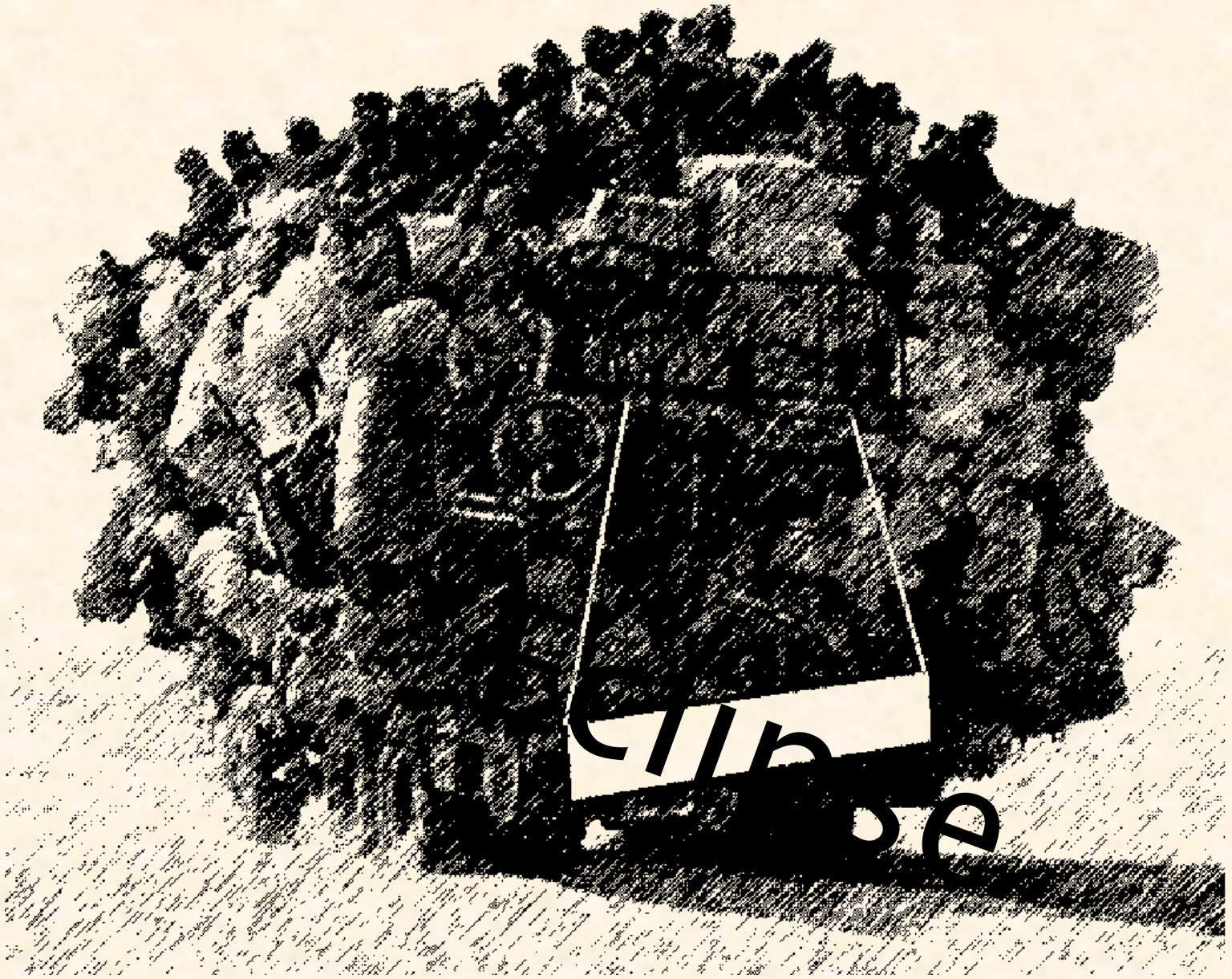
Eclipse
Performance
Chris Laffra
Rational Performance Engineering Team

Introduction

"Considering the current sad state of our computer programs, software development
is clearly still a black
art, and cannot yet
be called an
engineering discipline."

-- Bill Clinton

The Problem



Too Slow…

Too Big…

"Wirth's Law: Software gets slower faster than hardware gets faster." -- Nicklaus Wirth

"What Andy giveth', Bill taketh' away"

"What Bill gives, Gossling, Bray & Gamma take away"

“ The nice thing about standards is that there are so many of them to choose from.”

-- Andrew S. Tanenbaum

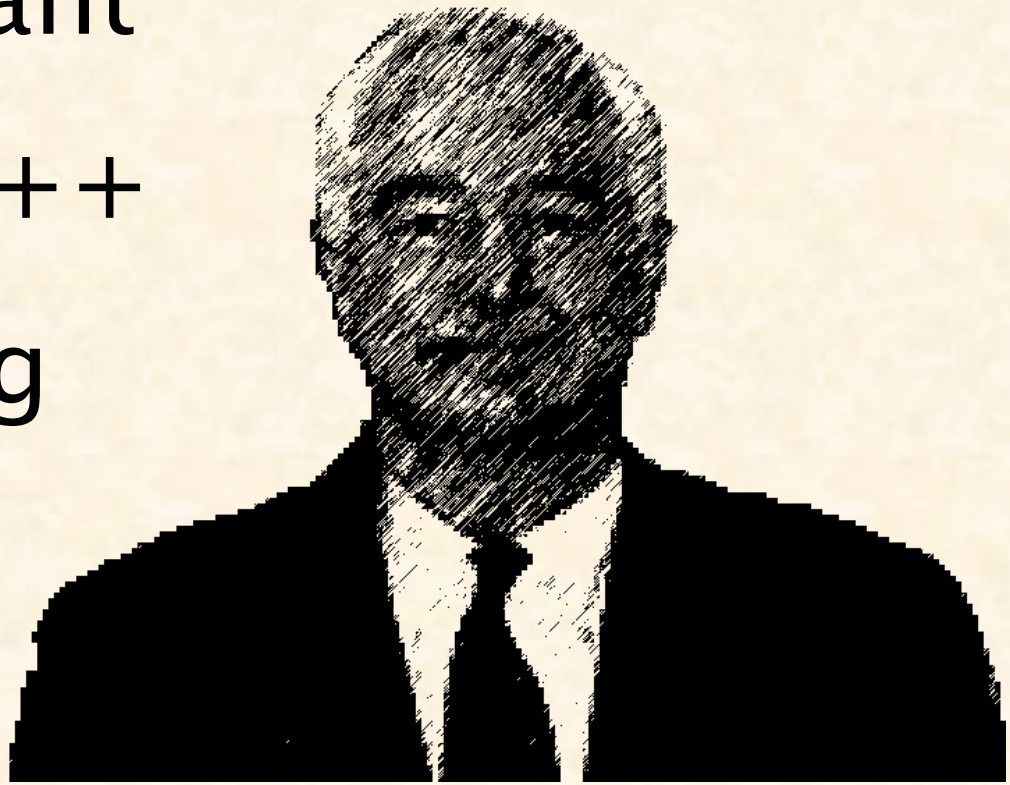
"I invented the term Object-Oriented, and I can tell you I did not have C++ in mind."

-- Alan Kay

"I have found that humans often use Smalltalk during awkward moments."

-- Data

"Java: the elegant simplicity of C++ and the blazing speed of Smalltalk."



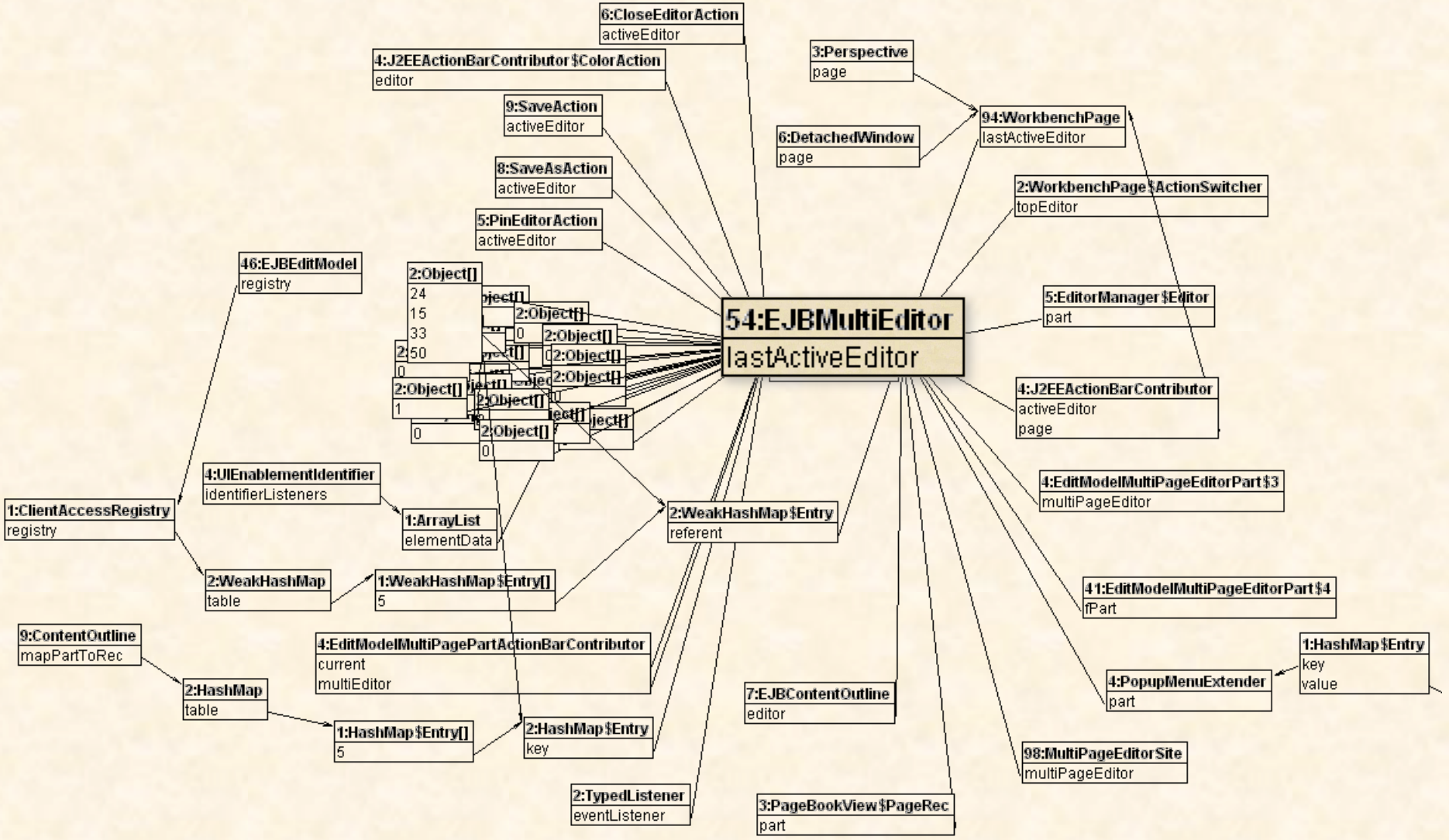
-- Roland Turner

Eclipse == OS

- Eclipse manages resources:
  - Workspaces, Projects, Programs, Files
  - UI Real Estate, Views, Editors, Menus
  - Version Management, Team Sharing
  - OS resources (File, GDI handles)
  - CPU (Threads, Competition)
  - Memory (Lots of it, GC)

Complexity

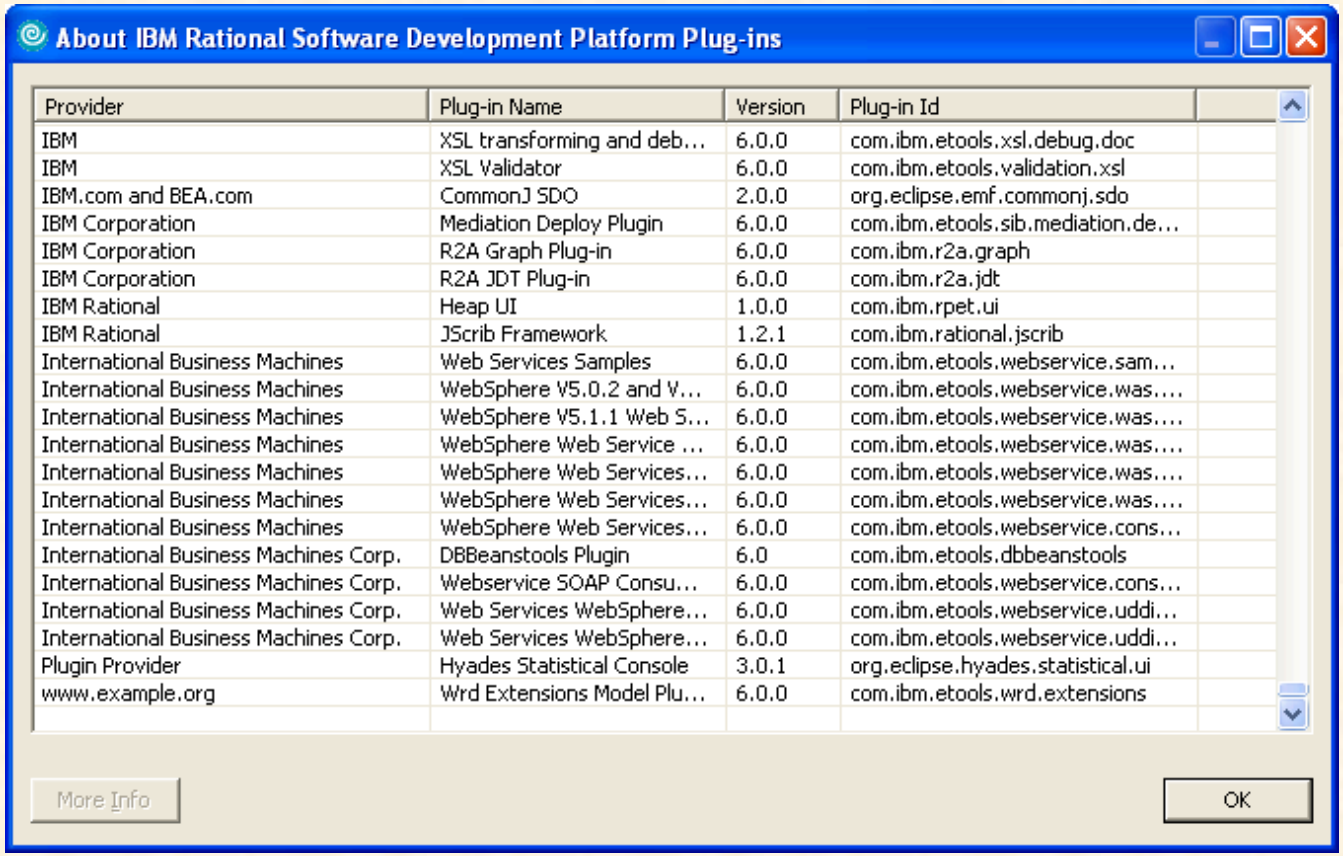
Links to Current Editor



Numerology

- RAD workspace with 2 EJB projects:
  - Build takes 150 million method calls
  - Stacks are 300 calls deep
  - Things run for up to 30 minutes
  - Profiling tools fall over
- Without effort, RSA grows to 600MB
- Some customers need 2.4GB memory

Value is in Composition



"There are two ways of constructing a software design:

1. Make it so simple that there are obviously no deficiencies, or
2. Make it so complicated that there are no obvious deficiencies."

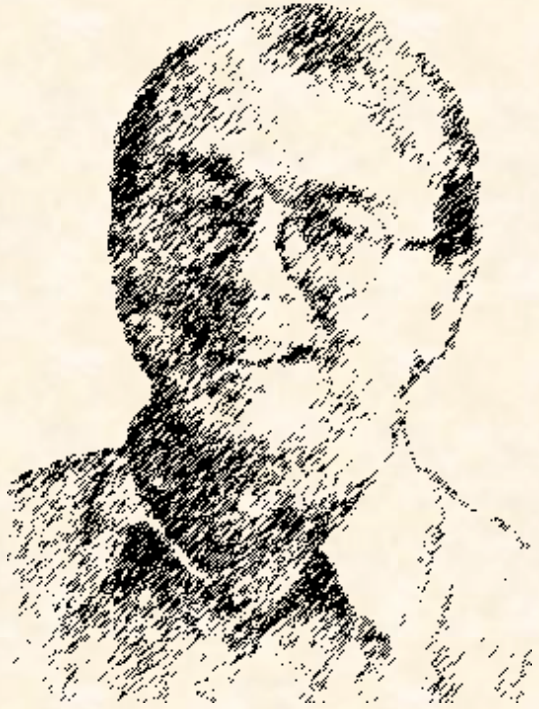
-- Tony Hoare

"I fear the new object-oriented systems may suffer the fate of LISP, in that they can do many things,
but the complexity
of the class hierarchies
may cause them to
collapse under their
own weight."

Bill Joy

Dealing with complexity

"Lloyd's Hypothesis: Everything worth understanding about a complex system, can be understood in terms of how it processes information."



-- Seth Lloyd

"Today, most software exists, not to solve a problem, but to interface with other software."

-- I. O. Angell

"The secrets to weight loss are to watch your diet, to exercise, to set reasonable goals, and to measure regularly."

-- Oprah Winfrey

[this is not a real quote]

"A specification, design, procedure, or test plan that will not fit on one page of 8.5-by-11 inch paper cannot be understood."

-- Mark Ardis

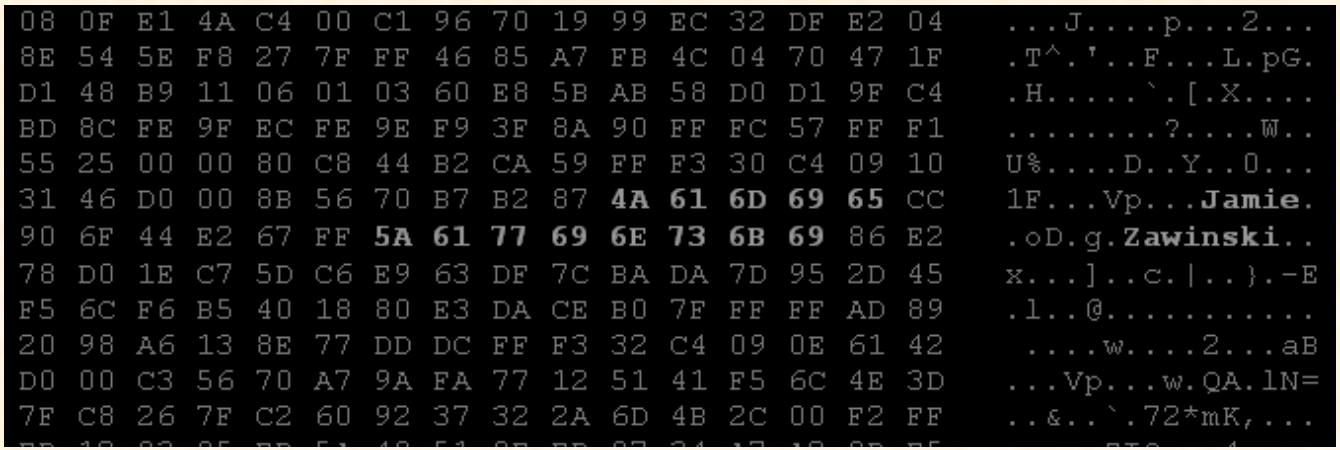
Design for Performance

- Define and enforce budgets for:
  - – CPU consumption
  - – Memory consumption
- Design for caching (a very important space/time tradeoff)
- Design for persistence life-cycle
- Design for scalability

Why do we write such big software?

"Zawinski's Law: Every program attempts to expand until it can read mail. Those programs which cannot so expand are replaced by ones which can."

-- Jamie Zawinski



"Any technology which is distinguishable from magic is insufficiently advanced."

-- Gregory Benford

"The way to succeed is to double your error rate.“

-- Thomas J. Watson

The Most Innovative Company

Computable
www.computable.nl

	Bedrijf	Score
1	IBM	1.003
2	Microsoft	865
3	Oracle	786
4	Hewlett-Packard	710
5	Cisco Systems	530
6	SAP	502
7	CMG	494
8	Cap Gemini	443
9	Compaq	433
10	Atos Origin	422
11	Dell	421
12	Philips	375
13	PinkRoccade	372
14	Sun Microsystems	346
15	Getronics	331
16	KPN	324
17	Nokia	294

IBM #1 in 2002 survey:

- Functionality (24%)
- Innovative Solutions (23%)
- Skilled People (18%)
- Flexible Solutions (13%)

“Effective Solution Provider”

What to do?

"Perfection is achieved not when there is nothing more to add, but rather when there is nothing more to take away."

-- Antoine de Saint-Exupery

SE
E
S
000
000

Games

Treat our software likes games

1. No manual
2. Need to “unlock” features
3. Real-time based (rendering, etc.)
4. AI built in
5. Maximize use of hardware

Splash Screen Design

"With all the plug-ins, WSAD can take 30 seconds to start up. I stare at the splash screen a lot."

http://blog.vikdavid.com/2005/05/whats_with_the_.html

Proposal mock-up – not an actual program

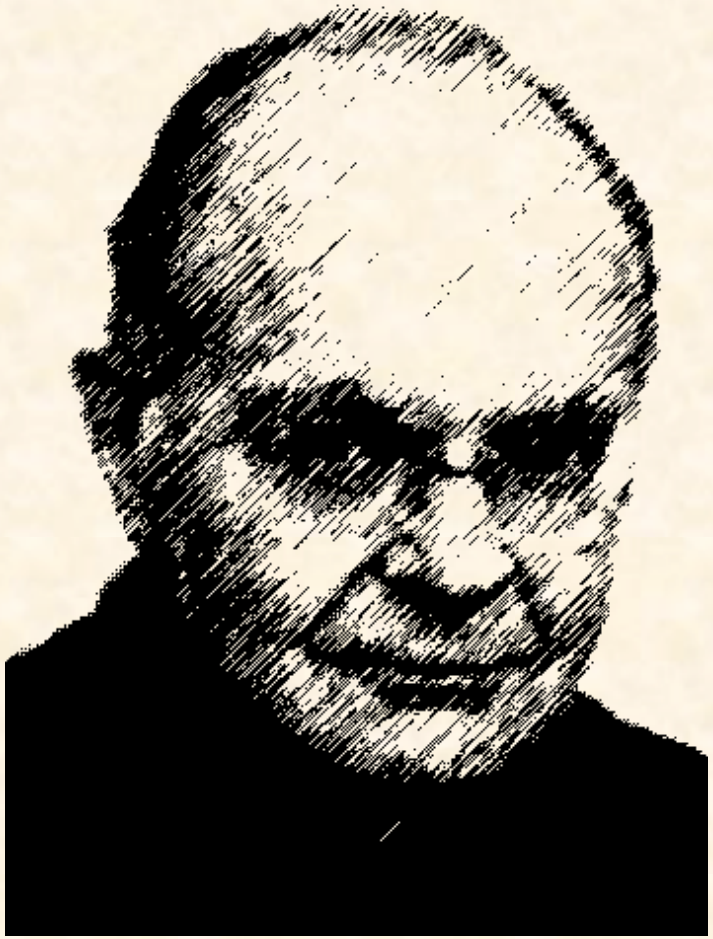
Rational. Software Development Platform
Version 6.0 Trial

Powered by Eclipse Technology

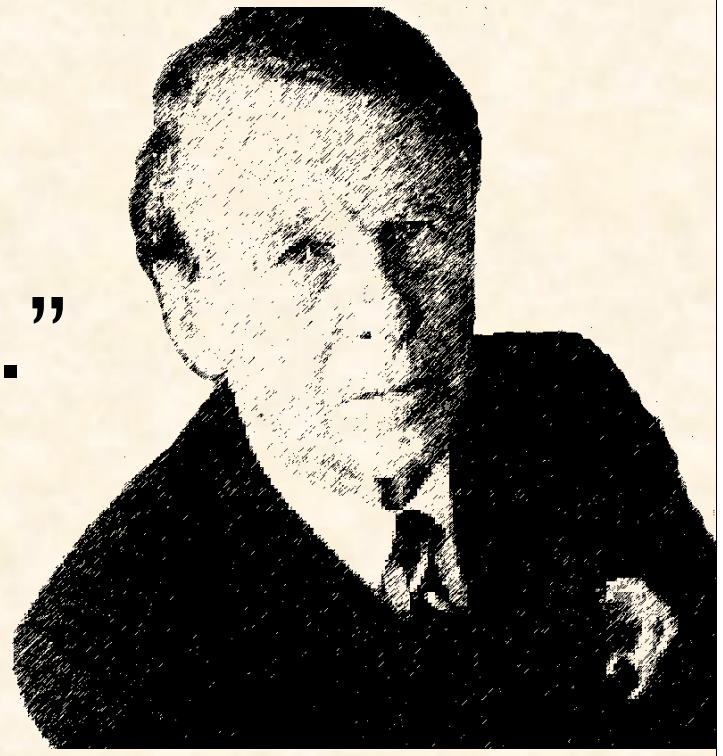
Our Approach?

"Premature optimization is the root of all evil."

--Donald Knuth



“Never stop testing,
and your products
never stop improving.”



-- David Ogilvy

A Practical Approach

- Fix stuff that has a real impact:
  - – Things that are really slow
  - – Things that are used all the time
  - – Key problem reports
- Pay attention to memory leaks
- Keep an eye out for regressions

"The best way to have a good idea is to have many of them."

-- Linus Pauling

"Successful visionaries start from where they are, what they have, and what their customers have."

-- Tom Gilb

Analyzing Performance

The 3 Step Approach

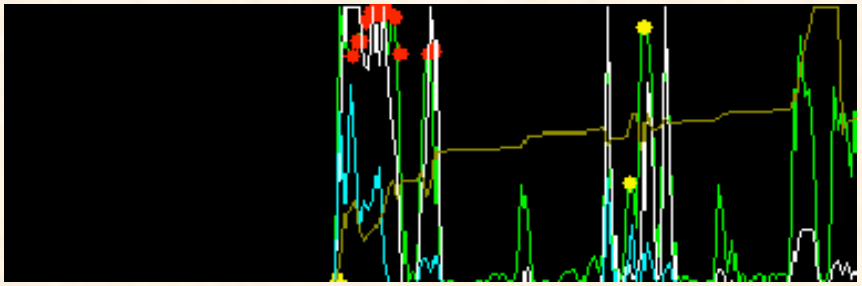
1. Run profiler →
   find bottleneck → fix
2. Run profiler →
   find bottleneck → fix
3. Run profiler…

Which Profiler?

- Pick one. Any is better than none
- Some choices:
  1. YourKit, JProbe, Purify/Quantify
  2. XRay (just moved to Eclipse TPTP)
  3. System.out.println
  4. Eclipse instrumentation

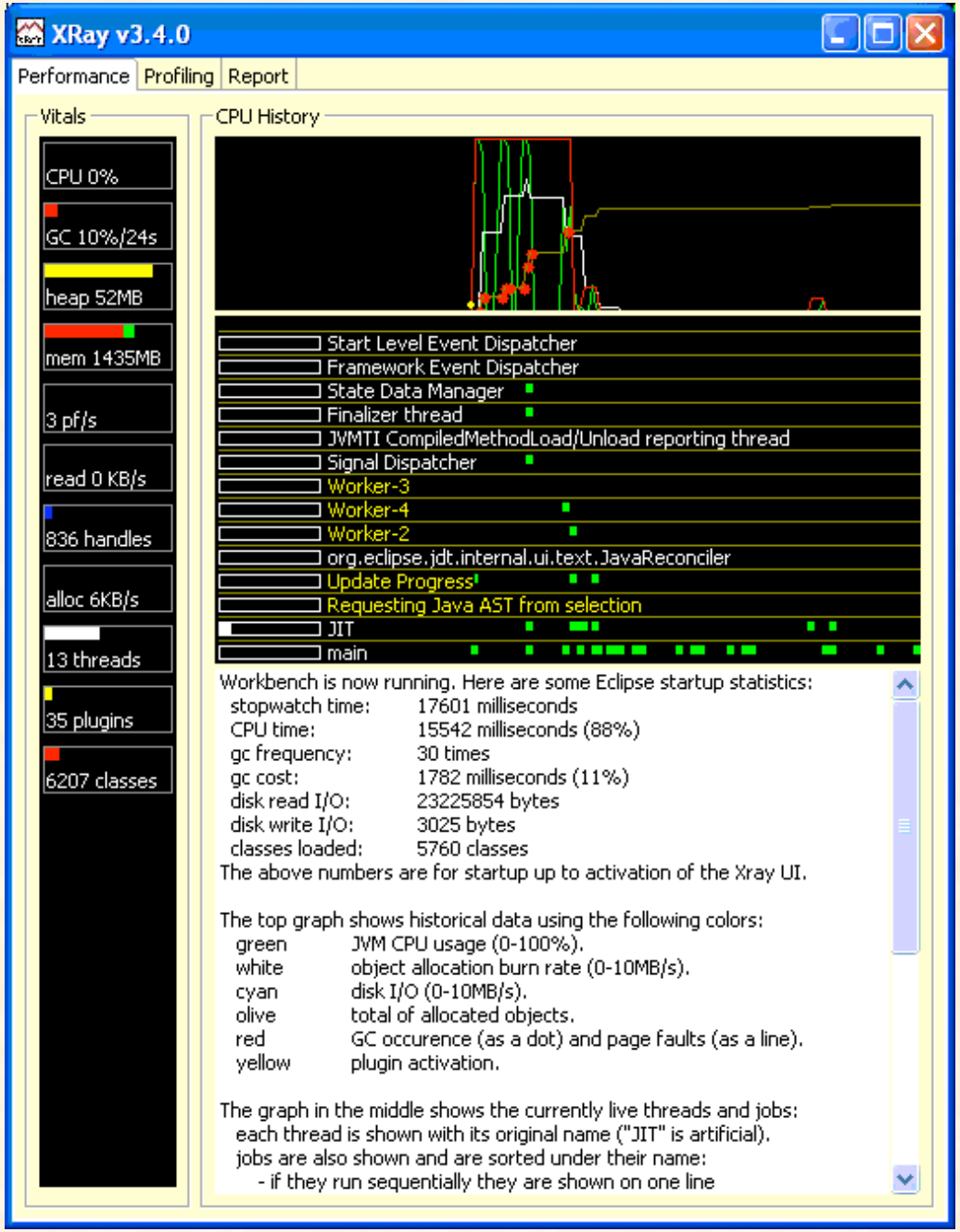
XRay

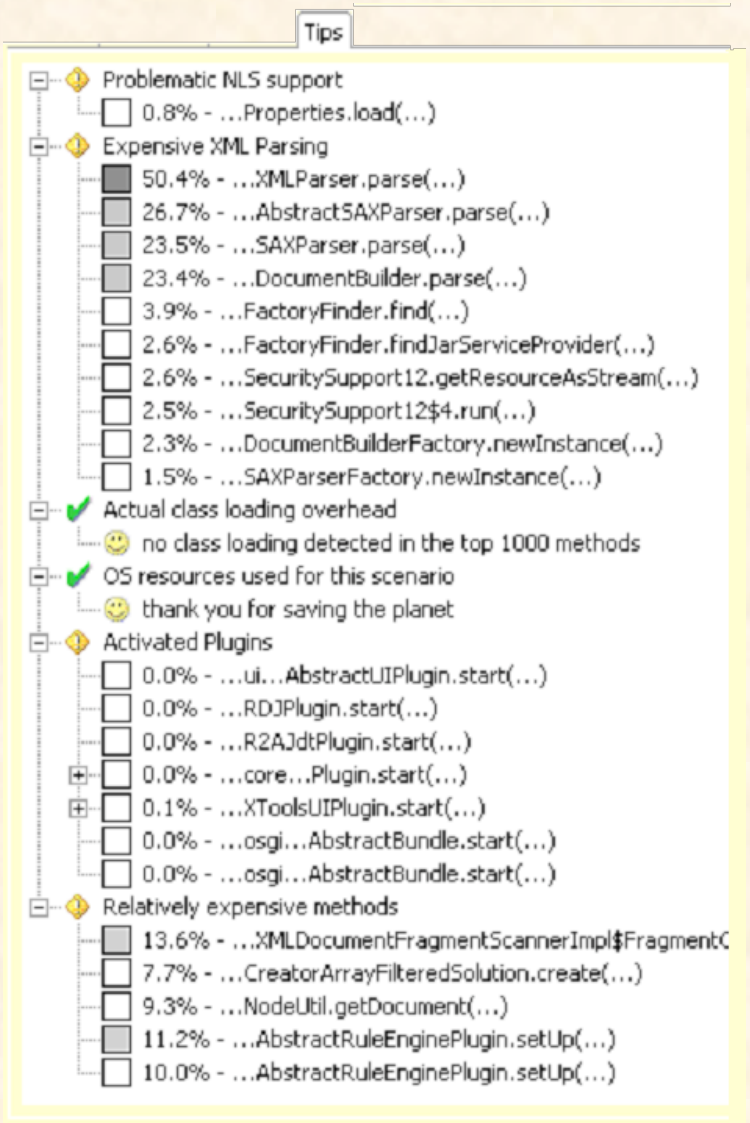
- Xray is modeled after “Task Manager”
- Minimal overhead

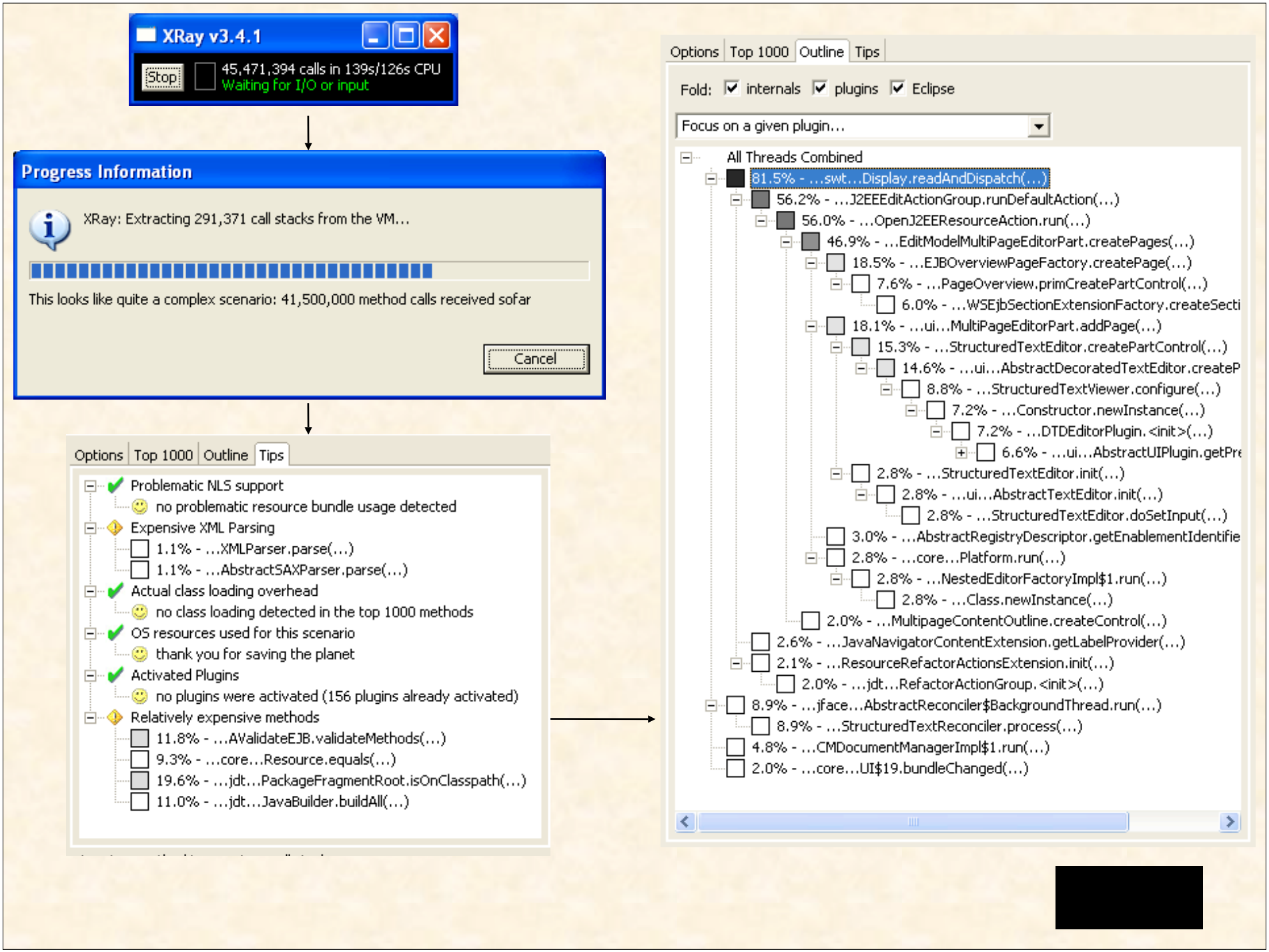


- Use one single graph & human pattern recognition
- Reports with screendumps

XRay


XRay v3.4.0
Performance
Profiling
Report
Vitals
CPU 0%
GC 10%/24s
heap 52MB
mem 1435MB
3 pf/s
read 0 KB/s
836 handles
alloc 6KB/s
13 threads
35 plugins
6207 classes
CPU History
Start Level Event Dispatcher
Framework Event Dispatcher
State Data Manager
Finalizer thread
JVMTI CompiledMethodLoad/Unload reporting thread
Signal Dispatcher
Worker-3
Worker-4
Worker-2
org.eclipse.jdt.internal.ui.text.JavaReconciler
Update Progress
Requesting Java AST from selection
JIT
main
Workbench is now running. Here are some Eclipse startup statistics:
stopwatch time: 17601 milliseconds
CPU time: 15542 milliseconds (88%)
gc frequency: 30 times
gc cost: 1782 milliseconds (11%)
disk read I/O: 23225854 bytes
disk write I/O: 3025 bytes
classes loaded: 5760 classes
The above numbers are for startup up to activation of the Xray UI.
The top graph shows historical data using the following colors:
green JVM CPU usage (0-100%).
white object allocation burn rate (0-10MB/s).
cyan disk I/O (0-10MB/s).
olive total of allocated objects.
red GC occurence (as a dot) and page faults (as a line).
yellow plugin activation.
The graph in the middle shows the currently live threads and jobs:
each thread is shown with its original name ("JIT" is artificial).
jobs are also shown and are sorted under their name:
- if they run sequentially they are shown on one line


Tips
Problematic NLS support
0.8% - ...Properties.load(...)
Expensive XML Parsing
50.4% - ...XMLParser.parse(...)
26.7% - ...AbstractSAXParser.parse(...)
23.5% - ...SAXParser.parse(...)
23.4% - ...DocumentBuilder.parse(...)
3.9% - ...FactoryFinder.find(...)
2.6% - ...FactoryFinder.findJarServiceProvider(...)
2.6% - ...SecuritySupport12.getResourceAsStream(...)
2.5% - ...SecuritySupport12$4.run(...)
2.3% - ...DocumentBuilderFactory.newInstance(...)
1.5% - ...SAXParserFactory.newInstance(...)
Actual class loading overhead
no class loading detected in the top 1000 methods
OS resources used for this scenario
thank you for saving the planet
Activated Plugins
0.0% - ...ui...AbstractUIPlugin.start(...)
0.0% - ...RDJPlugin.start(...)
0.0% - ...R2AJdtPlugin.start(...)
0.0% - ...core...Plugin.start(...)
0.1% - ...XToolsUIPlugin.start(...)
0.0% - ...osgi...AbstractBundle.start(...)
0.0% - ...osgi...AbstractBundle.start(...)
Relatively expensive methods
13.6% - ...XMLDocumentFragmentScannerImpl$FragmentC
7.7% - ...CreatorArrayFilteredSolution.create(...)
9.3% - ...NodeUtil.getDocument(...)
11.2% - ...AbstractRuleEnginePlugin.setUp(...)
10.0% - ...AbstractRuleEnginePlugin.setUp(...)


XRay v3.4.1
Stop
45,471,394 calls in 139s/126s CPU
Waiting for I/O or input
Progress Information
XRay: Extracting 291,371 call stacks from the VM...
This looks like quite a complex scenario: 41,500,000 method calls received sofar
Cancel
Options
Top 1000
Outline
Tips
Problematic NLS support
no problematic resource bundle usage detected
Expensive XML Parsing
1.1% - ...XMLParser.parse(...)
1.1% - ...AbstractSAXParser.parse(...)
Actual class loading overhead
no class loading detected in the top 1000 methods
OS resources used for this scenario
thank you for saving the planet
Activated Plugins
no plugins were activated (156 plugins already activated)
Relatively expensive methods
11.8% - ...AValidateEJB.validateMethods(...)
9.3% - ...core...Resource.equals(...)
19.6% - ...jdt...PackageFragmentRoot.isOnClasspath(...)
11.0% - ...jdt...JavaBuilder.buildAll(...)
Options
Top 1000
Outline
Tips
Fold:
internals
plugins
Eclipse
Focus on a given plugin...
All Threads Combined
81.5% - ...swt...Display.readAndDispatch(...)
56.2% - ...J2EEEditActionGroup.runDefaultAction(...)
56.0% - ...OpenJ2EEResourceAction.run(...)
46.9% - ...EditModelMultiPageEditorPart.createPages(...)
18.5% - ...EJBOverviewPageFactory.createPage(...)
7.6% - ...PageOverview.primCreatePartControl(...)
6.0% - ...WSEjbSectionExtensionFactory.createSecti
18.1% - ...ui...MultiPageEditorPart.addPage(...)
15.3% - ...StructuredTextEditor.createPartControl(...)
14.6% - ...ui...AbstractDecoratedTextEditor.createP
8.8% - ...StructuredTextViewer.configure(...)
7.2% - ...Constructor.newInstance(...)
7.2% - ...DTDEditorPlugin.<init>(...)
6.6% - ...ui...AbstractUIPlugin.getPre
2.8% - ...StructuredTextEditor.init(...)
2.8% - ...ui...AbstractTextEditor.init(...)
2.8% - ...StructuredTextEditor.doSetInput(...)
3.0% - ...AbstractRegistryDescriptor.getEnablementIdentifie
2.8% - ...core...Platform.run(...)
2.8% - ...NestedEditorFactoryImpl$1.run(...)
2.8% - ...Class.newInstance(...)
2.0% - ...MultipageContentOutline.createControl(...)
2.6% - ...JavaNavigatorContentExtension.getLabelProvider(...)
2.1% - ...ResourceRefactorActionsExtension.init(...)
2.0% - ...jdt...RefactorActionGroup.<init>(...)
8.9% - ...jface...AbstractReconciler$BackgroundThread.run(...)
8.9% - ...StructuredTextReconciler.process(...)
4.8% - ...CMDocumentManagerImpl$1.run(...)
2.0% - ...core...UI$19.bundleChanged(...)

"Simplicity and elegance are unpopular because they require hard work and discipline to achieve
and education
to be appreciated."

-- Edsger Dijkstra

Thanks.