



# OpenJDK & IcedTea

Pavel Tišnovský

Quality Assurance Eng.,  
Red Hat

# OpenJDK



# OpenJDK

- Open-source implementation of Java Platform SE
  - Compiler and other development tools
  - JRE
  - Libraries + JavaDoc (rt.jar etc.)
- Packages are now available for many systems
  - RHEL 5
  - Fedora 9, 10, 11, Rawhide...
  - Ubuntu  $\geq$  8.04
  - openSUSE
  - Debian
  - OpenSolaris



# History of OpenJDK and IcedTea

- November 2006
  - HotSpot VM and compiler released under GPL
  - **Not fully buildable JDK yet**
- May 2007
  - Class library released under GPL
  - ~ 96 % of functionality
  - ~ 4 % licensed by Sun from 3rd parties
- June 2007
  - IcedTea project launched
- November 2007
  - Red Hat announced agreement with Sun Microsystems (TCK license agreement etc.)
  - Porters group was created - BSD port, Mac OS X port...

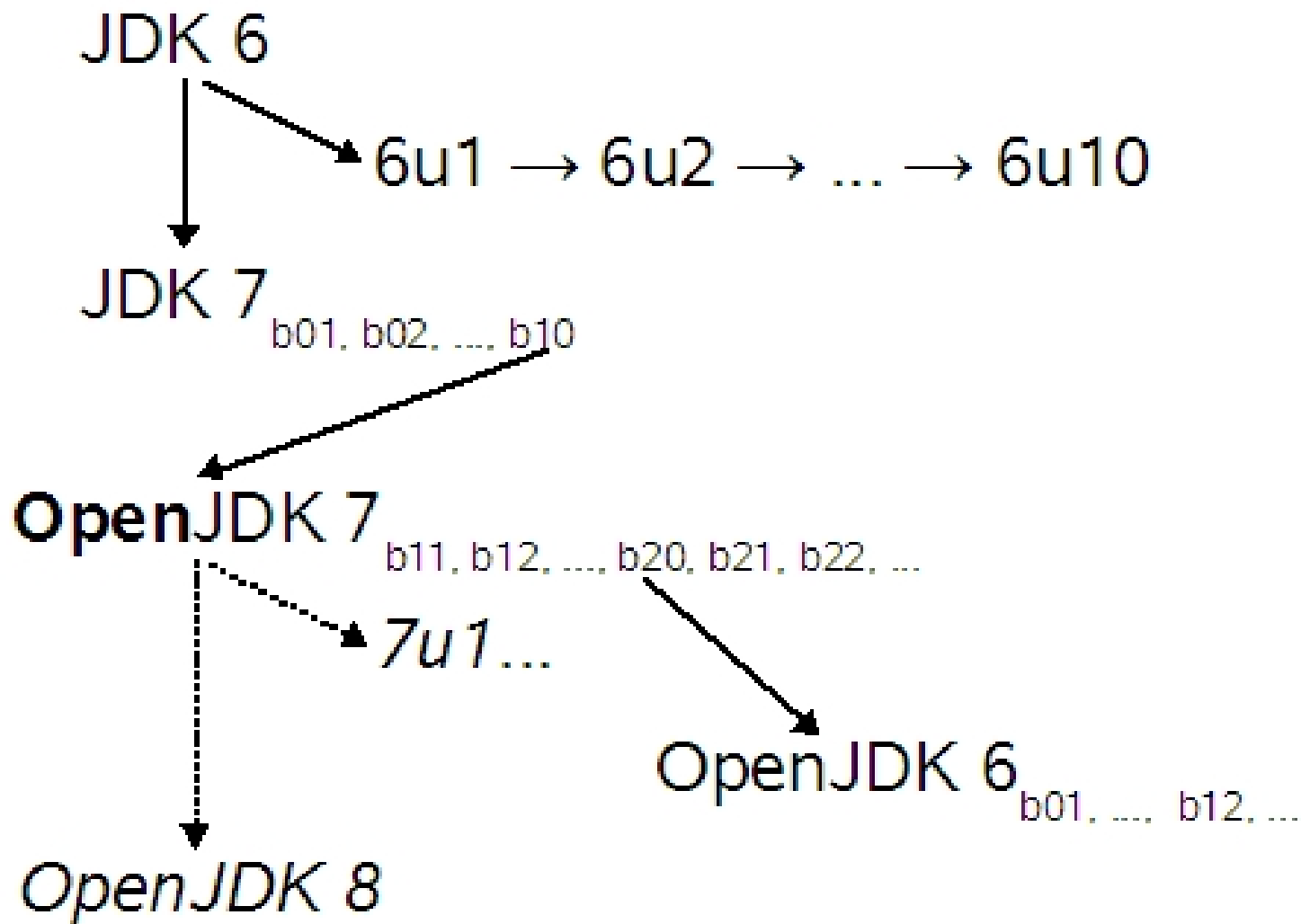


# History

- December 2007
  - Revision control moved from TeamWare to Mercurial
- May 2008
  - OpenJDK included in Fedora 9 and Ubuntu 8.04
- June 2008
  - **IcedTea6 passed TCK tests - fully compatible Java 6 implementation**
- July 2008
  - Debian accepted OpenJDK-6 in unstable
- August 2008
  - OpenJDK 7 on Mac OS X and \*BSD systems



# OpenJDK genealogy



# IcedTea



# “True” Open Source Java

- Started by Red Hat
- Team was working on gcj / classpath
- Sun released OpenJDK based on Java 1.7 (May 2007)
- Mostly free but with around 4% encumbered classes
- IcedTea announcement June 7, 2007
- Respect the OpenJDK trademark
- Built with exclusively free software and tools



≠

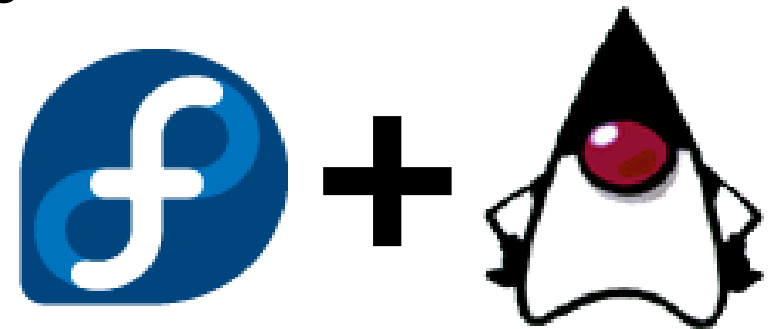
OpenJDK





# IcedTea Project

- Build with exclusively free software and tools
- Provide development infrastructure
  - Mercurial repo
  - Wiki
  - Mailing lists
  - Bug database
  - Code drop hosting
- Replace encumbrances with Free Software
- Porting
- ***Goal: IcedTea in Fedora***
- ***Now Fedora and other distros uses IcedTea6***



# IcedTea in Ubuntu

```
Actions Undo Package Resolver Search Options Views Help
C-T: Menu ?: Help q: Quit u: Update g: Download/Install/Remove Pkgs
aptitude 0.4.11.11 Will free 7803kB of disk space
i openjdk-6-jdk 6b14-1.4.1 6b14-1.4.1
i pkg-config 0.22-1 0.22-1
i python-debian 0.1.12ubun 0.1.12ubun
--- universe - Unsupported Free Software. (1)
--- doc - Documentation and specialized programs for viewing documentation (11)
--\ editors - Text editors and word processors (24)
--\ main - Fully supported Free Software. (23)
i ed 0.7-3ubunt 0.7-3ubunt
i nano 2.0.9-2 2.0.9-2
i openoffice.org 1:3.0.1-9u 1:3.0.1-9u
OpenJDK Development Kit (JDK)
OpenJDK is a development environment for building applications, applets, and
components using the Java programming language.

The packages are built using the IcedTea build support and patches from the
IcedTea project.
Homepage: http://openjdk.java.net/
```



# IcedTea in Ubuntu

```
tisnik@bender:~$ java -version
java version "1.6.0_0"
OpenJDK Runtime Environment (IcedTea6 1.4.1) (6b14-1.4.1-0ubuntu11)
OpenJDK Client VM (build 14.0-b08, mixed mode, sharing)
tisnik@bender:~$ █
```



# IcedTea - compiling

- Default Build requirements (F10)
  - autoconf automake gcc gcc-c++ ant cups-devel xorg-x11-proto-devel libjpeg-devel giflib-devel libXt-devel libXp-devel libXpm-devel libXext-devel libXinerama-devel libXrender-devel libpng-devel libXtst-devel freetype-devel alsa-lib-devel xulrunner-devel-unstable gtk2-devel patch xalan-j2 wget
- Only open-source packages & development tools!
- More info
  - [http://icedtea.classpath.org/wiki/Main\\_Page](http://icedtea.classpath.org/wiki/Main_Page)



# IcedTea - compiling SDK & JRE

- `hg clone`  
<http://icedtea.classpath.org/hg/release/icedtea6>
- `cd icedtea6`
- `./autogen.sh`
- `./configure` (flags)
- `make`



# IcedTea - compiling SDK & JRE

- Results:
  - SDK image
    - icedtea6/openjdk/build/linux-[platform](#)/j2sdk-image
  - JRE image
    - icedtea6/openjdk/build/linux-[platform](#)/j2re-image



# Who we are?



# Our Goals

- Open and free JDK [a journey started long ago]
- Best performing JDK on RHEL
- Improved supportability
- Improved performance of JBoss and other Red Hat java apps
- Community involvement





# IcedTea Timeline - Java 1.7

- June 7, 2007 IcedTea 1.0 Released
  - Bootstraps with Free Software tools
  - Depends only on Free Software available in Fedora
  - Encumbrances replaced with stubs
  - Autotools build system
- July 13, 2007: **1.1**: GNU Classpath crypto providers
- July 31, 2007: **1.2**: Font and Graphics support
- Aug 28, 2007: **1.3**: gcjwebplugin support
- Oct 12, 2007: **1.4**: Use system timezone data
- Jan 03, 2008: **1.5**: ppc/ppc64 interpreter
- Feb 13, 2008: **1.6**: libffi interpreter for all architectures
- June 02, 2008: **1.7**: synced with IcedTea6 1.2



# IcedTea in Fedora [Java 1.7 based]

November 8, 2007: Fedora 8 Release Includes IcedTea



# IcedTea Timeline - Java 1.6

- Remove the changes between 6 and 7 so that we could run the testsuite
- February 15, 2008: IcedTea6 1.0 Released
  - IcedTea adapted to OpenJDK 6 code base
- Subsequent releases on April 4, May 28, Oct 15, Oct 27
- Work on Java 7 based IcedTea on hold
- OpenJDK 6 / IcedTea released in Fedora 9 and Fedora 10



# IcedTea Timeline - Java 1.6

- Internal Java TCK [JCK] effort
- 19 June 2008: Passed JCK on x86 and x86\_64 on Fedora 9, results submitted to Sun
- Plugins, Porting and Performance
- The first (unpatched) OpenJDK 6 from Sun to pass JCK testing was b13, Dec 3 2008. Jan 2009: OpenJDK in RHEL 5.3, fully certified Java compatible
- July 2009: Ubuntu, following Red Hat's instructions and JCKDistilled (of which more later), finally announced that they had managed to pass the JCK



# OpenJDK Development and Packaging

- A key responsibility
- Not as simple as packaging what we get from upstream because upstream (still!) doesn't build on Free systems
- To make OpenJDK work on Red Hat / Fedora, we maintain a fairly significant divergence from upstream; some 31k lines of patches
- We're working with Sun to close the gap
- Some of these divergences are due to our need to use system **shared libraries** rather than **statically linking** everything



# OpenJDK

## Support for Linux



# OpenJDK Support for Linux

- Browser plug-in support
  - Despite the fact that for many people a web browser plugin is the fundamental reason they need Java, it is not part of OpenJDK. We've had to provide the plugins ourselves
  - There are two parts to the plugin: LiveConnect, which is a Java-to-Javascript bridge, and the Java applet plugin proper
  - Some security implications
    - In particular it means that we are the upstream for security fixes, not Sun



# OpenJDK Support for Linux

- Audio support
  - PulseAudio support for OpenJDK
  - We integrated Gervill (a software synthesizer) with OpenJDK to replace the proprietary MIDI backend
- We'd like to push all of this upstream, but it's very difficult
  - There's not really a group at Sun really dedicated to making everything work well on Linux, so we have to do it ourselves.
  - They're getting better at this, and our effort should go down over time





# OpenJDK

## Porting



# OpenJDK Porting

- OpenJDK is only available from Sun for:
  - i386
  - x86\_84
  - SPARC



# OpenJDK Porting

- Need (for Fedora and other distros)
  - ARM
  - PowerPC 32
  - PowerPC 64
  - ...Itanium, IBM zSeries...



# OpenJDK Porting

- Sun ports of Java have always contained significant amounts of assembly code.
- We've created a zero-assembler port (called Zero) that is used in Fedora where a full OpenJDK port is not available
  - Main developer: Gary Benson
  - Functionally fine but it's slow
  - **Includes only interpreter of Java bytecode, not JIT**



# Zero-assembler port

- Most of Sun's Java interpreters are written in assembly language
- Sun's Java has always used assembly language to interface between the interpreter and native code
- However, Sun had a C++ interpreter that wasn't originally included in OpenJDK but we persuaded them to release it
- Original HotSpot VM for OpenJDK
  - Over **10,000** lines of very-low level, system specific C++ code
  - And approx. **1,000** lines of assembler
  - Most critical code are written in assembly language
  - Used for more than 10 years on desktops, enterprise applications etc.
    - fast, optimized, reliable but not portable

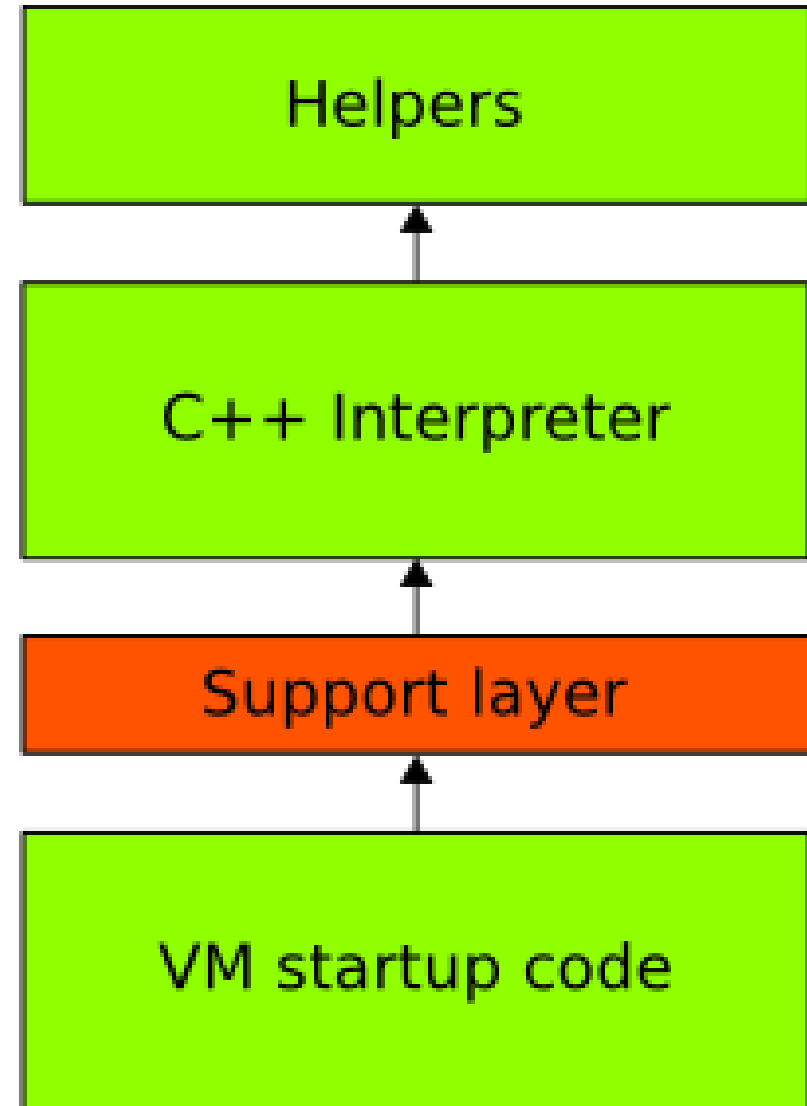
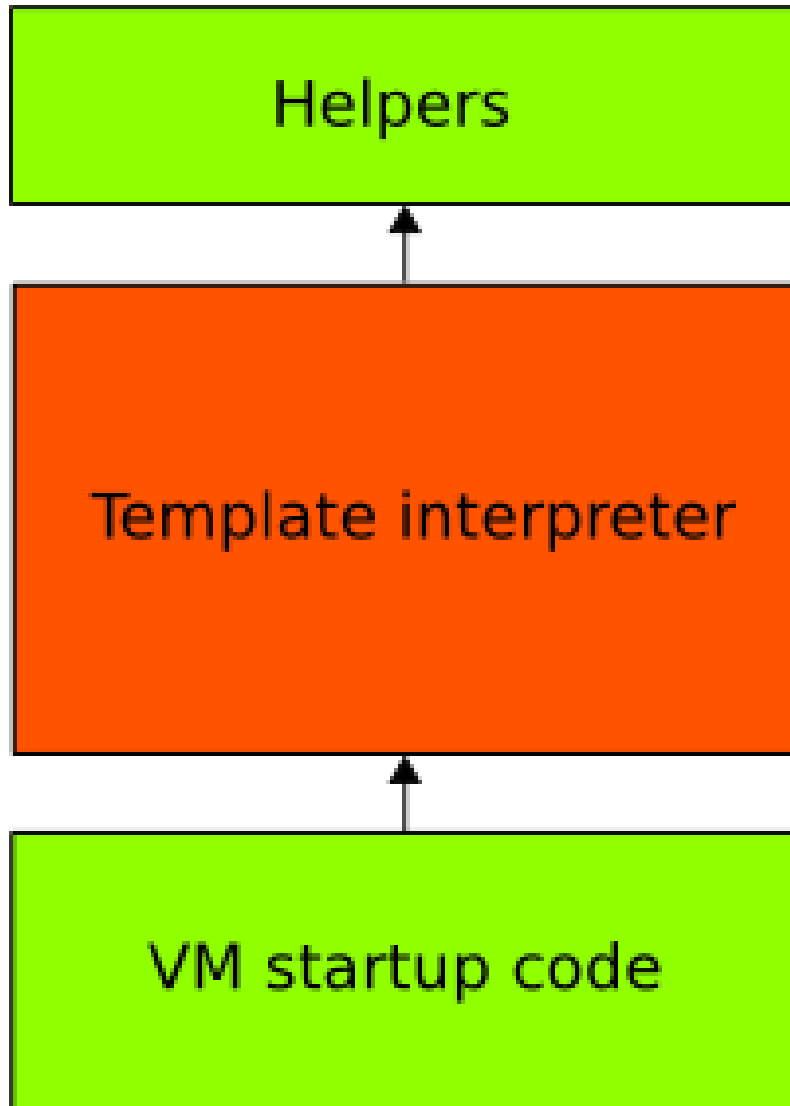


# Zero-assembler port

- We wanted a port of OpenJDK that would work everywhere, with no porting at all
- Zero does just that: the missing part [calls between native and interpreted code] is provided by libffi
- **Every platform with a libffi port can run OpenJDK**
- Zero is provided on many Linux platforms, even recently being used successfully on S/390
- However... it is rather slow
- Zero has been submitted for inclusion in JDK 7.



# HotSpot VM architecture



# Shark

- LLVM provides a JIT compiler
- Use the LLVM JIT compiler to generate code for a variety of platforms thus greatly reducing the time to port OpenJDK
  - Shark isn't yet stable, but initial benchmarking suggests that the performance, while not as good as the OpenJDK JIT, is much better than that of the C++ interpreter
    - 5-30 times faster than Zero (interpreter)
  - There's still a lot of low-hanging fruit for optimization
  - Shark is potentially very good indeed but it's a matter of how much work people want to do. **We'd like the community to take it up**
  - Many of the optimizations that are used in the HotSpot native ports could also be used in Shark
- Shark is not totally LLVM-specific: someone could port it to some other JIT compiler





# Zero & Shark

- Enabling Zero on all architectures
  - disabled by default on architectures with HotSpot VM
  - `./configure --enable-zero`
- Enabling Shark
  - disabled by default
  - `./configure --enable-shark`
- More info:
  - <http://icedtea.classpath.org/wiki/ZeroSharkFaq>



# OpenJDK

## Testing



# Java SE TCK (aka JCK) Testing

- Significant effort
- JCK test results are only valid for a particular combination of compiled binaries and operating system
- The JCK is very hard to set up
  - it took 2-3 months to pass all the tests
  - the maximum number of test failures is 0
  - 300+ page TCK manual to read
  - configuration in many cases is critical
  - tests every service that Java programs can use, so requires many servers to be set up
  - about 80k tests (compiler + devtoos + runtime)
- Complete JCK run now takes about one day on 8 core machine



# JCK Testing

- June 2008: Passed the JCK
- We were the first Linux distro to do so, and to date the *only* Linux distro that is certified
  - Ubuntu, Debian, SuSE, do not ship certified Java. (Unless they've passed the JCK and not told anyone!)
- Smaller distros do not have the ability to run the JCK on their own
- Created a wiki for other distros detailing how to avoid our pain
  - <http://icedtea.classpath.org/wiki/JCKDistilled>



JavaTest Harness: JCK6b-compiler: /jck/work

File Configure Run Tests Report View Tools Windows Help

View Filter: Status of All Tests Edit Filter...

Test Suite Root

- [-] api
  - [-] java\_rmi
    - [-] Naming
      - index.html#Bind
      - index.html#List
      - index.html#Lookup
      - index.html#Rebind
      - index.html#Unbind
    - registry
    - rmic
    - server
    - CreateTest.html
  - [-] javax\_annotation
  - [-] javax\_lang
  - [-] lang
    - [-] ANNOT
    - [-] ARR
    - [-] BINC
    - [-] CLSS
      - [-] clss001
      - [-] clss002
      - [-] clss003
      - [-] clss004
      - [-] clss005
      - [-] clss006
      - [-] clss007
      - [-] clss008
      - [-] clss009
      - [-] **clss010**
      - [-] clss011
      - [-] clss012
      - [-] clss013
      - [-] clss015
      - [-] clss016
      - [-] clss017
      - [-] clss018
      - [-] clss019

Folder: lang/CLSS/clss010

Summary Documentation Passed Failed Error Not Run Filtered Out

Selected Folder: clss010  
 Selected View Filter: All Tests  
 Unfiltered view of the test suite and results in the work directory.

Passed	25
Failed	1
Error	0
Not Run	19
<b>Sub-Total</b>	<b>45</b>
Filtered Out	0
<b>Total</b>	<b>45</b>

There are some failed tests in this folder.

Work Directory: work Configuration: compiler-config.jti Template: None

Finished test run. Elapsed Time 00:48:34

JCK6b-compiler JCK6b-compiler [2]



## Machine 'rhel5\_32bit'

20090630-1	Result location:	<a href="http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090630-1/">http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090630-1/</a>		
	Test suite	Passed	Failed	Error
	compiler	19996	12	0
	devtools	not run	not run	not run
	runtime	not run	not run	not run
20090714-1	Result location:	<a href="http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090714-1/">http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090714-1/</a>		
	Test suite	Passed	Failed	Error
	compiler	20008	0	0
	devtools	17566	0	0
	runtime	59021	0	0
20090720-1	Result location:	<a href="http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090720-1/">http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090720-1/</a>		
	Test suite	Passed	Failed	Error
	compiler	20008	0	0
	devtools	17566	0	0
	runtime	59021	0	0
20090813-1	Result location:	<a href="http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090813-1/">http://file.brq.redhat.com/~ptisnovs/jck/rhel5_32bit/20090813-1/</a>		
	Test suite	Passed	Failed	Error
	compiler	20008	0	0
	devtools	17537	29	0
	runtime	59020	0	0



# Regression tests

		2009	2009	2009	2009	2009	2009	2009	2009	2009	2009
		10-26	10-27	10-28	10-29	10-30	10-31	11-02	11-03	11-04	11-06
jdk-icedtea6											
java/awt/Focus/TranserFocusToWindow/TranserFocusToWindow.java	info										
java/awt/KeyboardFocusmanager/TypeAhead/ButtonActionKeyTest/ButtonActionKeyTest.html	info										
java/awt/KeyboardFocusmanager/TypeAhead/TestDialogTypeAhead.html	info										
java/awt/Multiscreen/LocationRelativeToTest/LocationRelativeToTest.java	info										
java/awt/TextArea/UsingWithMouse/SelectionAutoscrollTest.html	info										
java/awt/Window/AlwaysOnTop/AlwaysOnTopEvenOfWindow.java	info										
java/awt/event/HierarchyEvent/AncestorResized/AncestorResized.java	info										
java/beans/XMLEncoder/javax_swing_tree_TreePath.java	info										
java/net/InetAddress/CheckJNI.java	info										
java/net/MulticastSocket/NoLoopbackPackets.java	info										
java/net/MulticastSocket/SetLoopbackMode.java	info										
java/net/MulticastSocket/SetOutgoingIf.java	info										
java/net/MulticastSocket/Test.java	info										
java/net/ipv6tests/UdpTest.java	info										
java/rmi/activation/Activatable/checkActivateRef/CheckActivateRef.java	info										
java/util/logging/LoggingDeadlock2.java	info										
javax/swing/JLabel/6501991/bug6501991.java	info										
javax/xml/crypto/dsig/GenerationTests.java	info										
javax/xml/crypto/dsig/ValidationTests.java	info										
sun/java2d/cmm/ColorConvertOp/ColConvCCMTest.java	info										
sun/java2d/cmm/ColorConvertOp/ColConvDCMTest.java	info										
sun/java2d/cmm/ColorConvertOp/ConstructorsNullTest/ConstructorsNullTest.html	info										
sun/java2d/cmm/ColorConvertOp/MTColConvTest.java	info										
sun/security/ssl/javax/net/ssl/NewAPIs/SessionTimeOutTests.java	info										



# OpenJDK

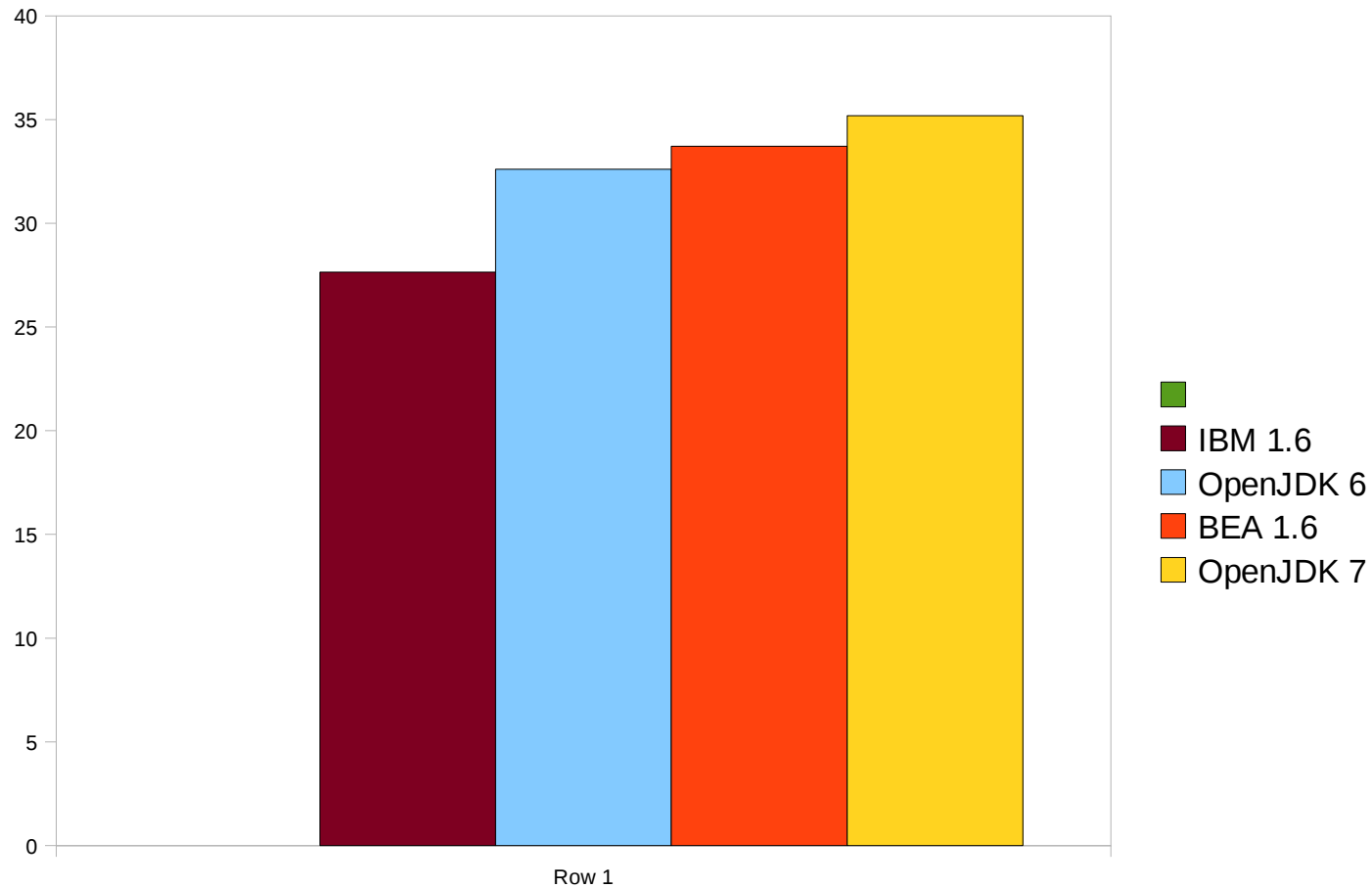
## Performance





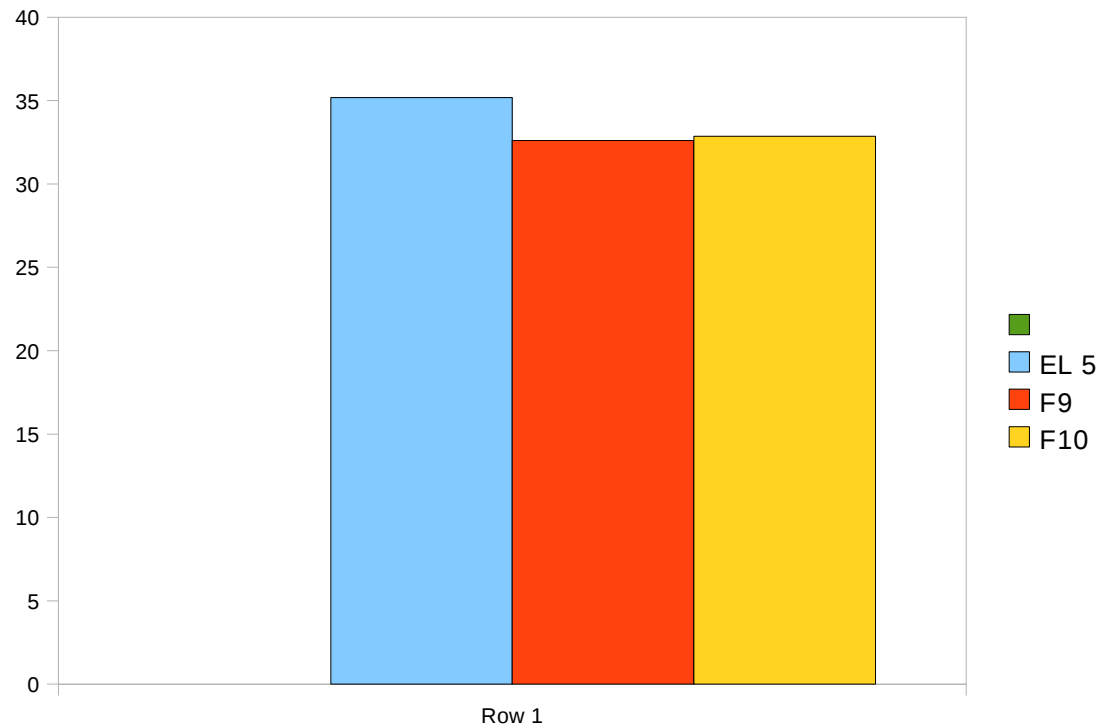
# Raw VM performance

SPEC jvm98, x86\_64, in SPEC ops/m:



# Raw VM performance (cont.)

- But we have seen decreasing VM performance with newer kernels. OpenJDK 7:

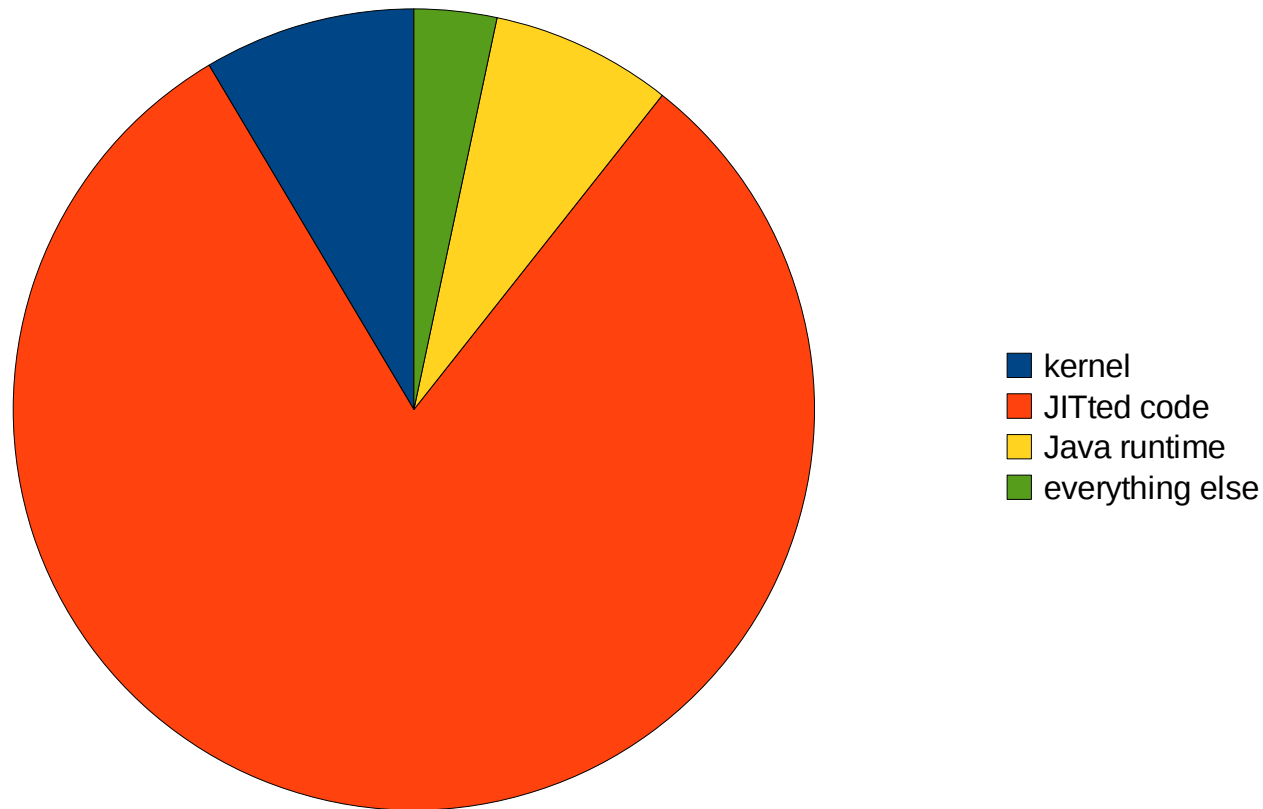


We don't think this is to do with CFQ, but we don't know what really causes it



# Benchmarking

SPECjvm 2008 test:



This is an extreme case as it's just testing the VM in isolation



# SPECjbb2005

SPECjbb2005

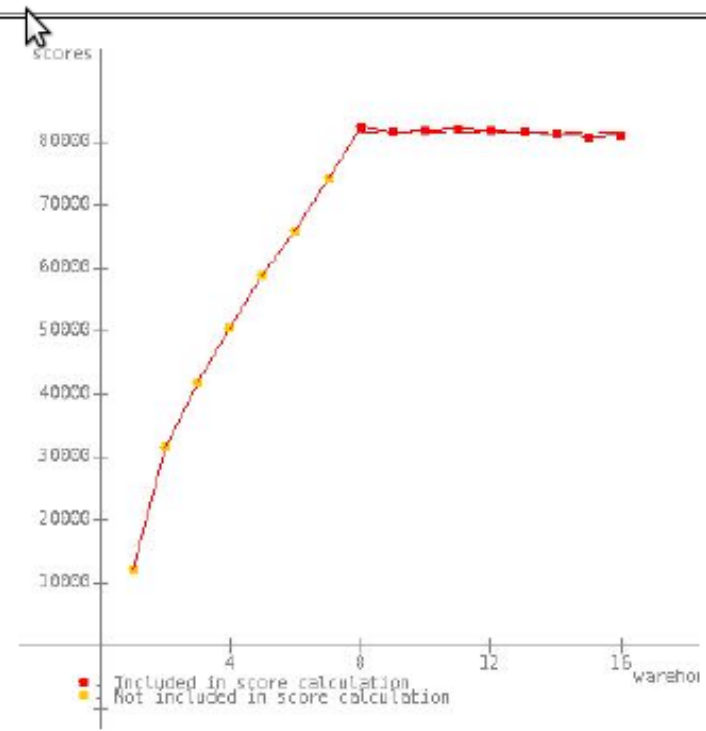
**SPECjbb2005 bops = 81787, SPECjbb2005 bops/JVM = 81787**

Dell PowerEdge CS1435

Red Hat/GNU Classpath IcedTea6 1.5.3

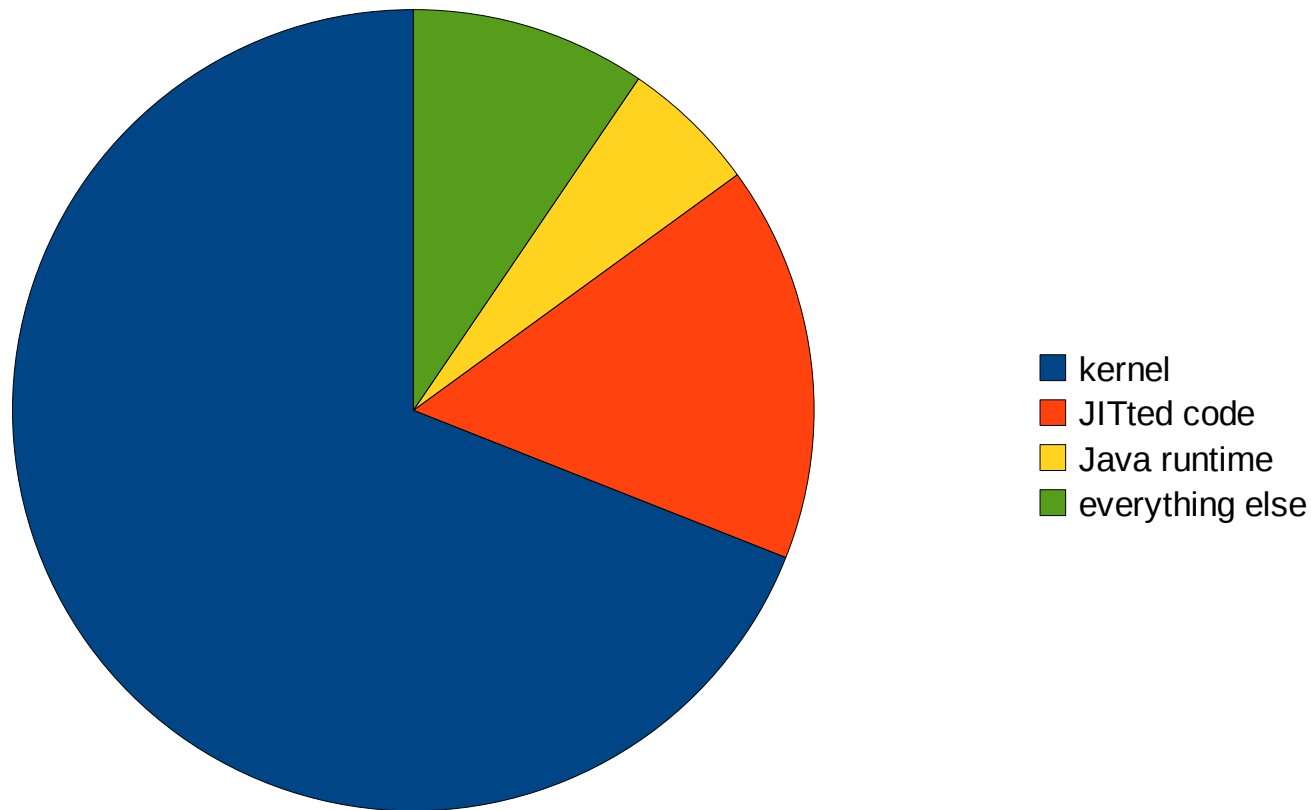
**No errors. Valid run.**

Warehouses	SPECjbb2005 bops	Incl. in metric
1	12191	
2	31670	
3	41967	
4	50577	
5	58885	
6	66095	
7	74293	
8	82607	*
9	81736	*
10	82060	*
11	82392	*
12	81967	*
13	81758	*
14	81510	*
15	80996	*
16	81055	*
<b>SPECjbb 2005</b>	<b>(from 8 to 16)</b>	<b>81787 SPECjbb 2005 bops</b>



# Whole system performance

JGroups test:



This is an extreme case (again :-)) as it's just messaging with no underlying database, but it shows that we shouldn't obsess about JIT performance to the exclusion of everything else



# URLs of Interest

- OpenJDK

`http://openjdk.java.net`

- IcedTea

`http://icedtea.classpath.org`

- Our Work List

`http://intranet.corp.redhat.com/ic/intranet/JavaProjectOpenJDK.html`

- JCK Distilled

`http://icedtea.classpath.org/wiki/JCKDistilled`

- Zero and Shark

`http://icedtea.classpath.org/wiki/ZeroSharkFaq`



# IRC

- [irc.oftc.net](https://irc.oftc.net)
- Channel [#openjdk](#)

You are welcome!

