

IntelliJ 90.137 Experiences



[Erik Pragt](#) 677 posts since

Oct 26, 2004

Hi all,

I felt adventurous today, so I installed IntelliJ 90.137. After a quick download, and an even quicker install (on Mac OS Snow Leopard), I ran IntelliJ, and everything looked nice. Except for the indexing part. The indexer indexed my Java 1.6 JDK approximately 7 times, cause my project to be valid, invalid, no valid, ehmm, invalid...I decided to give it a rest, and after a couple of minutes (argh) the indexer stopped. I went to the options screen, to see which nice things were added, and when going to the 'file types' part, my IDE got stuck in the option screen. Nothing worked anymore, and the only way to get out of the screen was by killing IntelliJ.

The second time I started IntelliJ, the indexing took place again, but after that, I started browsing through my source code. However, each time I click a source file, I see a spinning loader (or something) really annoying. It pops up, and goes away. The I started clicking through some POM's, and IntelliJ started a background indexing process....again!!! argh! Mind you, this is a 'project' with 20 source files in it, and 3 pom's, but currently, IntelliJ takes around 100% of my CPU time, doing...well, I have no idea what it is doing. I wanted to send a heap dump or something, but I forgot how to do it, and I cannot find the information to do so.

My plan was to give some constructive feedback, but my current experience with IntelliJ 90.137 doesn't allow me to give any feedback and I don't have time to work around all these issues. I hope the next release will work better, then I'll might try again. For now, back to IntelliJ 8.1.3!

Erik

Tags: slow, idea9, experience, does_not_work



[Erik Pragt](#) 677 posts since

Oct 26, 2004 1. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 7:45 PM

After reading some other threads about increasing Permgen to 250MB, I decided to give it another go. IntelliJ still keeps reindexing my files, but at least now it doesn't take up 100% CPU anymore. Actually, it takes up only 3.5%, but that is while typing this message.

After typing some classes, I have some feedback. Not being able to create interfaces, enums, etc is a big annoyance. The menu is still full with files I never use (HTML, XHTML, Mxml, PHP, Singleton (wtf?), XSLT stylesheets, etc), but the thing I use IntelliJ most for (Java editing!) is now crippled! Please revert this change, since it breaks my normal flow of creating enums, interfaces and classes with ease. Now I have to go to another menu to select which kind of file to create, and this is really keyboard unfriendly. What's next, a wizard?? ;-)

Btw, I also tried to fix this in the options, but I cannot find the actions to create an enum, interface, @interface. So again, please revert this (IMO) unnecessary change.

There also seems to be a new button 'Scroll from source'. I clicked on it, but it doesn't seem to do anything. It also doesn't 'stick', like in IntelliJ 8, so my guess is that it's broken? I couldn't find the function of the button in the help. I also don't like the new 'View options', I have to use my mouse more this way, but the options there weren't keyboard friendly anyway, so that's a minor thing. Though it would be helpful to get rid of it, since I click the 'show members' quite often, and now I have to click more often (sorry to complain about that, but I'm a keyboard user, which is why I like IntelliJ. Every mouse click I have to do breaks my workflow, so I would like to keep that to an absolute minimum. This just added a couple of clicks.)

Adding a Java class, or actually anything to my project, causes the project window to reparse(??) itself. Some folders just collapse and expand, which I also find quite distracting.

Btw, the overall editing speed seems to have improved enormously. IntelliJ feels incredibly fast (except for the reparsing stuff and the project view). If only the create new interface/enum/@interface could be reverted, that would be incredibly helpful, and it would also be

great if the multiple reparsing of the JDK could be fixed, though I have no doubt about that. Lastly, it would be great if the spinner wouldn't popup all the time (maybe increase the wait time to 1.5 seconds or something?) since it gives a the impression that IntelliJ is a bit nervous or something.

Erik

PS: Just tried a last thing, and that was to open the Dependency Matrix. IntelliJ showed me a popup asking me to recompile the project. I said 'okay, let's do it', but after 4 minutes, IntelliJ is still in process of doing a 'make', without any visible results. Also, no Dependency Matrix shows up....



[Erik Pragt](#) 677 posts since

Oct 26, 2004 [2](#). **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 7:46 PM

(btw, just tried to cancel the 'make' operation...but that also doesn't work. IntelliJ is now stuck in the background task 'Stopping - Make'...)



Guest [3](#). **Re: IntelliJ 90.137 Experiences** Nov 2, 2009 12:20 AM

I'd like to second that indexing problem you've mentioned. It's a pain in the butt on both IDEA 8 and 9. I don't understand this constant indexing. What is the reason you need to do this over and over across the whole project?

On my project I have 8 modules, each has 100's of class files. Our project's indexing at startup, and a few times throughout the day takes a good 5-10 minutes sometimes. We have a running joke that we should take our coffee break when idea indexes.

I hope something is done about it in IDEA 9.

R

On 11/1/09 2:46 PM, in article

30552516.149731257094016587.JavaMail.clearspace@app8.labs.intellij.net,

"Erik Pragt" <no_reply@jetbrains.com> wrote:

(btw, just tried to cancel the 'make' operation...but that also doesn't work.

IntelliJ is now stuck in the background task 'Stopping - Make'...)

Original message URL: <http://www.jetbrains.net/devnet/message/5248607#5248607>



[Erik Pragt](#) 677 posts since

Oct 26, 2004 4. **Re: IntelliJ 90.137 Experiences** Nov 2, 2009 12:59 AM

Ah, well, I usually only experience that in version 8 when I synchronize my Maven dependencies (which I do around 10-20 times a day). I was hoping this was one of the major improvements in IntelliJ 9. And it seems like it is, since adding dependencies to poms seems much faster compared to 8. If only the constant reindexing of the JDK (1.6 in my case, which

IntelliJ 90.137 Experiences

has been indexed 30 times now in 3 startups of IntelliJ), and probably other dependencies, wouldn't happen, it would be great!

Dmitry, please make it happen, and please don't have us wait for it to be stable in 9.0.3!



[Rui Vieira](#) 24 posts since

Nov 21, 2008 5. **Re: IntelliJ 90.137 Experiences** Nov 17, 2009 7:02 PM

I've been having this same problem, but with IU-90.193.



[Erik Pragt](#) 677 posts since

Oct 26, 2004 6. **Re: IntelliJ 90.137 Experiences** Nov 18, 2009 9:45 AM

Hmm, there's a new version? How did you know?



[Rui Vieira](#) 24 posts since

Nov 21, 2008 7. **Re: IntelliJ 90.137 Experiences** Nov 18, 2009 2:51 PM

I used to get the new EAP release announcements via RSS, but apparently that has stopped.

Unless you find a better way, just check here regularly:

<http://www.jetbrains.net/confluence/display/IDEADEV/Maia+EAP>



[Ronny Løvtangen](#) 47 posts since

Sep 13, 2002 8. **Re: IntelliJ 90.137 Experiences** Nov 18, 2009 3:29 PM

The RSS feed kind of stopped. The thing is they are updating the same announcement over and over rather than posting a new one. So my RSS client doesn't discover the new announcement.



[Yann Cebron](#) 1,289 posts since

Aug 22, 2002 9. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 8:37 PM

Not being able to create interfaces, enums, etc is a big annoyance

<http://blogs.jetbrains.com/idea/2009/05/compact-create-class-action/>

Read about all new features and changes in Maia: <http://blogs.jetbrains.com/idea/tag/maia>



[Erik Pragt](#) 677 posts since

Oct 26, 2004 10. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 11:05 PM

Well, the blog posting looks pretty nice, and I think I wouldn't have any problems with that. The current problem I have is that when I create a new class, I have to specify what kind of class I want to create using a drop down box, and that's something I'm not very fond of...I think I'd like the typing version better, though I haven't seen that in the 90.137 version.

Attachments:

-



[Screen shot 2009-11-01 at 21.03.31.png](#) (26.7 K)



[M. J. Milicevic](#) 688 posts since

Aug 22, 2002 11. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 11:49 PM

yes, for some reason they changed that



. Previous implementation was much better IMO, this new implementation makes me feel like I am using Eclipse..



[Erik Pragt](#) 677 posts since

Oct 26, 2004 12. **Re: IntelliJ 90.137 Experiences** Nov 2, 2009 12:55 AM

I totally agree. Hence my remark about wizards, since Eclipse equals wizards to me, while IntelliJ equals keyboard operated actions.



[Mike Liu](#) 66 posts since

Aug 22, 2002 13. **Re: IntelliJ 90.137 Experiences** Jan 29, 2010 8:15 AM

+1 on the whole idea-turned-wizard disappointment. I used to pride myself in showing off how fast I can create classes and edit java code in IDEA. With maia (9.0.1), I can no longer do that. In fact, I'm embarrassed to tell anyone I am using maia. Backup 8.1.4 for now...

IntelliJ developers : please fix this! no reason to conform to the rest of the world, when what you had going for you was exactly the selling point!



[Erik Pragt](#) 677 posts since

Oct 26, 2004 14. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 7:48 PM

Okay, really last remark: when are you going to bundle Tabswitch? I find it incredible nice to work with, and incredibly annoying to work without! +1 for bundling it! If you'd ask me, you can replace 'Recent files' by Tabswitch!



[Ronny Løvtangen](#) 47 posts since

Sep 13, 2002 15. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 9:51 PM

Bodiam wrote:

Okay, really last remark: when are you going to bundle Tabswitch? I find it incredible nice to work with, and incredibly annoying to work without! +1 for bundling it! If you'd ask me, you can replace 'Recent files' by Tabswitch!

<http://blogs.jetbrains.com/idea/2009/07/ctrltab-switcher-in-maia/>



[Erik Pragt](#) 677 posts since

Oct 26, 2004 16. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 11:00 PM

Great! I didn't know that, that's great to hear...read! Thanks for the reply!



[Ronny Løvtangen](#) 47 posts since

Sep 13, 2002 17. **Re: IntelliJ 90.137 Experiences** Nov 1, 2009 11:06 PM

You're welcome. Tabswitch has always been the first plugin I install. It's great that this feature now is included.



[Erik Pragt](#) 677 posts since

Oct 26, 2004 18. Re: **IntelliJ 90.137 Experiences** Nov 1, 2009 11:12 PM

So do I! And binding it to CTRL+TAB is actually a bit better than ALT+A, since it wouldn't be the first time that my code doesn't compile anymore due to an extra 'a' somewhere in my code..... (oops!)



[Dmitry Jemerov](#) 11,708 posts since

Aug 19, 2002 19. Re: **IntelliJ 90.137 Experiences** Nov 3, 2009 3:52 PM

Hello Erik,

I felt adventurous today, so I installed IntelliJ 90.137. After a quick download, and an even quicker install (on Mac OS Snow Leopard), I ran IntelliJ, and everything looked nice. Except for the indexing part. The indexer indexed my Java 1.6 JDK approximately 7 times, cause my project to be valid, invalid, no valid, ehmm, invalid...I decided to give it a rest, and after a couple of minutes (argh) the indexer stopped.

Are you using the Native Neighborhood plugin? If you are, please uninstall (most of its functionality is provided as core features in IDEA 9).

--

Dmitry Jemerov

IntelliJ 90.137 Experiences

Development Lead

JetBrains, Inc.

<http://www.jetbrains.com/>

"Develop with Pleasure!"



[Erik Pragt](#) 677 posts since

Oct 26, 2004 20. Re: **IntelliJ 90.137 Experiences** Nov 3, 2009 6:12 PM

Hi Dmitry,

I haven't installed that plugin. I haven't installed any (non-bundled) plugins for that matter.

Erik

Btw, I'm on a Mac, I don't even know if the Native Neighborhood plugin works on that.



[Nicholas Zoss](#) 1 posts since

Jan 16, 2004 21. Re: **IntelliJ 90.137 Experiences** Nov 3, 2009 10:45 PM

Unfortunately, I am experiencing similar issues. In my case, all I have to do is open a file and start typing. idea.exe is completely taking over a CPU, and once it dies off, I can 'reproduce' the condition by typing a few characters in the file.

At this time, the current release is unusable for me with the normal plugins turned on. I've disabled even most of the standard plugins -- I have 'CVS', 'Java Server Pages', and the

various simple JS ones (not Flex, Quirks, W3C). The only non-standard plugin I have is the SQL Query Plugin. I can at least get around a bit in it, but it is slow still.

It is a largish project, but I really don't think I'm doing anything special. I'm doing searches and editing files, and the application grinds to a halt. It'll be at 0% CPU, like as I type this message, then it jumps to 50-80% CPU if I type a few characters in the editor. Again, most plugins are off, just a relatively few inspections -- in fact, I'm using the Default, just in case.

It updates the index once on startup, but as far as I can tell (in Windows here) it isn't reporting that it's constantly reindexing or so on.

I'd like to contribute whatever information I can, but I don't have much.

Thanks for the continued efforts.



Mike Liu 66 posts since

Aug 22, 2002 22. **Re: IntelliJ 90.137 Experiences** Jan 29, 2010 8:27 AM

I am experiencing the same issue with the IU-94.192. Even with a small project. Even when I'm editing a non-Java file like Properties or HTML. The IDE performance is the only barrier for me to upgrade my license to 9. It's just crazy... what I do in IntelliJ 8 for 5 minutes would take more than 30 minutes in maia....

I'm an avid user of IntelliJ, and have been using it even before it was called IntelliJ (RefactorIT). Whenever possible, I'd showcase to my co-workers all the cool and powerful features available only in IntelliJ. Now, I'm embarrassed to even fire it up on my laptop. I have to say that this is the most disappointing release yet...

C'mon guys, I know you can fix this!



[Dmitry Jemerov](#) 11,708 posts since

Aug 19, 2002 23. **Re: IntelliJ 90.137 Experiences** Jan 29, 2010 12:23 PM

Hello Mike,

Please submit a CPU snapshot as described in <http://jetbrains.net/devnet/docs/DOC-192>

I am experiencing the same issue with the IU-94.192. Even with a small project. Even when I'm editing a non-Java file like Properties or HTML. The IDE performance is the only barrier for me to upgrade my license to 9. It's just crazy... what I do in IntelliJ 8 for 5 minutes would take more than 30 minutes in IntelliJ 9...

I'm an avid user of IntelliJ, and have been using it even before it was called IntelliJ (RefactorIT). Whenever possible, I'd showcase to my co-workers all the cool and powerful features available only in IntelliJ. Now, I'm embarrassed to even fire it up on my laptop. I have to say that this is the most disappointing release yet...

C'mon guys, I know you can fix this!

Original message URL:

<http://www.jetbrains.net/devnet/message/5255520#5255520>

--

Dmitry Jemerov

Development Lead

JetBrains, Inc.

<http://www.jetbrains.com/>

"Develop with Pleasure!"



Mike Liu 66 posts since

Aug 22, 2002 24. **Re: IntelliJ 90.137 Experiences** Feb 1, 2010 9:53 PM

well... wouldn't you know it. My 30-day eval license ended today. 😞

Maybe I'll wait for the next EAP... would I get more eval time for the next EAP if I install it over the current one?

Don't get me wrong... I'm not trying to get freebie eval license here. I'm being a registered user since intellij3. I'd absolutely go ahead with the upgrade if I know for certain that the performance problem will be resolved very shortly.

Thanks,

mike



[Dmitry Jemerov](#) 11,708 posts since

Aug 19, 2002 25. **Re: IntelliJ 90.137 Experiences** Feb 2, 2010 1:09 PM

Hello Mike,

well... wouldn't you know it. My 30-day eval license ended today.

Maybe I'll wait for the next EAP... would I get more eval time for the next EAP if I install it over the current one?

Post-release EAPs do not give you any additional eval time. However, you can contact our sales department at sales@jetbrains.com - they usually honor requests for extended evaluation.

Don't get me wrong... I'm not trying to get freebie eval license here. I'm being a registered user since intellij3. I'd absolutely go ahead with the upgrade if I know for certain that the performance problem will be resolved very shortly.

Unfortunately we are unable to do anything about your performance problem unless we have specific information about what causes it (a CPU snapshot

IntelliJ 90.137 Experiences

or a thread dump).

Does the performance problem persist if you open your project with the Community Edition?

--

Dmitry Jemerov

Development Lead

JetBrains, Inc.

<http://www.jetbrains.com/>

"Develop with Pleasure!"



[Mike Liu](#) 66 posts since

Aug 22, 2002 26. **Re: IntelliJ 90.137 Experiences** Feb 4, 2010 9:40 PM

Thanks for the suggestion. I did get another eval from sales. Just minutes ago, I've upload my CPU snapshots to the ftp site as IU-94.192_Lium_04.02.2010_10.25.16.zip. Email has been sent to support@jetbrains.com as well. Let's see what happens from here.

Thanks again.



[Dmitry Jemerov](#) 11,708 posts since

Aug 19, 2002 27. **Re: IntelliJ 90.137 Experiences** Feb 5, 2010 1:39 PM

Hello Mike,

Thanks for the suggestion. I did get another eval from sales. Just minutes ago, I've upload my CPU snapshots to the ftp site as IU-94.192_Lium_04.02.2010_10.25.16.zip. Email has been sent to <mailto:support@jetbrains.com> as well. Let's see what happens from here.

This is in fact quite strange because the CPU snapshot shows that IntelliJ IDEA is responsive for you pretty much all of the time, and there are no significant pauses in the event dispatch thread. The only strange thing is that your maximum memory is set to only 256M. The snapshot doesn't show the GC data, but it could be the case that all the pauses are caused by GC. Does it help to increase -Xmx to 512M?

--

Dmitry Jemerov

Development Lead

JetBrains, Inc.

<http://www.jetbrains.com/>

"Develop with Pleasure!"



Mike Liu 66 posts since

Aug 22, 2002 28. Re: IntelliJ 90.137 Experiences Feb 5, 2010 10:00 PM

Hehe... even more strange is when I increase heap to 512m, I got the following error when my project is being loaded:

Error while indexing C:\Program Files\IBM\SDP\runtimes\base_v61\java\jre\lib\core.jar!\sun\rm\server\Activation.class

To reindex this file IDEA has to be restarted: java.lang.OutOfMemoryError: Java heap space
java.lang.RuntimeException: java.lang.OutOfMemoryError: Java heap space

```
at com.intellij.util.io.PersistentEnumerator.valueOf(PersistentEnumerator.java:468)
at com.intellij.util.io.StringRef.getString(StringRef.java:48)
at com.intellij.util.io.StringRef.toString(StringRef.java:92)
at
com.intellij.psi.impl.java.stubs.impl.PsiClassStubImpl.getName(PsiClassStubImpl.java:71)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaClassElementType.java:145)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaClassElementType.java:48)
at com.intellij.psi.stubs.StubTree.indexStubTree(StubTree.java:79)
at com.intellij.psi.stubs.StubUpdatingIndex$MyIndex.a(StubUpdatingIndex.java:294)
at com.intellij.psi.stubs.StubUpdatingIndex
$MyIndex.updateWithMap(StubUpdatingIndex.java:249)
at com.intellij.util.indexing.MapReduceIndex.update(MapReduceIndex.java:232)
at com.intellij.util.indexing.FileBasedIndex$18.run(FileBasedIndex.java:1216)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeNonCancelableSection(ProgressManagerImpl.java:121)
at com.intellij.util.indexing.FileBasedIndex.a(FileBasedIndex.java:1213)
at com.intellij.util.indexing.FileBasedIndex.indexFileContent(FileBasedIndex.java:1180)
```

```
    at
com.intellij.util.indexing.UnindexedFilesUpdater.processFile(UnindexedFilesUpdater.java:49)
    at
com.intellij.openapi.project.CacheUpdateSession.processFile(CacheUpdateSession.java:84)
    at com.intellij.openapi.project.CacheUpdateRunner
$3$1.run(CacheUpdateRunner.java:144)
    at
com.intellij.openapi.application.impl.ApplicationImpl.runReadAction(ApplicationImpl.java:691)
    at com.intellij.openapi.project.CacheUpdateRunner$3.run(CacheUpdateRunner.java:148)
    at com.intellij.openapi.progress.impl.ProgressManagerImpl
$2.run(ProgressManagerImpl.java:197)
    at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessUnderProgress(ProgressManagerImpl.java:188)
    at
com.intellij.openapi.progress.impl.ProgressManagerImpl.runProcess(ProgressManagerImpl.java:188)
    at com.intellij.openapi.project.CacheUpdateRunner.a(CacheUpdateRunner.java:123)
    at
com.intellij.openapi.project.CacheUpdateRunner.processFiles(CacheUpdateRunner.java:81)
    at com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable
$1.a(DumbServiceImpl.java:294)
    at com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable
$1.run(DumbServiceImpl.java:278)
    at com.intellij.openapi.progress.impl.ProgressManagerImpl
$TaskRunnable.run(ProgressManagerImpl.java:422)
    at com.intellij.openapi.progress.impl.ProgressManagerImpl
$2.run(ProgressManagerImpl.java:197)
    at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessUnderProgress(ProgressManagerImpl.java:188)
    at
com.intellij.openapi.progress.impl.ProgressManagerImpl.runProcess(ProgressManagerImpl.java:188)
    at com.intellij.openapi.progress.impl.ProgressManagerImpl
$6.run(ProgressManagerImpl.java:337)
    at com.intellij.openapi.application.impl.ApplicationImpl$5.run(ApplicationImpl.java:328)
    at java.util.concurrent.Executors$RunnableAdapter.call(Unknown Source)
    at java.util.concurrent.FutureTask$Sync.innerRun(Unknown Source)
    at java.util.concurrent.FutureTask.run(Unknown Source)
```

```
at java.util.concurrent.ThreadPoolExecutor$Worker.runTask(Unknown Source)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
at java.lang.Thread.run(Unknown Source)
at com.intellij.openapi.application.impl.ApplicationImpl$1$1.run(ApplicationImpl.java:124)
Caused by: java.lang.OutOfMemoryError: Java heap space
at java.util.Arrays.copyOf(Unknown Source)
at java.lang.StringValue.from(Unknown Source)
at java.lang.String.<init>(Unknown Source)
at com.intellij.util.io.IOUtil.readUTFFast(IOUtil.java:119)
at
com.intellij.util.io.EnumeratorStringDescriptor.read(EnumeratorStringDescriptor.java:42)
at
com.intellij.util.io.EnumeratorStringDescriptor.read(EnumeratorStringDescriptor.java:26)
at com.intellij.util.io.PersistentEnumerator.valueOf(PersistentEnumerator.java:460)
at com.intellij.util.io.StringRef.getString(StringRef.java:48)
at com.intellij.util.io.StringRef.toString(StringRef.java:92)
at
com.intellij.psi.impl.java.stubs.impl.PsiClassStubImpl.getName(PsiClassStubImpl.java:71)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaClassElementType.java:145)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaClassElementType.java:48)
at com.intellij.psi.stubs.StubTree.indexStubTree(StubTree.java:79)
at com.intellij.psi.stubs.StubUpdatingIndex$MyIndex.a(StubUpdatingIndex.java:294)
at com.intellij.psi.stubs.StubUpdatingIndex
$MyIndex.updateWithMap(StubUpdatingIndex.java:249)
at com.intellij.util.indexing.MapReduceIndex.update(MapReduceIndex.java:232)
at com.intellij.util.indexing.FileBasedIndex$18.run(FileBasedIndex.java:1216)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeNonCancelableSection(ProgressManagerImpl.java:121)
at com.intellij.util.indexing.FileBasedIndex.a(FileBasedIndex.java:1213)
at com.intellij.util.indexing.FileBasedIndex.indexFileContent(FileBasedIndex.java:1180)
at
com.intellij.util.indexing.UnindexedFilesUpdater.processFile(UnindexedFilesUpdater.java:49)
```

```
    at
com.intellij.openapi.project.CacheUpdateSession.processFile(CacheUpdateSession.java:84)
    at com.intellij.openapi.project.CacheUpdateRunner
$3$1.run(CacheUpdateRunner.java:144)
    at
com.intellij.openapi.application.impl.ApplicationImpl.runReadAction(ApplicationImpl.java:691)
    at com.intellij.openapi.project.CacheUpdateRunner$3.run(CacheUpdateRunner.java:148)
    at com.intellij.openapi.progress.impl.ProgressManagerImpl
$2.run(ProgressManagerImpl.java:197)
    at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessUnderProgress(ProgressManagerImpl.java:188)
    at com.intellij.openapi.project.CacheUpdateRunner.a(CacheUpdateRunner.java:123)
    at
com.intellij.openapi.project.CacheUpdateRunner.processFiles(CacheUpdateRunner.java:81)
    at com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable
$1.a(DumbServiceImpl.java:294)
    at com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable
$1.run(DumbServiceImpl.java:278)
```

This is my vmoptions:

```
-Dsun.awt.keepWorkingSetOnMinimize=true
-XX:+UnlockExperimentalVMOptions
-XX:+UseG1GC
-XX:MaxGCPauseMillis=1000
-XX:GCPauseIntervalMillis=10000
-XX:+G1ParallelRSetUpdatingEnabled
-XX:+G1ParallelRSetScanningEnabled
-XX:+DoEscapeAnalysis
-XX:+AggressiveOpts
-Xms128m
-Xmx512m
```

IntelliJ 90.137 Experiences

-XX:MaxPermSize=150m

-agentlib:yjpagent=disablej2ee,sessionname=IntelliJIdea90

I will try to move ahead with more testing with 512M. Will let you know what happens.

Thanks,

Mike



[Dmitry Jemerov](#) 11,708 posts since

Aug 19, 2002 29. **Re: IntelliJ 90.137 Experiences** Feb 8, 2010 1:35 PM

Hello Mike,

And if you clear the system directory and then run IDEA again?

Hehe... even more strange is when I increase heap to 512m, I got the following error when my project is being loaded:

Error while indexing C:\Program

Files\IBM\SDP\runtimes\base_v61\java\jre\lib\core.jar!\sun\rmi\server\

Activation.class

To reindex this file IDEA has to be restarted:

java.lang.OutOfMemoryError: Java heap space

java.lang.RuntimeException: java.lang.OutOfMemoryError: Java heap space

```
at
com.intellij.util.io.PersistentEnumerator.valueOf(PersistentEnumerator
.java:468)
at com.intellij.util.io.StringRef.getString(StringRef.java:48)
at com.intellij.util.io.StringRef.toString(StringRef.java:92)
at
com.intellij.psi.impl.java.stubs.impl.PsiClassStubImpl.getName(PsiClas
sStubImpl.java:71)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaCl
assElementType.java:145)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaCl
assElementType.java:48)
at com.intellij.psi.stubs.StubTree.indexStubTree(StubTree.java:79)
at
com.intellij.psi.stubs.StubUpdatingIndex$MyIndex.a(StubUpdatingIndex.j
ava:294)
at
com.intellij.psi.stubs.StubUpdatingIndex$MyIndex.updateWithMap(StubUpd
atingIndex.java:249)
at
```

```
com.intellij.util.indexing.MapReduceIndex.update(MapReduceIndex.java:2
32)
at
com.intellij.util.indexing.FileBasedIndex$18.run(FileBasedIndex.java:1
216)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeNonCance
lableSection(ProgressManagerImpl.java:134)
at
com.intellij.util.indexing.FileBasedIndex.a(FileBasedIndex.java:1213)
at
com.intellij.util.indexing.FileBasedIndex.indexFileContent(FileBasedIn
dex.java:1180)
at
com.intellij.util.indexing.UnindexedFilesUpdater.processFile(Unindexed
FilesUpdater.java:49)
at
com.intellij.openapi.project.CacheUpdateSession.processFile(CacheUpdat
eSession.java:84)
at
com.intellij.openapi.project.CacheUpdateRunner
$3$1.run(CacheUpdateRunn
er.java:144)
```

```
at
com.intellij.openapi.application.impl.ApplicationImpl.runReadAction(Ap
plicationImpl.java:691)
at
com.intellij.openapi.project.CacheUpdateRunner
$3.run(CacheUpdateRunner
.java:148)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl$2.run(ProgressM
anagerImpl.java:197)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessU
nderProgress(ProgressManagerImpl.java:222)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.runProcess(Prog
ressManagerImpl.java:188)
at
com.intellij.openapi.project.CacheUpdateRunner.a(CacheUpdateRunner.jav
a:123)
at
com.intellij.openapi.project.CacheUpdateRunner.processFiles(CacheUpdat
eRunner.java:81)
at
```

```
com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable$1.a(D  
umbServiceImpl.java:294)  
at  
com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable$1.run  
(DumbServiceImpl.java:278)  
at  
com.intellij.openapi.progress.impl.ProgressManagerImpl$TaskRunnable.ru  
n(ProgressManagerImpl.java:422)  
at  
com.intellij.openapi.progress.impl.ProgressManagerImpl$2.run(ProgressM  
anagerImpl.java:197)  
at  
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessU  
nderProgress(ProgressManagerImpl.java:222)  
at  
com.intellij.openapi.progress.impl.ProgressManagerImpl.runProcess(Prog  
ressManagerImpl.java:188)  
at  
com.intellij.openapi.progress.impl.ProgressManagerImpl$6.run(ProgressM  
anagerImpl.java:337)  
at  
com.intellij.openapi.application.impl.ApplicationImpl$5.run(Applicatio
```

```
nImpl.java:328)
at java.util.concurrent.Executors$RunnableAdapter.call(Unknown
Source)
at java.util.concurrent.FutureTask$Sync.innerRun(Unknown Source)
at java.util.concurrent.FutureTask.run(Unknown Source)
at java.util.concurrent.ThreadPoolExecutor$Worker.runTask(Unknown
Source)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown
Source)
at java.lang.Thread.run(Unknown Source)
at
com.intellij.openapi.application.impl.ApplicationImpl$1$1.run(Applicat
ionImpl.java:124)
Caused by: java.lang.OutOfMemoryError: Java heap space
at java.util.Arrays.copyOf(Unknown Source)
at java.lang.StringValue.from(Unknown Source)
at java.lang.String.<init>(Unknown Source)
at com.intellij.util.io.IOUtil.readUTFFast(IOUtil.java:119)
at
com.intellij.util.io.EnumeratorStringDescriptor.read(EnumeratorStringD
escriptor.java:42)
at
```

```
com.intellij.util.io.EnumeratorStringDescriptor.read(EnumeratorStringD
descriptor.java:26)
at
com.intellij.util.io.PersistentEnumerator.valueOf(PersistentEnumerator
.java:460)
at com.intellij.util.io.StringRef.getString(StringRef.java:48)
at com.intellij.util.io.StringRef.toString(StringRef.java:92)
at
com.intellij.psi.impl.java.stubs.impl.PsiClassStubImpl.getName(PsiClas
sStubImpl.java:71)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaCl
assElementType.java:145)
at
com.intellij.psi.impl.java.stubs.JavaClassElementType.indexStub(JavaCl
assElementType.java:48)
at com.intellij.psi.stubs.StubTree.indexStubTree(StubTree.java:79)
at
com.intellij.psi.stubs.StubUpdatingIndex$MyIndex.a(StubUpdatingIndex.j
ava:294)
at
com.intellij.psi.stubs.StubUpdatingIndex$MyIndex.updateWithMap(StubUpd
```

```
atingIndex.java:249)
at
com.intellij.util.indexing.MapReduceIndex.update(MapReduceIndex.java:2
32)
at
com.intellij.util.indexing.FileBasedIndex$18.run(FileBasedIndex.java:1
216)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeNonCance
lableSection(ProgressManagerImpl.java:134)
at
com.intellij.util.indexing.FileBasedIndex.a(FileBasedIndex.java:1213)
at
com.intellij.util.indexing.FileBasedIndex.indexFileContent(FileBasedIn
dex.java:1180)
at
com.intellij.util.indexing.UnindexedFilesUpdater.processFile(Unindexed
FilesUpdater.java:49)
at
com.intellij.openapi.project.CacheUpdateSession.processFile(CacheUpdat
eSession.java:84)
at
```

```
com.intellij.openapi.project.CacheUpdateRunner
$3$1.run(CacheUpdateRunn

er.java:144)
at
com.intellij.openapi.application.impl.ApplicationImpl.runReadAction(Ap
plicationImpl.java:691)
at
com.intellij.openapi.project.CacheUpdateRunner
$3.run(CacheUpdateRunner
.java:148)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl$2.run(ProgressM
anagerImpl.java:197)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.executeProcessU
nderProgress(ProgressManagerImpl.java:222)
at
com.intellij.openapi.progress.impl.ProgressManagerImpl.runProcess(Prog
ressManagerImpl.java:188)
at
com.intellij.openapi.project.CacheUpdateRunner.a(CacheUpdateRunner.jav
a:123)
at
```

```
com.intellij.openapi.project.CacheUpdateRunner.processFiles(CacheUpdateRunner.java:81)
at
com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable$1.a(DumbServiceImpl.java:294)
at
com.intellij.openapi.project.DumbServiceImpl$IndexUpdateRunnable$1.run(DumbServiceImpl.java:278)
This is my vmoptions:
-Dsun.awt.keepWorkingSetOnMinimize=true
-XX:+UnlockExperimentalVMOptions
-XX:+UseG1GC
-XX:MaxGCPauseMillis=1000
-XX:GCPauseIntervalMillis=10000
-XX:+G1ParallelRSetUpdatingEnabled
-XX:+G1ParallelRSetScanningEnabled
-XX:+DoEscapeAnalysis
-XX:+AggressiveOpts
-Xms128m
-Xmx512m
-XX:MaxPermSize=150m
-agentlib:yjpagent=disablej2ee,sessionname=IntelliJIdea90
```

I will try to move ahead with more testing with 512M. Will let you know what happens.

Thanks,

Mike

Original message URL:

<http://www.jetbrains.com/devnet/message/5256094#5256094>

--

Dmitry Jemerov

Development Lead

JetBrains, Inc.

<http://www.jetbrains.com/>

"Develop with Pleasure!"



[Bas Leijdekkers](#) *2,208 posts since*

Aug 19, 2002 30. **Re: IntelliJ 90.137 Experiences** Feb 8, 2010 2:17 PM

Hello Mike,

javaexpert wrote:

This is my vmoptions:

```
-Dsun.awt.keepWorkingSetOnMinimize=true  
-XX:+UnlockExperimentalVMOptions  
-XX:+UseG1GC  
-XX:MaxGCPauseMillis=1000  
-XX:GCPauseIntervalMillis=10000  
-XX:+G1ParallelRSetUpdatingEnabled  
-XX:+G1ParallelRSetScanningEnabled  
-XX:+DoEscapeAnalysis  
-XX:+AggressiveOpts  
-Xms128m  
-Xmx512m  
-XX:MaxPermSize=150m  
-agentlib:yjpagent=disablej2ee,sessionname=IntelliJIdea90
```

I would recommend to not (yet) use the G1 garbage collector. In my experience it can not provide the performance and stability to comfortably run IntelliJ IDEA.

Secondly the MaxPermSize is most likely set too low. On a 32 bit JVM 250m is better, and on a 64 bit JVM I would set it as high as 500m.

Both these issue may be a cause for your performance problems.

Bas



[Dmitry Jemerov](#) 11,708 posts since

Aug 19, 2002 31. **Re: IntelliJ 90.137 Experiences** Feb 8, 2010 2:21 PM

Hello Bas,

Secondly the MaxPermSize is most likely set too low. On a 32 bit JVM

250m is better, and on a 64 bit JVM I would set it as high as 500m.

Both these issue may be a cause for your performance problems.

The comment about MaxPermSize is not exactly true, as far as I understand.

Either you have enough permgen space (then everything runs normally) or you don't (then everything grinds to a halt).

--

Dmitry Jemerov

Development Lead

JetBrains, Inc.

<http://www.jetbrains.com/>

"Develop with Pleasure!"



[Bas Leijdekkers](#) 2,208 posts since

Aug 19, 2002 32. **Re: IntelliJ 90.137 Experiences** Feb 8, 2010 2:30 PM

Hello Dmitry,

yole wrote:

Secondly the MaxPermSize is most likely set too low. On a 32 bit JVM 250m is better, and on a 64 bit JVM I would set it as high as 500m.

Both these issue may be a cause for your performance problems.

The comment about MaxPermSize is not exactly true, as far as I understand.

Either you have enough permgen space (then everything runs normally) or you don't (then everything grinds to a halt).

I have seen cases where IntelliJ IDEA had apparently enough permgen space to not throw out of memory errors, but acted very slow and periodically unresponsive (probably because the jvm was garbage collecting). Increasing the MaxPermSize specified fixed the issue.

Bas



[Mike Liu](#) 66 posts since

Aug 22, 2002 33. **Re: IntelliJ 90.137 Experiences** Feb 8, 2010 9:05 PM

Here's my current setting:

```
-Dsun.awt.keepWorkingSetOnMinimize=true  
-Xms256m  
-Xmx512m  
-XX:MaxPermSize=250m  
-agentlib:yjpagent=disablej2ee,sessionname=IntelliJIdea90
```

It's running on Windows XP (32bit) with 94.192/JDK 1.6.18:

```
java version "1.6.0_18"
```

```
Java(TM) SE Runtime Environment (build 1.6.0_18-b07)
```

```
Java HotSpot(TM) Client VM (build 16.0-b13, mixed mode, sharing)
```

So far, I haven't been any OOM exception, the project did take a while to open since I cleared the system directory. With just a few minutes of using IDEA with the above setting, everything seems ok for now. But it's a little premature to make any reasonable assessment. I'll keep things running like this for a few more days.

Thanks again for all who had contributed to this issue!

mike



[Mike Liu](#) 66 posts since

Aug 22, 2002 34. **Re: IntelliJ 90.137 Experiences** Feb 11, 2010 12:30 PM

OK. I think I'm fairly confident of what's happening now (but, never say never... I could still be wrong).

If I open the project with no files open in the editor, or very few simple files in the editors, then in general IntelliJ's behavior quite speedily.

Suppose if I left a bunch of files opened (like 20 - 30 files) and exit IntelliJ. The next time I start up IntelliJ, it becomes noticeably slower. Also, if I open at once 15 - 20 files, IntelliJ seems to behave slower -- not just during the files being open, but the degradation persist

for quite some time. This problem is especially obvious when I'm opening large (more than 500 lines) JSP, XML or properties files.

Again, if I'm working on a small set of files or just Java code, IntelliJ is doing quite well. Although I must say that "-Xms256m -Xmx512m" seems to be necessary. Anything less than that would at some point result in OOM or performance issue (as was the case for me with "-Xms128 -Xmx192m"). I wonder why Maia requires so much more memory than its predecessor? Anyone to shed light on this?

For completeness:

I'm using 94.273/JDK 1.6.0_18/WindowsXP SP2. Here's my vmoptions:

```
-Dsun.awt.keepWorkingSetOnMinimize=true  
-Xms256m  
-Xmx512m  
-XX:MaxPermSize=250m  
-agentlib:yjpagent=disablej2ee,sessionname=IntelliJIdea90
```

One more caveat: my project is using IBM JDK 1.5. Not sure if this matter at all.

Thanks,

Mike