integrity (MKS)
  - TeamCode (Interwoven)

* March 2003
Eclipse Java Debugger

- Launch configurations
  - How to run a program (debug mode option)
- Generic debug model
  - Standard debug events: suspended, exit, ...
  - Standard debug actions: resume, terminate, step, ...
  - Breakpoints
  - Expressions
  - Source code locator
- Generic debug UI
  - Debug perspective
  - Debug views: stack frames, breakpoints, ...
- Example: JDT supplies Java launcher and debugger
  - Java debugger based on JPDA
- Debug mechanisms available to other plug-ins
Eclipse Java Debugger

- Run or debug Java programs

- Local variables

- Threads and stack frames

- Editor with breakpoint marks

- Console I/O

Sopra 2003
Refactoring

- Refactoring actions rewrite source code
  - Within a single Java source file
  - Across multiple interrelated Java source files
- Refactoring actions preserve program semantics
  - Does not alter what program does
  - Just affects the way it does it

- Encourages exploratory programming
- Encourages higher code quality
  - Makes it easier to rewrite poor code
Refactoring

- Full preview of all ensuing code changes
  - Programmer can veto individual changes