



APIS Informationstechnologien GmbH

16th ESUG Conference 2008

**A Moribund Smalltalk Still Alive and
Kicking**

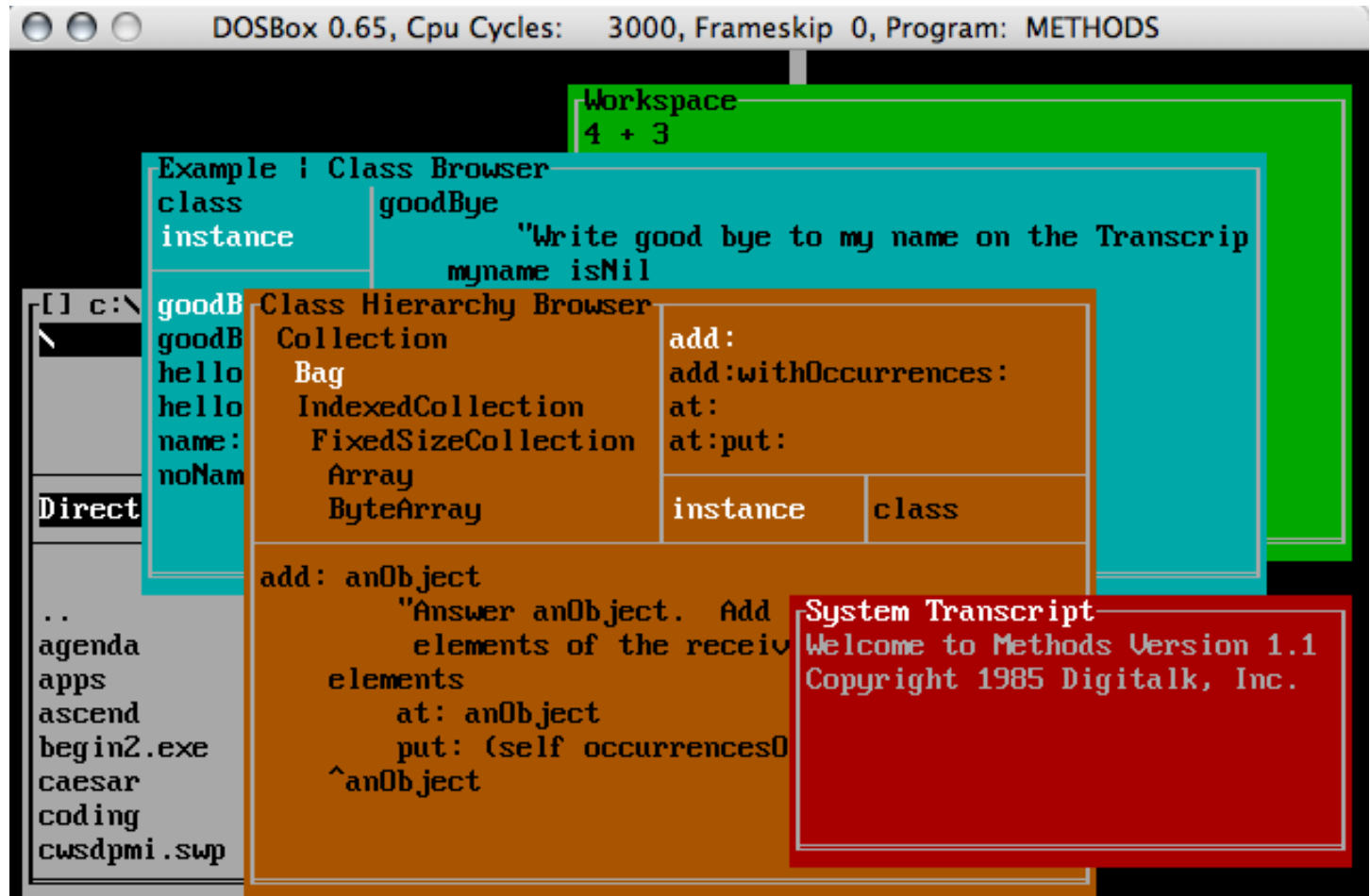
Tom Brey, Heiko Wagner, Jan Kaiser, Andreas Rosenberg

APIS

2008-08-29 / 1

History of Visual Smalltalk (Enterprise)

- 1984 Digitalk published *Methods*



History of Visual Smalltalk (Enterprise)

- 1986 Digitalk published *Smalltalk/V*
- 1992 Smalltalk/V 2.0
- 1995 Visual Smalltalk 3.0
- 1995 ParcPlace Digitalk Merger
- 1996 Visual Smalltalk (Enterprise) 3.1 (Team/V, ENVY no longer supported)

History of Visual Smalltalk (Enterprise)

- 1997 PPD => ObjectShare
- 1999 ObjectShare is dissolved, Cincom buys VisualWorks; VSE rights are sold to Cincom (Support) and Seagull
- Maintenance release VSE 3.2 (aka VSE 2000) by Cincom
- VSEW Mailing List
- 2008 ??? (F.Lesser - LSWST)

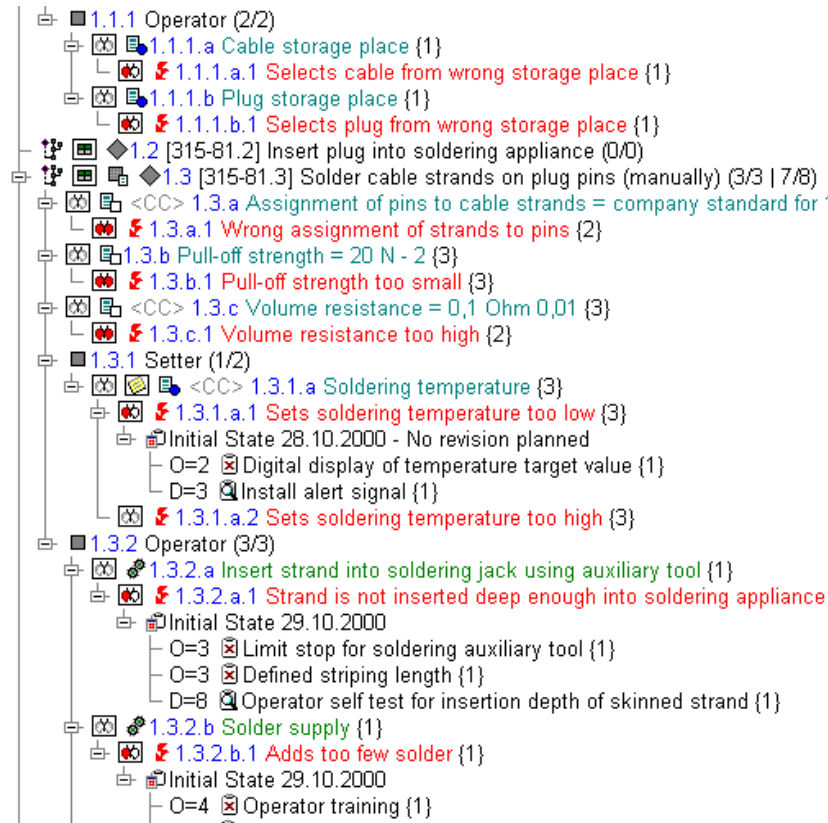
Problems for VSE Users

- Keep VSE (VM + Base) up to date with new OS versions
- No new features
- No Unicode and Multithreading Support
- Integrate customer extensions and add-ons into own modified VSE

Visual Smalltalk at APIS

- APIS started development of Risk & Quality Management Software in 1992 with VS and Envy
- focus is on usability and intelligent User Interfaces

Visual Smalltalk at APIS



(APIS TreeView)

Visual Smalltalk at APIS

Process Flow Diagram: Process Flow Diagram (CC 2042 - Manufacture Connector Cable [Process])

Symbol	Symbol	Symbol	Nr.	Process Name/Operation Description	No.	Product Characteristics	Clas
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="font-size: 2em; font-weight: bold;">APIS</div> <div style="font-size: 3em; border: 2px solid red; border-radius: 50%; padding: 5px; color: red; font-weight: bold;">1</div> </div> <p style="margin: 0;">Informationstechnologien GmbH</p>							
Program: Manufacture Cable				Revision Level: V 1.0		Created: 29.10.2000	
Part number/Latest Change Level: 1 600 376 K47				Revised by: Schranz, Franz, FV-PO; Hehre, Claudia		Modified: 26.06.2006	
Part Description: RG 2042 Connector Cable						Page:	
Symbol	Symbol	Symbol	Nr.	Process Name/Operation Description	No.	Product Characteristics	Clas
□	→	□	1.1	315-81.1 ◆ Prepare workplace	1.1.a	☑ Type of cable	
					1.1.b	☑ Type of plug	
□	↓	□	1.2	315-81.2 ◆ Insert plug into soldering appliance			
□	↓	□	1.3	[soldering iron, soldering gun] 315-81.3 ◆ Solder cable strands on plug pins (manually)	1.3.a	☑ Pull-off strength = 20 N - 2	
					1.3.b	☑ Volume resistance = 0,1 Ohm 0,01	
□	↓	□	1.4	315-81.4 ◆ Solder shielding on plug housing (one sided; manually)	1.4.a	☑ Pull-off strength > 5 mm 0,2	

3 (left sidebar)

2 (cyan area)

1 (red circle)

symbol + name (arrow pointing to 1.1.a)

item code (arrow pointing to 315-81.1)

specification (arrow pointing to 0,2 in 1.4.a)

(APIS TableVision)

Visual Smalltalk at APIS

- VM enhancements and full Unicode-Support
- Other proprietary frameworks:
 - ✓ Object-oriented database,
 - ✓ Parsing/Generating SGML, XML, HTML
 - ✓ TCP and UDP based networking support
 - ✓ SMTP Client, MAPI, ODBC, Registry Support

Development Support

- Refactoring and UnitTest Browser
- CodePane:
 - ✓ Highlighting,
 - ✓ Formatting
 - ✓ Auto-Completion
 - ✓ Direct in-place renaming
 - ✓ Consistency Checks
- Toolbar (e.g. with intelligent class search)
- In progress: Type-System

Development Support

The screenshot shows the Smalltalk IDE interface. The main window is titled "Bag" and contains a menu bar (File, Edit, Smalltalk, Classes, Applications, Methods, Options) and a toolbar. On the left, a class browser shows a tree structure with "Bag" selected under "Collection". The center pane shows the class hierarchy for "Bag", including "FmeaDevelopmentTools", "Kernel", and "KernelTKB". The right pane displays the methods of the "Bag" class: "add:", "add:withOccurrences:", "any", "arithmeticMean", "asBag", "at:", "at:put:", "do:", and "elements". Below the panes, there are radio buttons for "Public" and "Private", and a "Category" dropdown. The main editor area shows the source code for the "add:" method, with a tooltip for the "elements" variable listing possible classes: "OrderedCollection", "OrderedCollectionInspector", "OriginalScreenExtent", "OriginalSysFontSize", and "OrNode". The status bar at the bottom shows the current class is "Bag", the date is "(28.02.1991 01:17:14)", and the user is "Dieter".

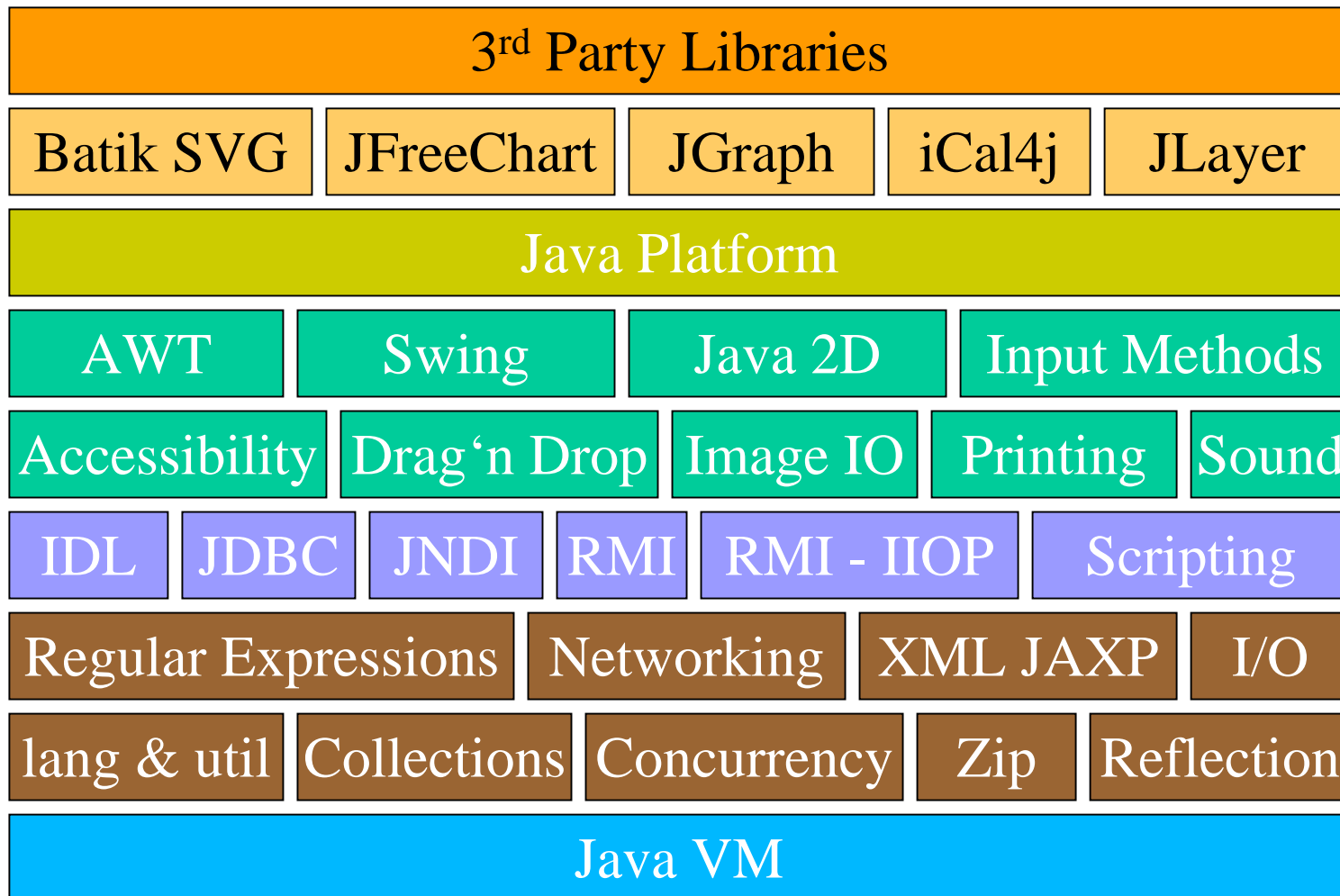
Java Standard Edition 6.0 Integration

- developed independently from JNIPort and JavaConnect
- completely implemented from scratch using proprietary APIS frameworks and VM enhancements
- full IDE integration
- automatic wrapper class generation (JavaClassImportBrowser)
- complete coverage of Java SE 6.0 APIs

Java Integration – Why Java?

- stop reinventing the wheel
- approved and well tested libraries
- completely driven by leading industry standards (JCP)
- open source license (GPL + classpath exception)
- most advanced/optimized multithreaded VM available
- no vendor deadlock (various implementations available)

Java Integration – Java Libraries



Java Integration – Implementation

- interaction with Smalltalk by JNI
- Smalltalk VM augmented with support for IEEE-754 float and 64bit integer types
- Smalltalk can take advantage of Java i18n and l10n features including Unicode support
- full support of Java basic, array and object data types

Java Integration – Tools

- create a Smalltalk SLL from arbitrary any Java Library via automated import (JavaClassImportBroser)
- directly view Java source code in Smalltalk IDE
- compile a Java class within the Smalltalk IDE
- browse Java API documentation (JavaDoc) in Smalltalk IDE

Java Integration – JavaClassImportBrowser

The screenshot displays the JavaClassImportBrowser application. The left pane shows a tree view of classes and interfaces, with the following structure:

- org.omg.PortableServer.DynamicImplementation (Org_omg_PortableServer_Dy)
- org.omg.PortableServer.ServantActivatorPOA (Org_omg_PortableServer_ServantActivatorPOA)
- org.omg.PortableServer.ServantLocatorPOA (Org_omg_PortableServer_ServantLocatorPOA)
- org.omg.PortableServer.ServantActivatorHelper (Org_omg_PortableServer_ServantActivatorHelper)
- org.omg.PortableServer.ServantLocatorHelper (Org_omg_PortableServer_ServantLocatorHelper)
- org.omg.PortableServer.ServantLocatorPackage.CookieHolder (Org_omg_PortableServer_ServantLocatorPackage.CookieHolder)
- org.omg.PortableServer.ServantRetentionPolicyValue (Org_omg_PortableServer_ServantRetentionPolicyValue)
- org.omg.PortableServer.ThreadPolicyValue (Org_omg_PortableServer_ThreadPolicyValue)
- org.w3c.dom.bootstrap.DOMImplementationRegistry (Org_w3c_dom_bootstrap_DOMImplementationRegistry)
- org.xml.sax.HandlerBase (Org_xml_sax_HandlerBase)
- javax.swing.plaf.synth.SynthParser
- org.xml.sax.helpers.AttributeListImpl (Org_xml_sax_helpers_AttributeListImpl)
- org.xml.sax.helpers.AttributesImpl (Org_xml_sax_helpers_AttributesImpl)
- org.xml.sax.ext.Attributes2Impl (Org_xml_sax_ext_Attributes2Impl)
- org.xml.sax.helpers.DefaultHandler (Org_xml_sax_helpers_DefaultHandler)
- org.xml.sax.ext.DefaultHandler2 (Org_xml_sax_ext_DefaultHandler2)
- org.xml.sax.helpers.LocatorImpl (Org_xml_sax_helpers_LocatorImpl)
- org.xml.sax.ext.Locator2Impl (Org_xml_sax_ext_Locator2Impl)
- org.xml.sax.helpers.NamespaceSupport (Org_xml_sax_helpers_NamespaceSupport)
- org.xml.sax.helpers.NewInstance
- org.xml.sax.helpers.ParserAdapter (Org_xml_sax_helpers_ParserAdapter)
- org.xml.sax.helpers.ParserFactory (Org_xml_sax_helpers_ParserFactory)
- org.xml.sax.helpers.XMLFilterImpl (Org_xml_sax_helpers_XMLFilterImpl)
- org.xml.sax.helpers.XMLReaderAdapter (Org_xml_sax_helpers_XMLReaderAdapter)
- org.xml.sax.helpers.XMLReaderFactory (Org_xml_sax_helpers_XMLReaderFactory)
- org.xml.sax.InputSource (Org_xml_sax_InputSource)
- sun.awt.PaintEventDispatcher (Sun_aws_PaintEventDispatcher)
- javax.swing.SwingPaintEventDispatcher
- sun.awt.SunGraphicsCallback (Sun_aws_SunGraphicsCallback)
- java.awt.GraphicsCallback
- sun.security.provider.certpath.CertPathHelper (Sun_security_provider_certpath_CertPathHelper)
- java.security.cert.CertPathHelperImpl
- sun.swing.DefaultLookup (Sun_swing_DefaultLookup)
- javax.swing.plaf.synth.SynthDefaultLookup
- Interface
 - java.applet.AppletContext (Java_applet_AppletContext)
 - java.applet.AppletStub (Java_applet_AppletStub)
 - java.applet.AudioClip (Java_applet_AudioClip)
 - java.awt.ActiveEvent (Java_aws_ActiveEvent)

The right pane shows the code for the selected class, `getDOMImplementation: aJString`:

```
getDOMImplementation: aJString
| jvm cls mID params res |

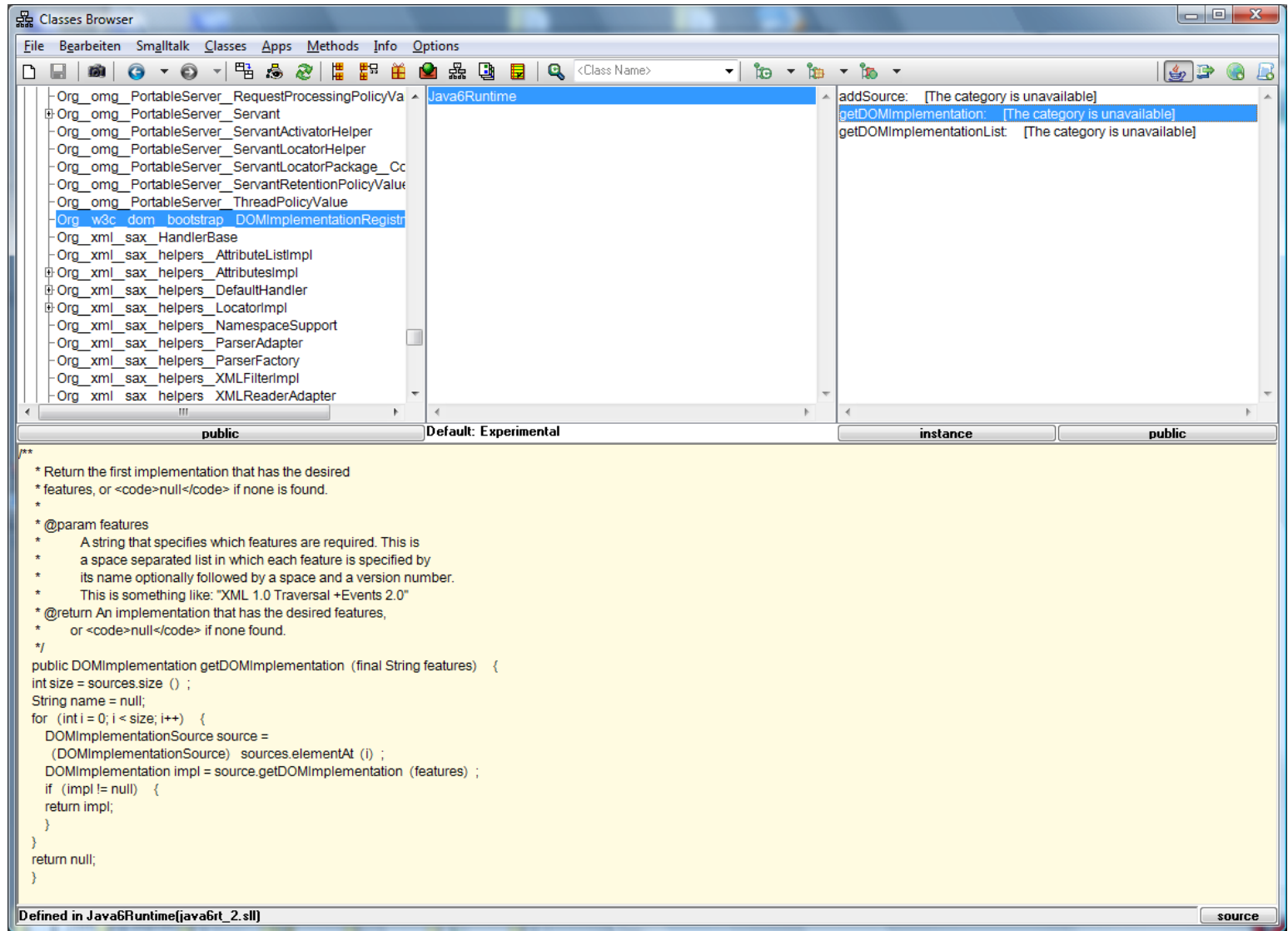
"-- Code auto generated by JavaUtilities --"
jvm := JavaVM current.
cls := self getClass.
params := Array new: 1.
params at: 1 put: aJString.
mID := cls getMethodID: 'getDOMImplementation' signature: '(Ljava/lang/String;Lorg/w3c/dom/DOMImplementation;Ljava/lang/String;)Ljava/lang/String;'.
res := jvm callObjectMethod: mID instance: self params: params.
jvm handleException.
^Org_w3c_dom_DOMImplementation fromJNIReference: res vm: jvm
```

The bottom pane shows the source code for the selected class, `addSource(Lorg/w3c/dom/DOMImplementationSource;J)V`:

```
addSource(Lorg/w3c/dom/DOMImplementationSource;J)V
getDOMImplementation(Ljava/lang/String;Lorg/w3c/dom/DOMImplementationSource;J)Ljava/lang/String;
getDOMImplementationList(Ljava/lang/String;Lorg/w3c/dom/DOMImplementationList;J)Ljava/lang/String;
static newInstance(Lorg/w3c/dom/bootstrap/DOMImplementationRegistry;J)Lorg/w3c/dom/bootstrap/DOMImplementationRegistry;
static final java.lang.String PROPERTY
```

The bottom of the application has radio buttons for filtering the view: Class, Instance, All, Methods, Fields, All.

Java Integration – Java Source

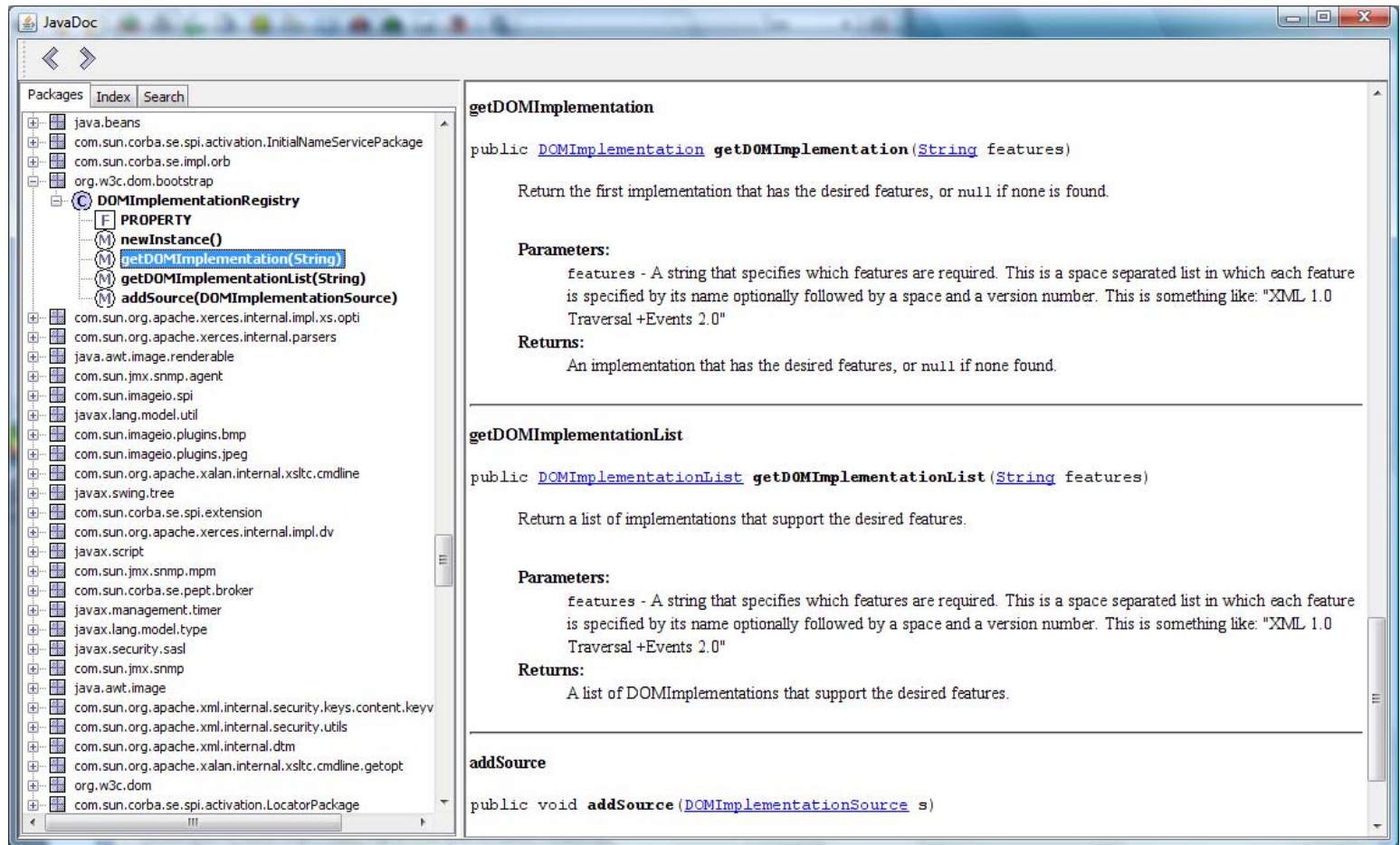


The screenshot displays the Eclipse IDE's 'Classes Browser' window. The left pane shows a tree view of classes, with 'Java6Runtime' selected. The right pane shows the source code for the 'getDOMImplementation' method. The code includes a Javadoc comment and a Java implementation that iterates through a list of sources to find the first one matching the specified features.

```
public DOMImplementation getDOMImplementation (final String features) {
    int size = sources.size ();
    String name = null;
    for (int i = 0; i < size; i++) {
        DOMImplementationSource source =
            (DOMImplementationSource) sources.elementAt (i) ;
        DOMImplementation impl = source.getDOMImplementation (features) ;
        if (impl != null) {
            return impl;
        }
    }
    return null;
}
```

Defined in Java6Runtime[java6rt_2.sll] [source](#)

Java Integration – JavaDoc



The screenshot shows the JavaDoc application interface. On the left, a tree view displays the package structure, with `DOMImplementationRegistry` selected. The right pane shows the documentation for `getDOMImplementation` and `getDOMImplementationList` methods.

getDOMImplementation

```
public DOMImplementation getDOMImplementation(String features)
```

Return the first implementation that has the desired features, or null if none is found.

Parameters:

- `features` - A string that specifies which features are required. This is a space separated list in which each feature is specified by its name optionally followed by a space and a version number. This is something like: "XML 1.0 Traversal +Events 2.0"

Returns:

An implementation that has the desired features, or null if none found.

getDOMImplementationList

```
public DOMImplementationList getDOMImplementationList(String features)
```

Return a list of implementations that support the desired features.

Parameters:

- `features` - A string that specifies which features are required. This is a space separated list in which each feature is specified by its name optionally followed by a space and a version number. This is something like: "XML 1.0 Traversal +Events 2.0"

Returns:

A list of DOMImplementations that support the desired features.

addSource

```
public void addSource(DOMImplementationSource s)
```