Boosting the Performance of your Eclipse IDE

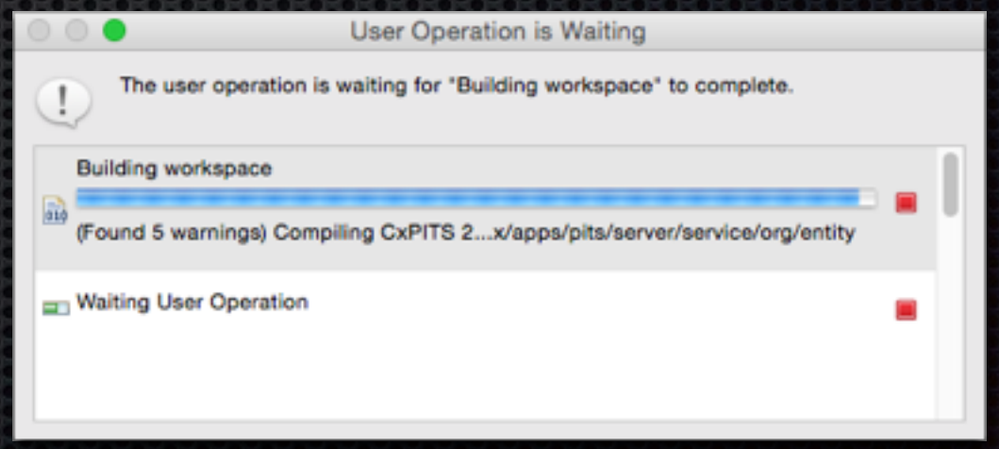
Karsten Thoms
itemis

EclipseCon Europe, Ludwigsburg, 2015-11-05

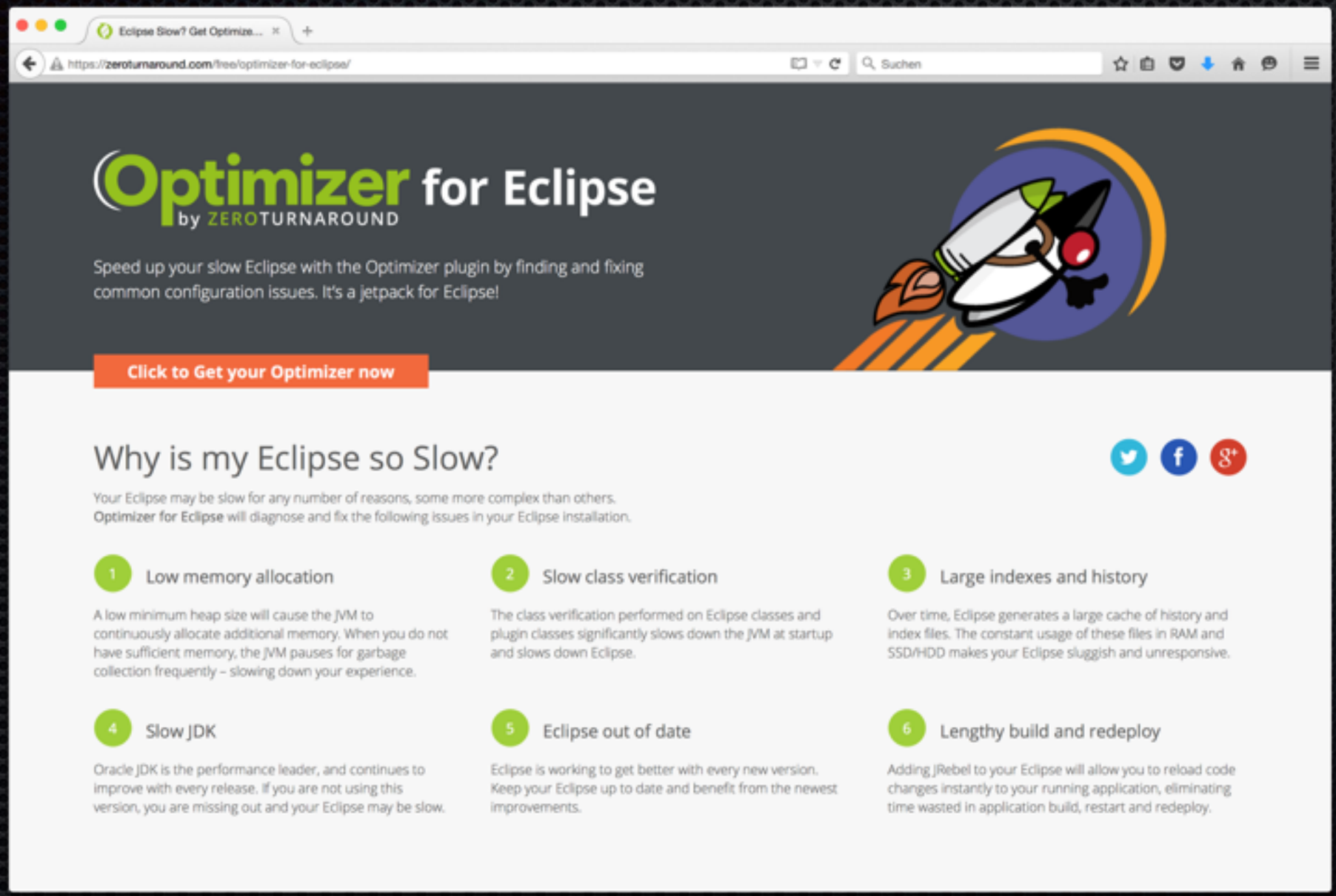
https://stocksnap.io/photo/PTGK3N8C1C

Where performance issues are annoying

- Startup
- Build
- UI Responsiveness



The convenient way



Optimizer for Eclipse
by ZEROTURNAROUND

Speed up your Eclipse by finding and fixing common configuration issues in just a few clicks.

We did a quick scan for you.
Here's what can be done:

- ⚠ Tune memory settings
- ⚠ Disable class verification
- ⚠ Get the latest Eclipse
- ⚠ Skip build and redeploy

- ✓ Clean indexes and history
- ✓ Use the latest Oracle JDK

Sign up and sta

Full name

Work email

Phone

Company

Developers in your compa

N/A | 1-2

Get tweaking →

[R] evolutionary developer tools by ZeroTurnaround

Optimizer for Eclipse
by ZEROTURNAROUND

Speed up your Eclipse by finding and fixing common configuration issues in just a few clicks.

Tune memory settings

Your current heap size is **40-2048 MB**. With insufficient memory, the JVM continuously allocates additional memory or pauses for garbage collection.

Tune settings

Disable class verification

Class verification is currently **enabled**. Disabling this significantly speeds up Eclipse at startup.

Disable verification

Clean indexes and history

Your indexes and history are already nice and clean (last cleaned **2 months** ago). Great job!

✓

Use the latest Oracle JDK

You are currently running the latest — **Oracle Java 1.8.0_51**. Excellent!

✓

Get the latest Eclipse

You are using **Luna SR2 (4.4.2)**. Upgrade to **Mars (4.5.0)** and benefit from the newest improvements.

Upgrade Eclipse

Skip build and redeploy

Adding **JRebel** to your Eclipse will allow you to instantly reload code changes to the running application. Learn more...

Install JRebel

Close | Restart Eclipse

[R] evolutionary developer tools by ZeroTurnaround

Support Forum

Optimizer by ZEROTURNAROUND for Eclipse

You just saved 4 seconds (17%) at startup!

Challenge your friends to do better! Let them know.

P.S. Sharing is caring.
Especially with things as cool as this!

← Back to Optimizer

[R] evolutionary developer tools by ZeroTurnaround

Support Forum

Wow! That was too easy!

Hunting performance issues

http://www.stockvault.net/photo/164167/iceberg

Hardware

Hidden costs of waiting

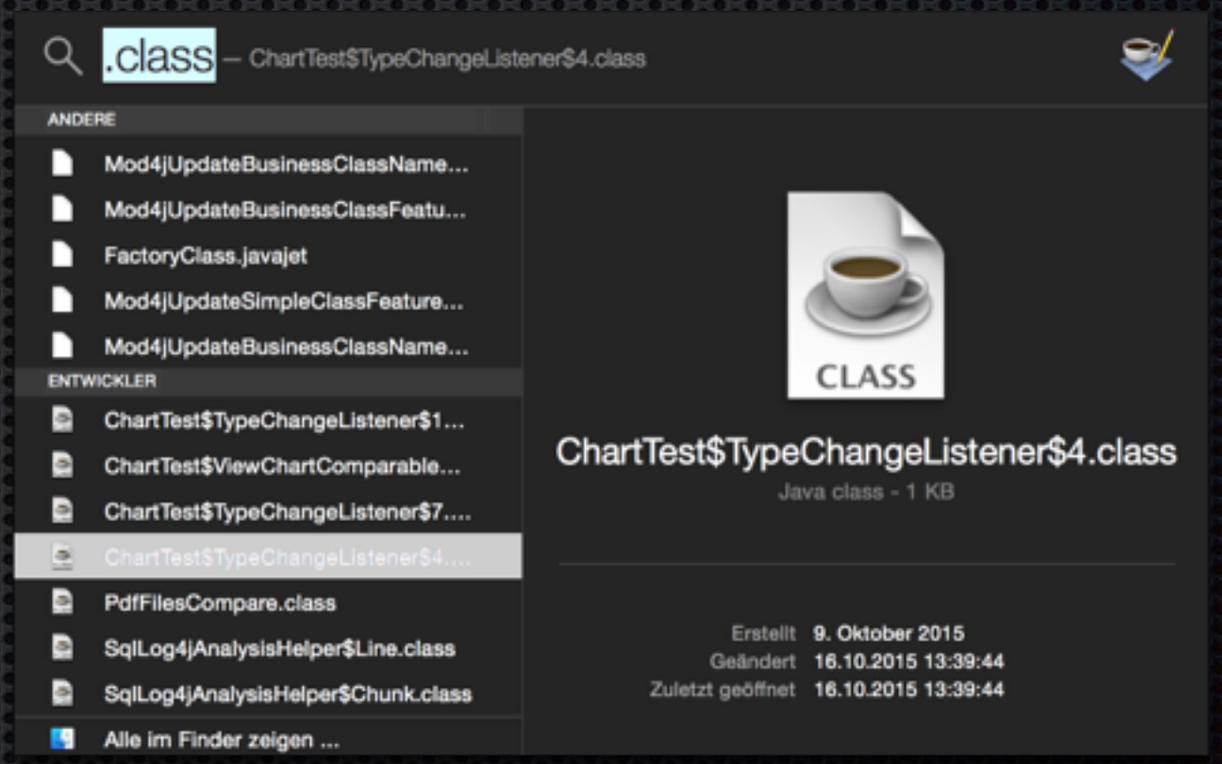
Configure Anti-Virus

- JRE/JDK
- Eclipse Installation Folder
- ~/.p2
- Workspace Folder
- Project Folders
- Output Locations

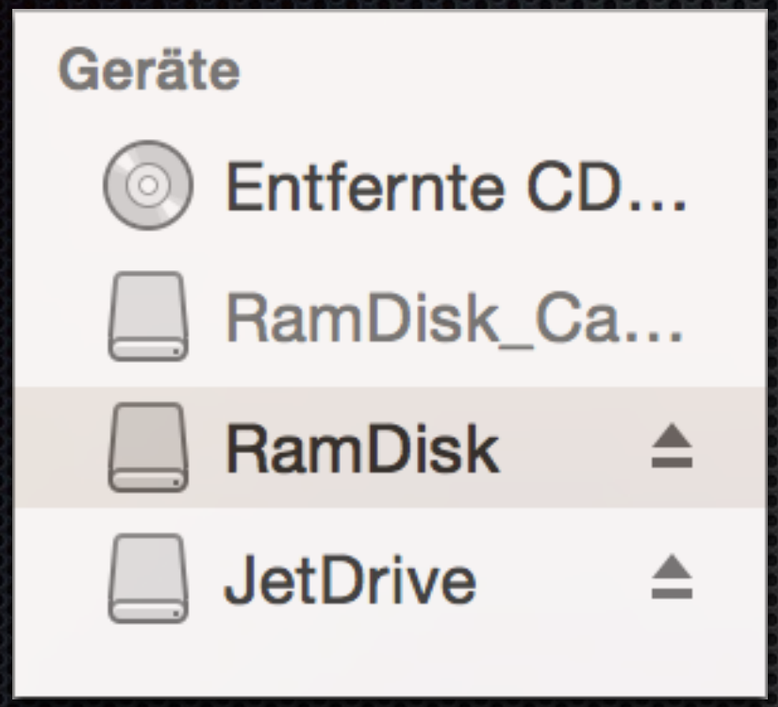
http://www.stockvault.net/photo/180403/cyber-security-concept-with-umbrella-on-data-screen

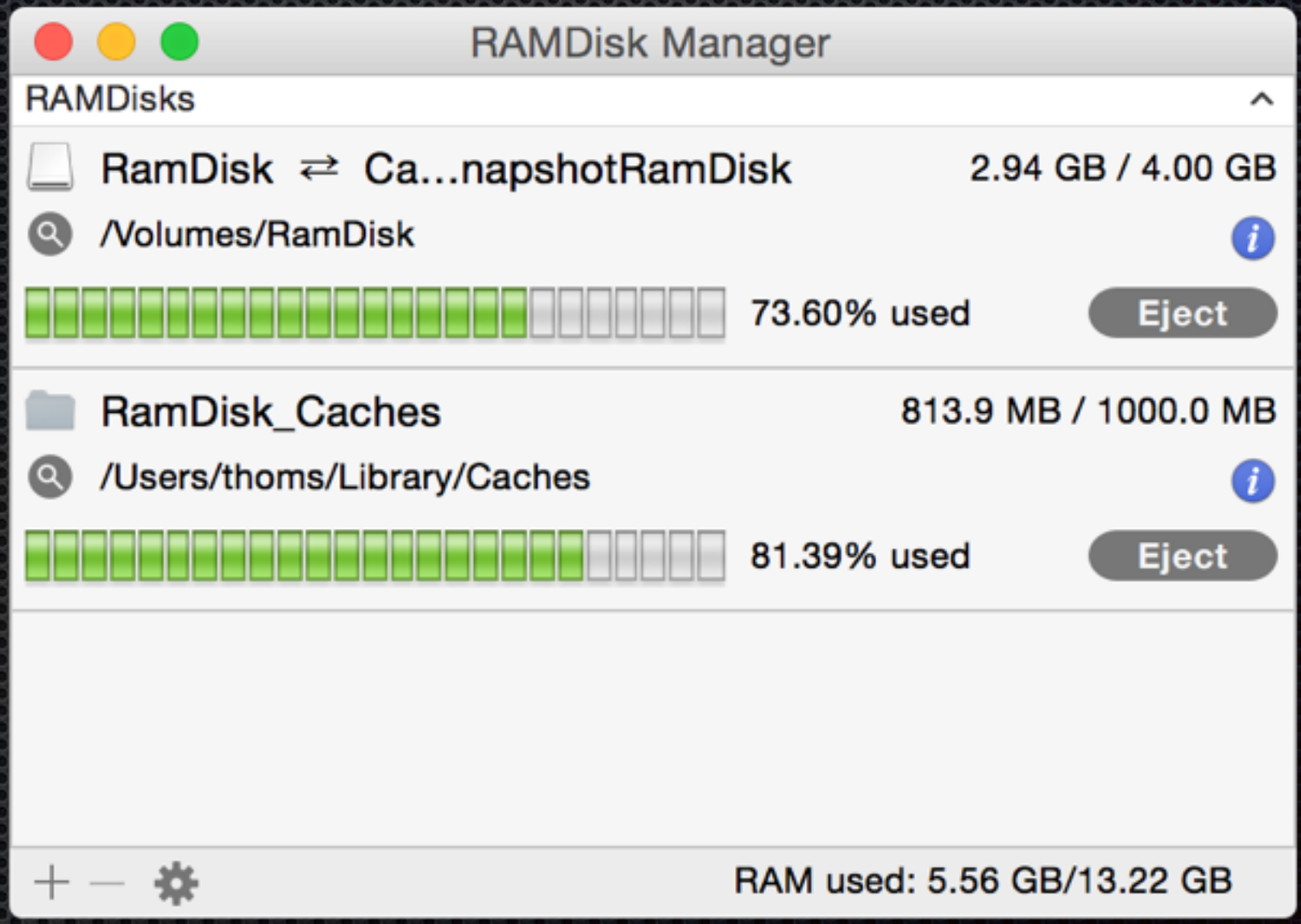
Disable Indexing

- OS indexes every changed file
- Consumes CPU & IO
- Usually senseless for **build output** and **workspace metadata**
- Can only be disabled easily for directories, not by file types
- Programmatically:
  - Spotlight: Flag File `.metadata_never_index`
  - Windows: `attrib.exe /s -i *.*`

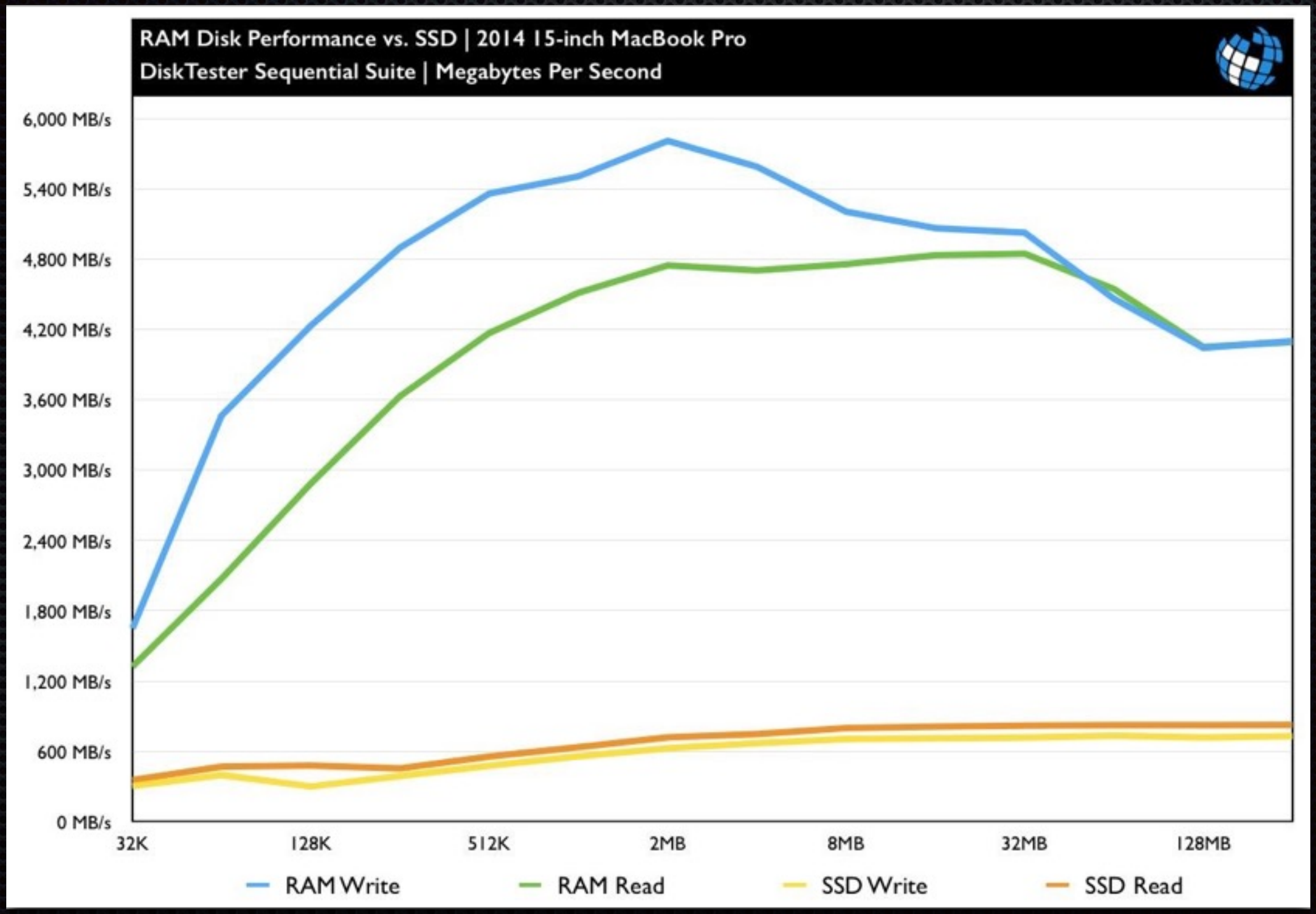


RAM Disk


Geräte
Entfernte CD…
RamDisk_Ca…
RamDisk
JetDrive


RAMDisk Manager
RAMDisks
RamDisk ⇄ Ca…napshotRamDisk
2.94 GB / 4.00 GB
/Volumes/RamDisk
73.60% used
Eject
RamDisk_Caches
813.9 MB / 1000.0 MB
/Users/thoms/Library/Caches
81.39% used
Eject
RAM used: 5.56 GB/13.22 GB

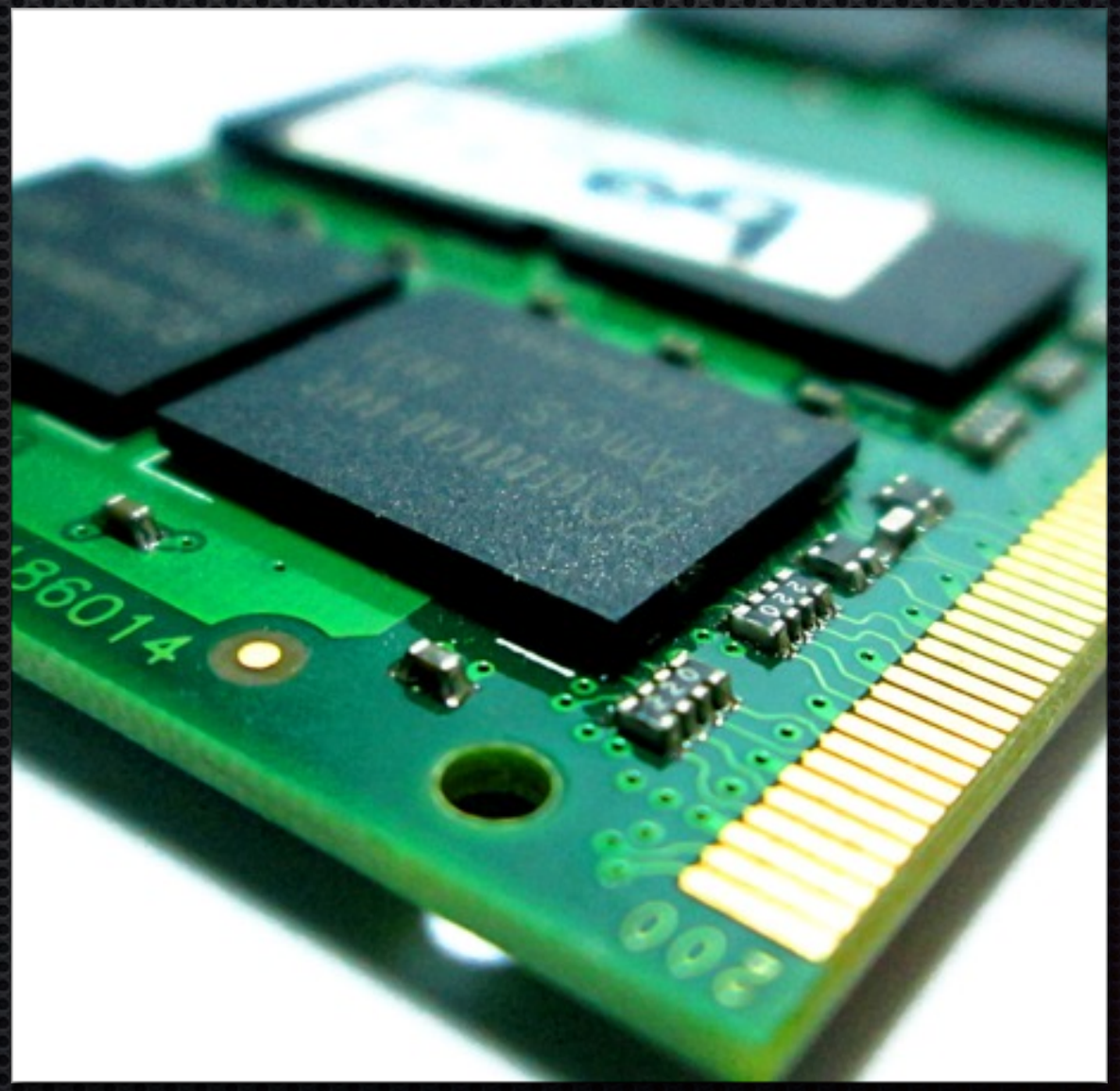
RAM Disk



http://www.tekrevue.com/ram-disk-2013-mac-pro-2014-macbook-pro/

RAM Disk

- What to store?
  - Read-only Data
    - JRE
    - Bundle Pool
  - Output folders
- Use Symbolic Links
- Store / Restore state to/from persistent storage



Eclipse Installation

Eclipse Installation

- Don’t store Eclipse / Workspace on Network Share
- Use a **current** Eclipse distribution
- Don’t install every feature any team member *might* use
  - Different feature set for different tasks?
  - Expensive: Mylyn, Subversion
  - Use **Oomph** setups or **Eclipse Profiles**

eclipse.ini Java Settings

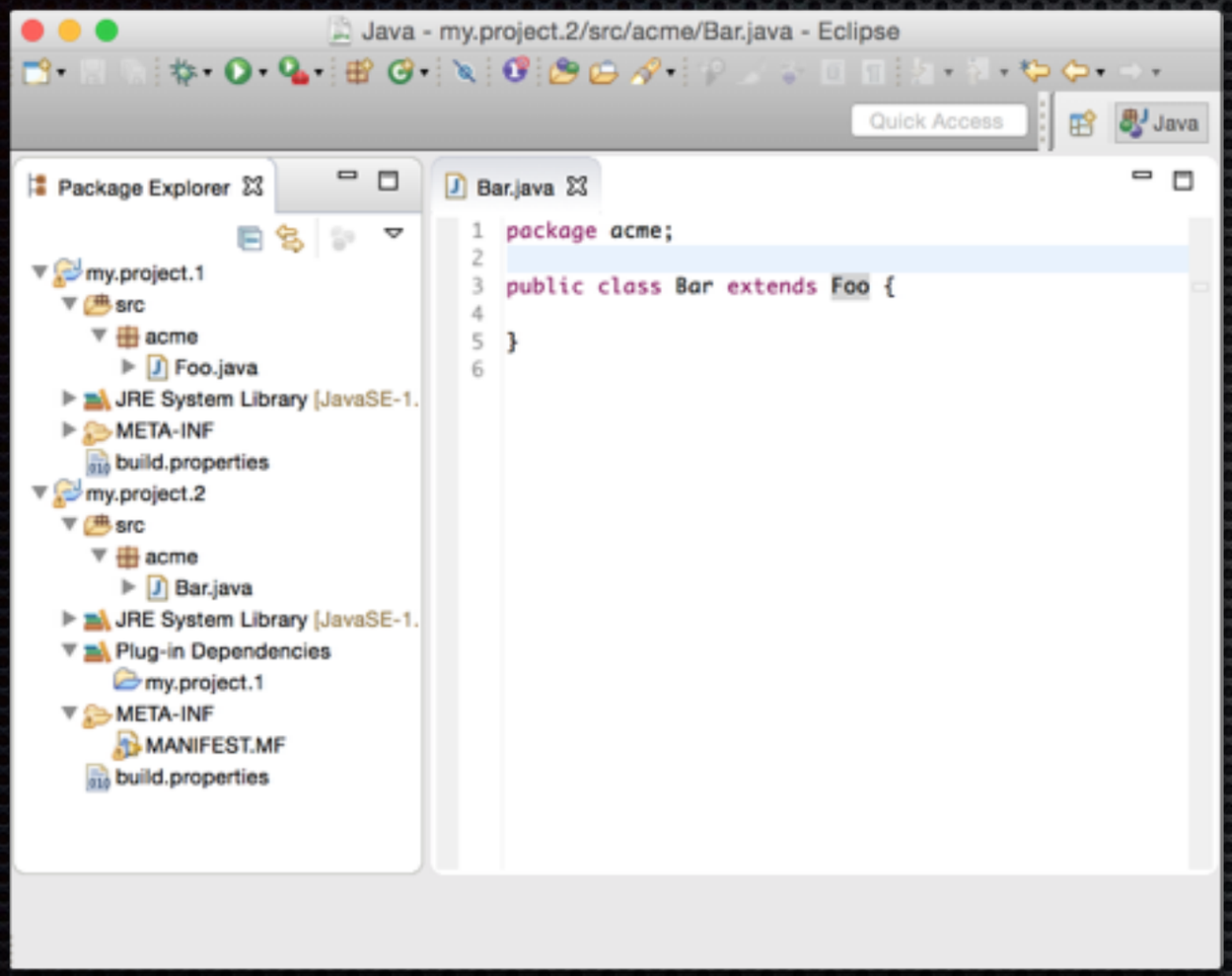
- Use the latest JRE
- Use Server VM
- Use enough Heap
- Increase Young Generation Space
- Disable Class Validation
- Turn on Compiler Optimizations
- Activate Parallel GC

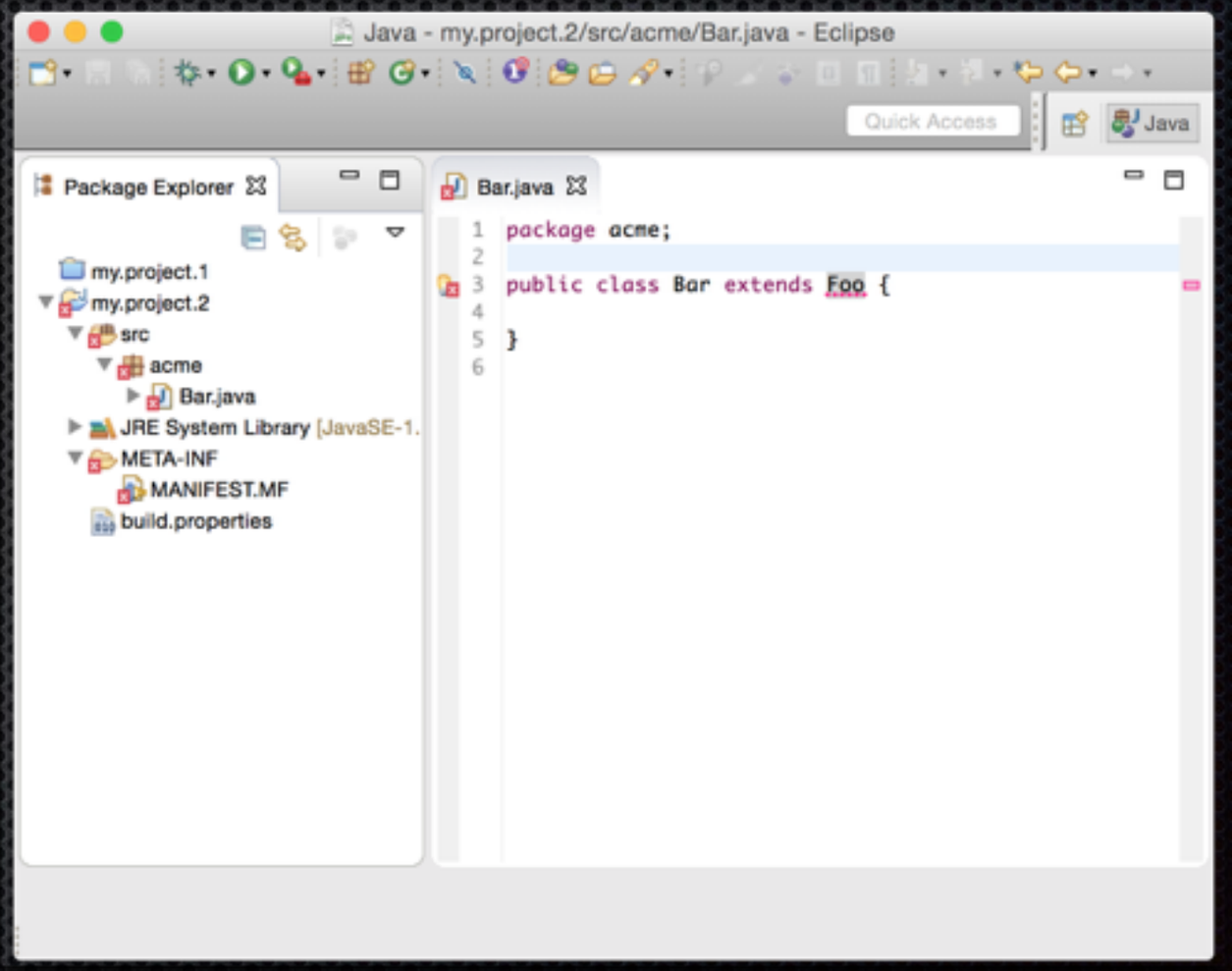
```
-vm
/Library/Java/JavaVirtualMachines/
jdk1.8.0_66.jdk/Contents/Home/bin
-server
-Xms512m
-Xmx2g
-Xmn512m
-Xverify:none
-XX:+AggressiveOpts
-XX:+UseParallelGC
```

Workspace Resources

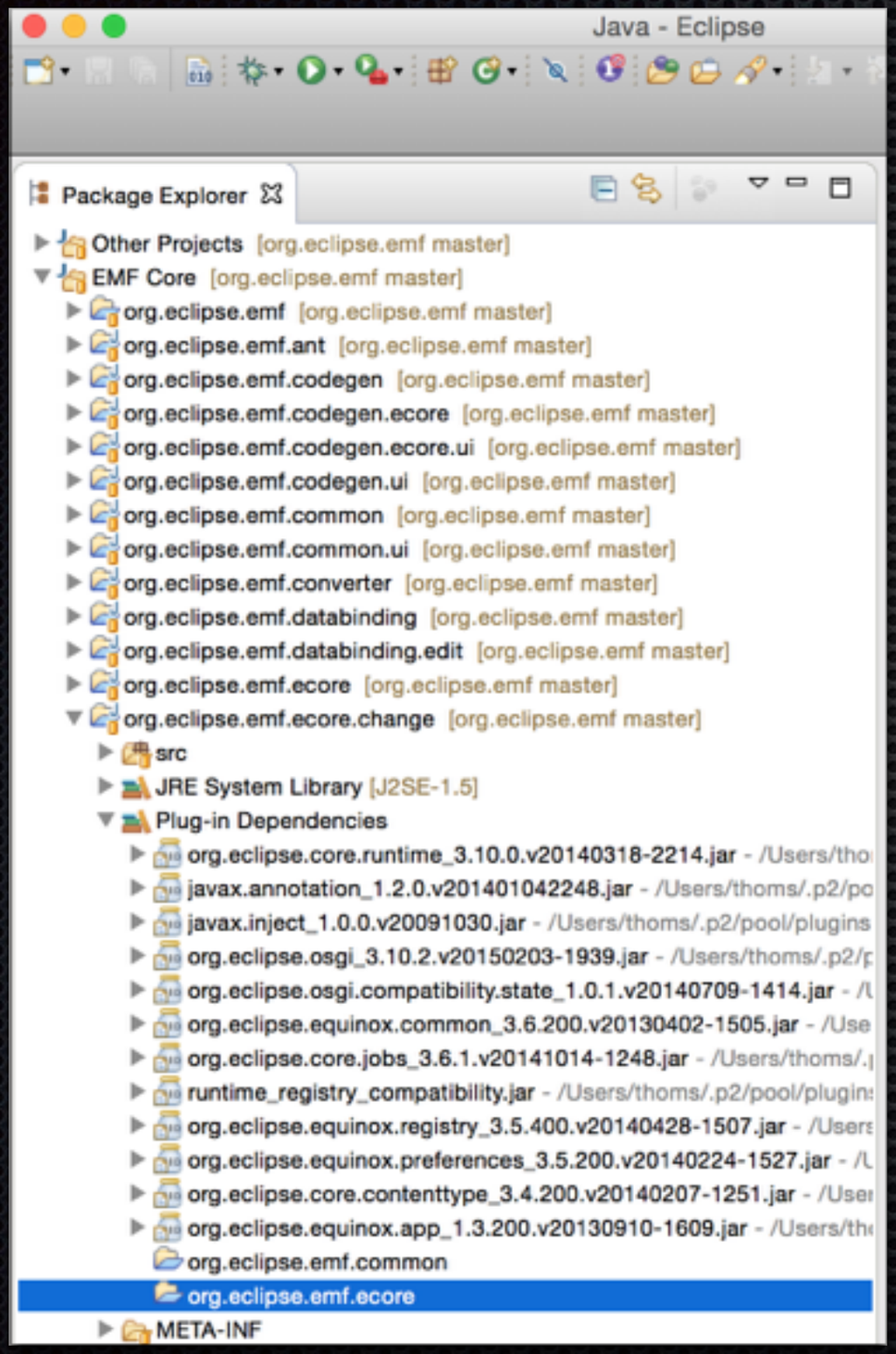
- Close unused **projects**
- Close unused **views**
- Close unused **perspectives**

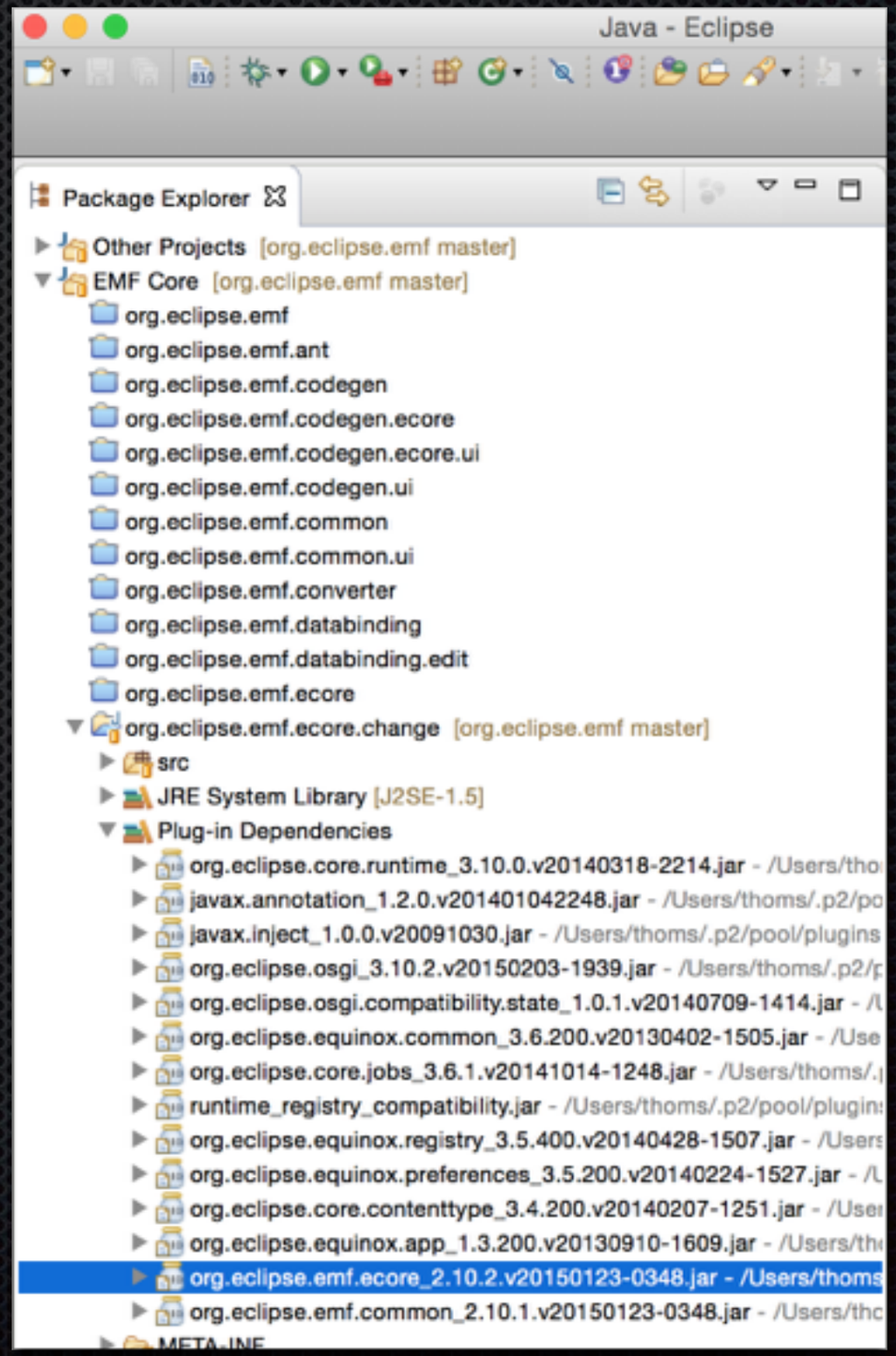
Make Projects *Closeable*



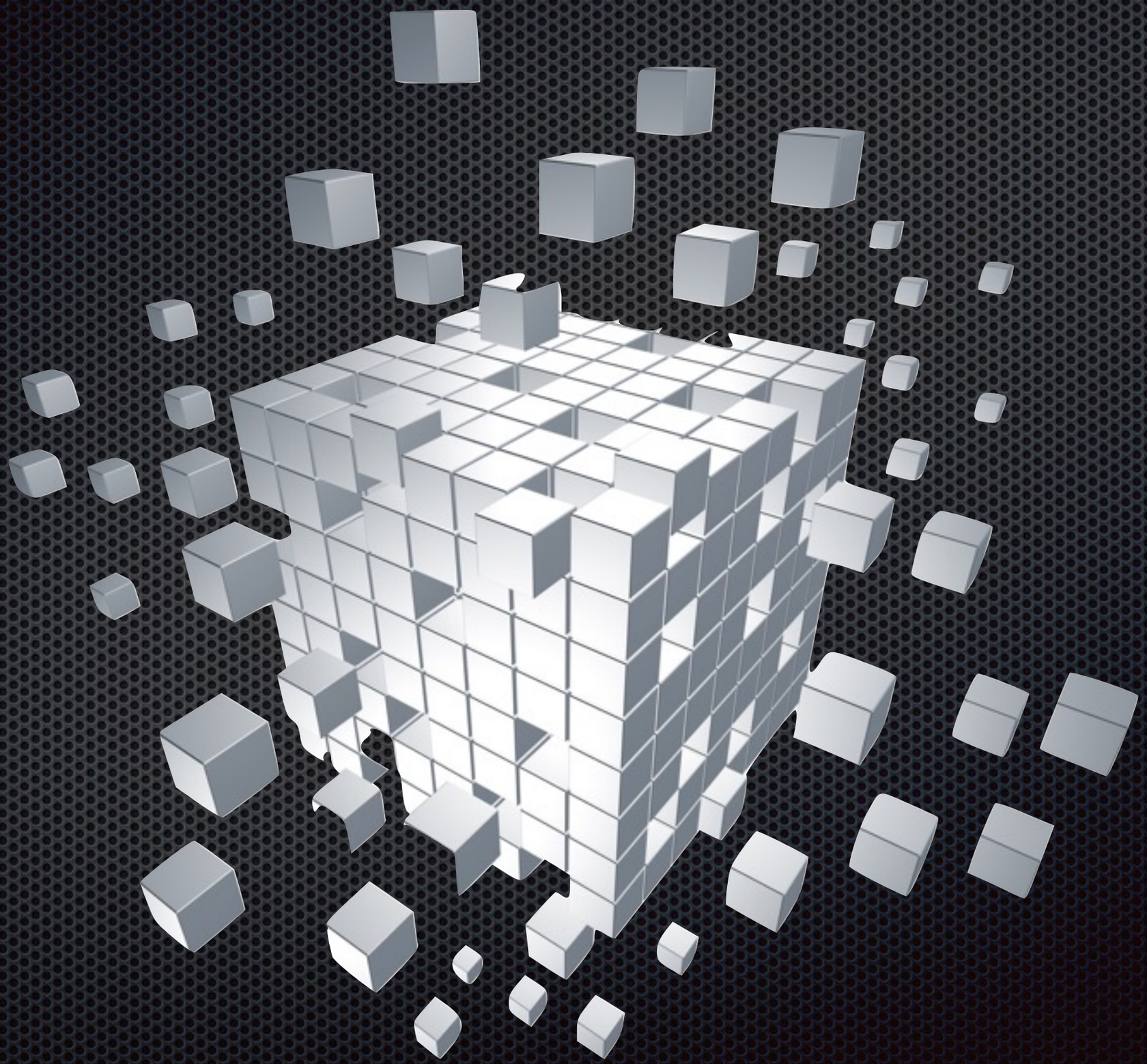


Make Projects *Closeable*

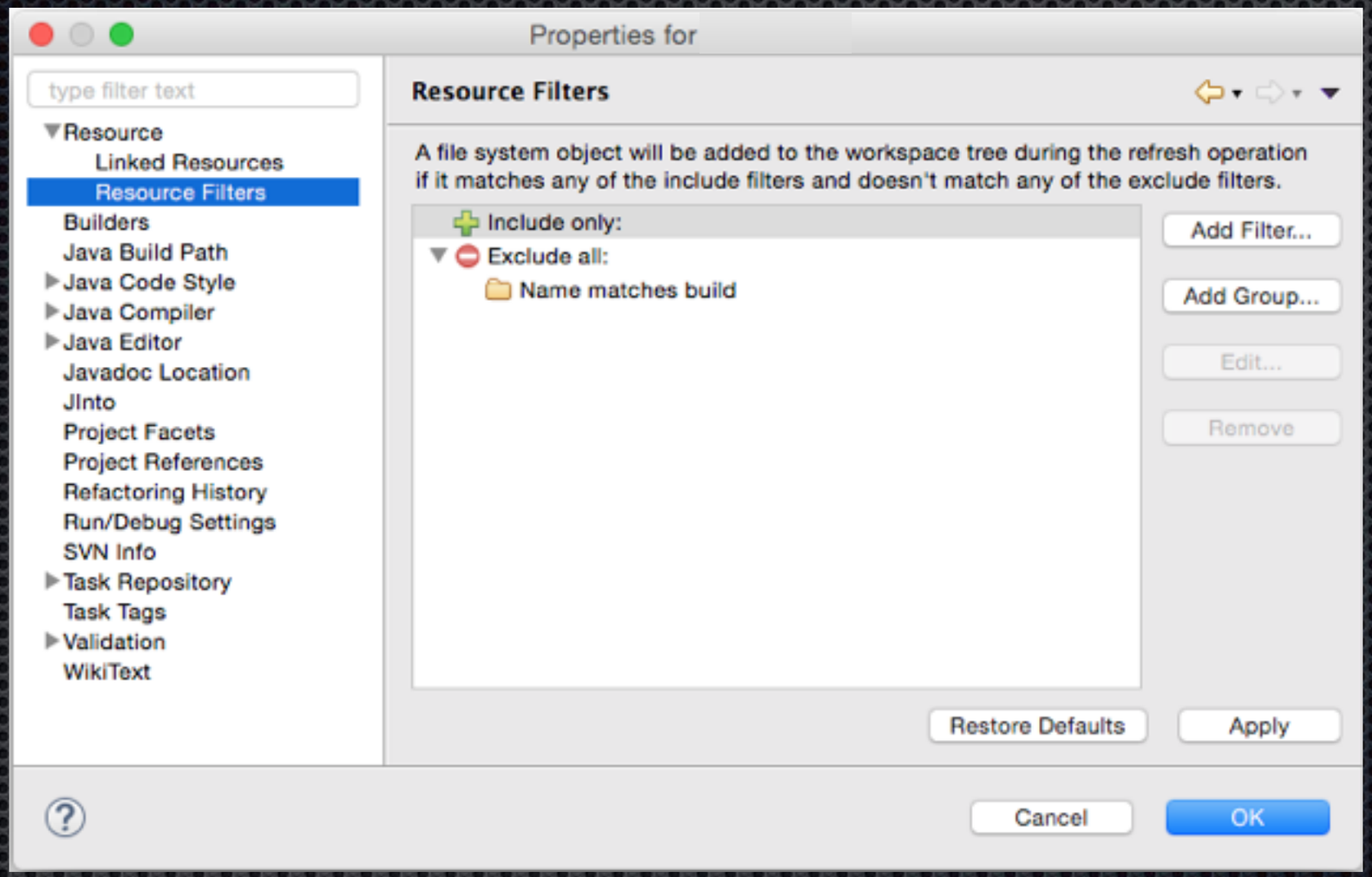




Split Projects

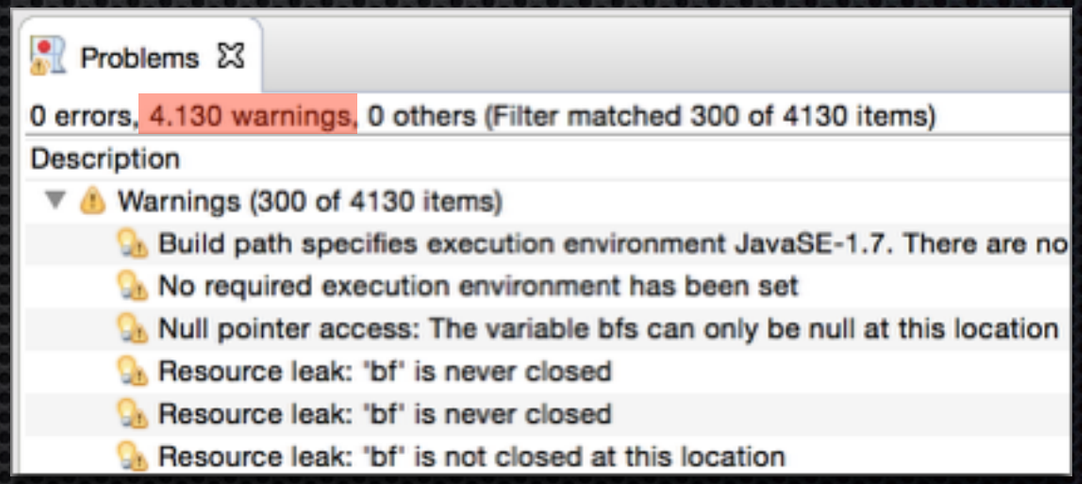


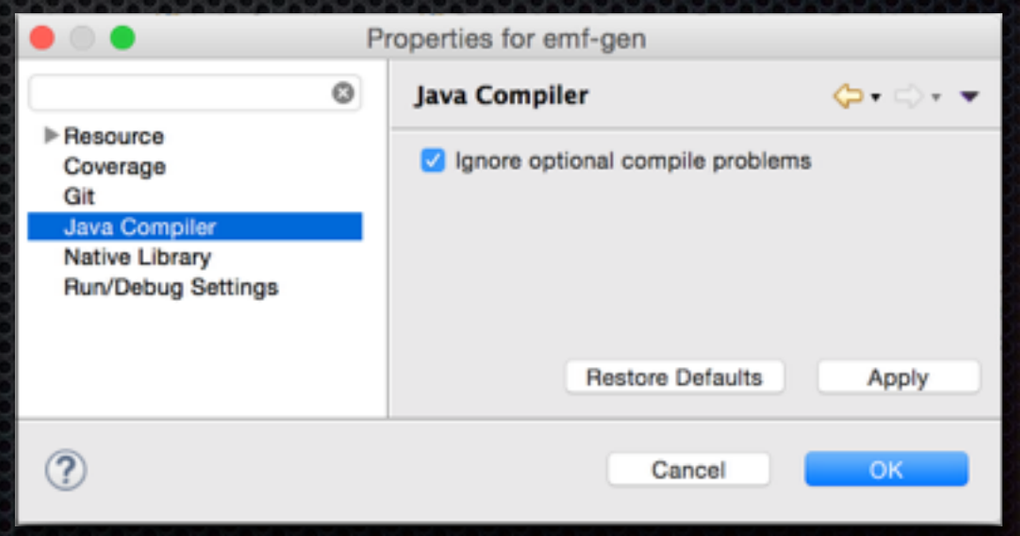
Resource Filters



Compiler Warnings

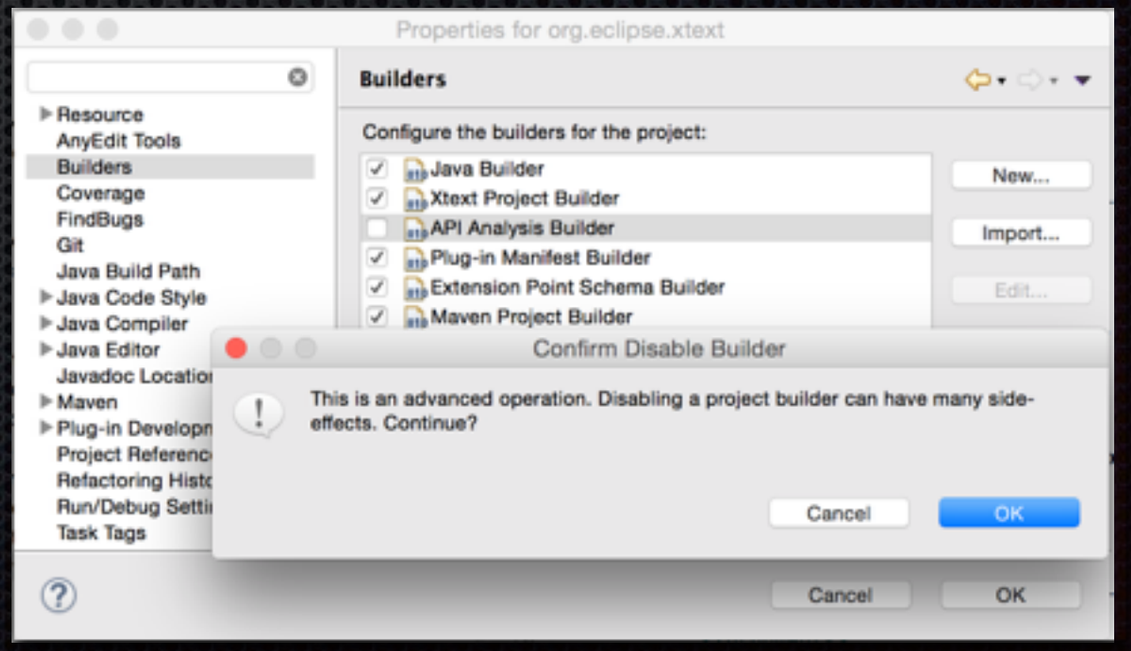
- Reduce compiler warnings
- If you ignore them anyway, filter them
- Ignore optional warnings on selected source folders (e.g. generator output)





Plug-in Development

- Target Platform
  - Mirror/Aggregate Public p2 Repositories
    b3 Aggregator, Buckminster, p2 Tools
  - Deploy on Local Network / Repository Manager
- Launch Config for Eclipse Application
  - Required bundles only
- Disable API Tooling during development
  Enable on demand, in CI, before milestones



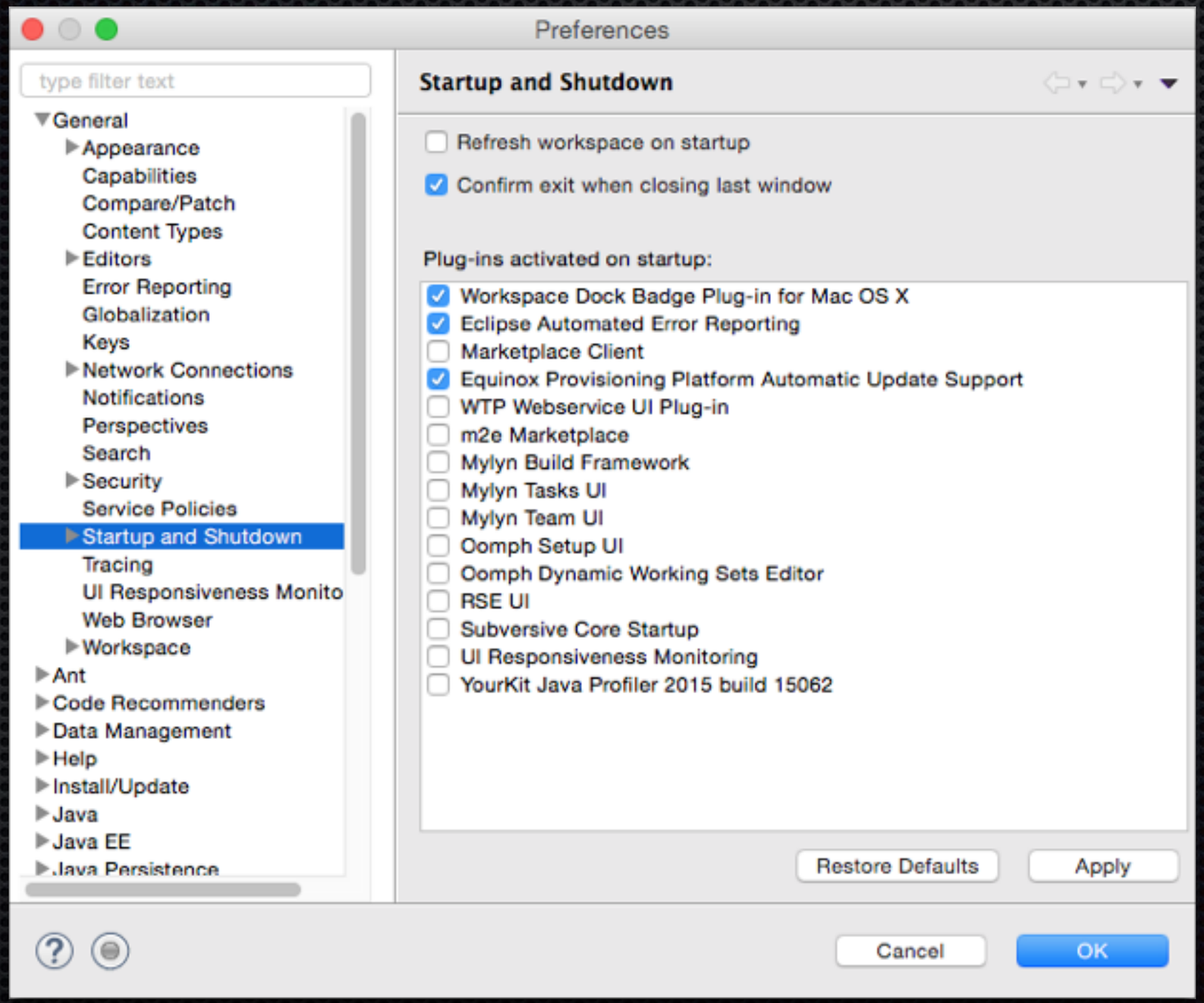
Cleanup Metadata

- Clean JDT index
  `<WS>\.metadata\.plugins\org.eclipse.jdt.core`
- Resource History
  `<WS>\.metadata\.plugins`
  `\org.eclipse.core.resources\.history`
- PDE caches / Bundle Pool
  `<WS>\.metadata\.plugins\org.eclipse.pde.core`
- or even fresh workspace

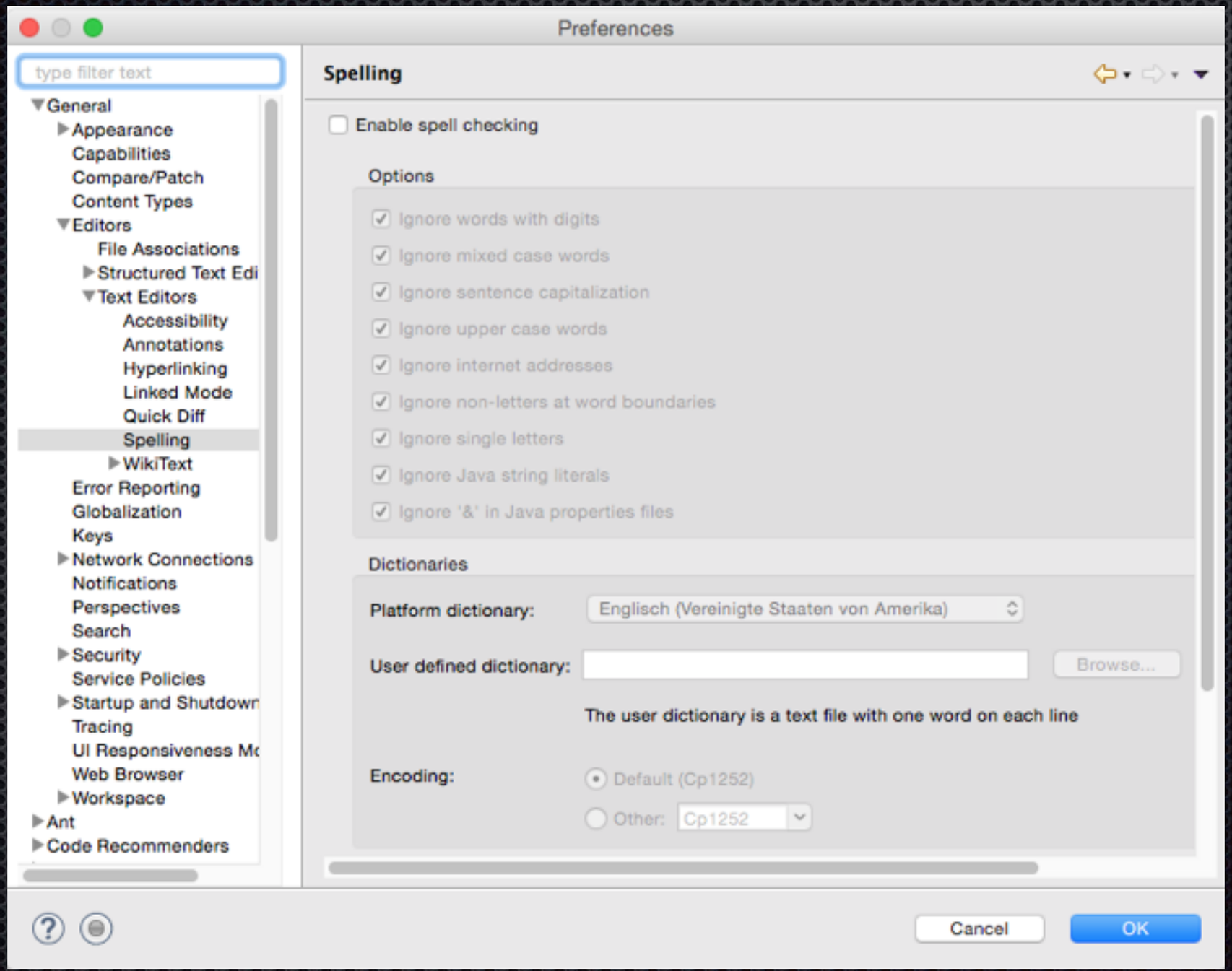
http://www.stockvault.net/photo/133998/recycling-grunge-sign

Preferences

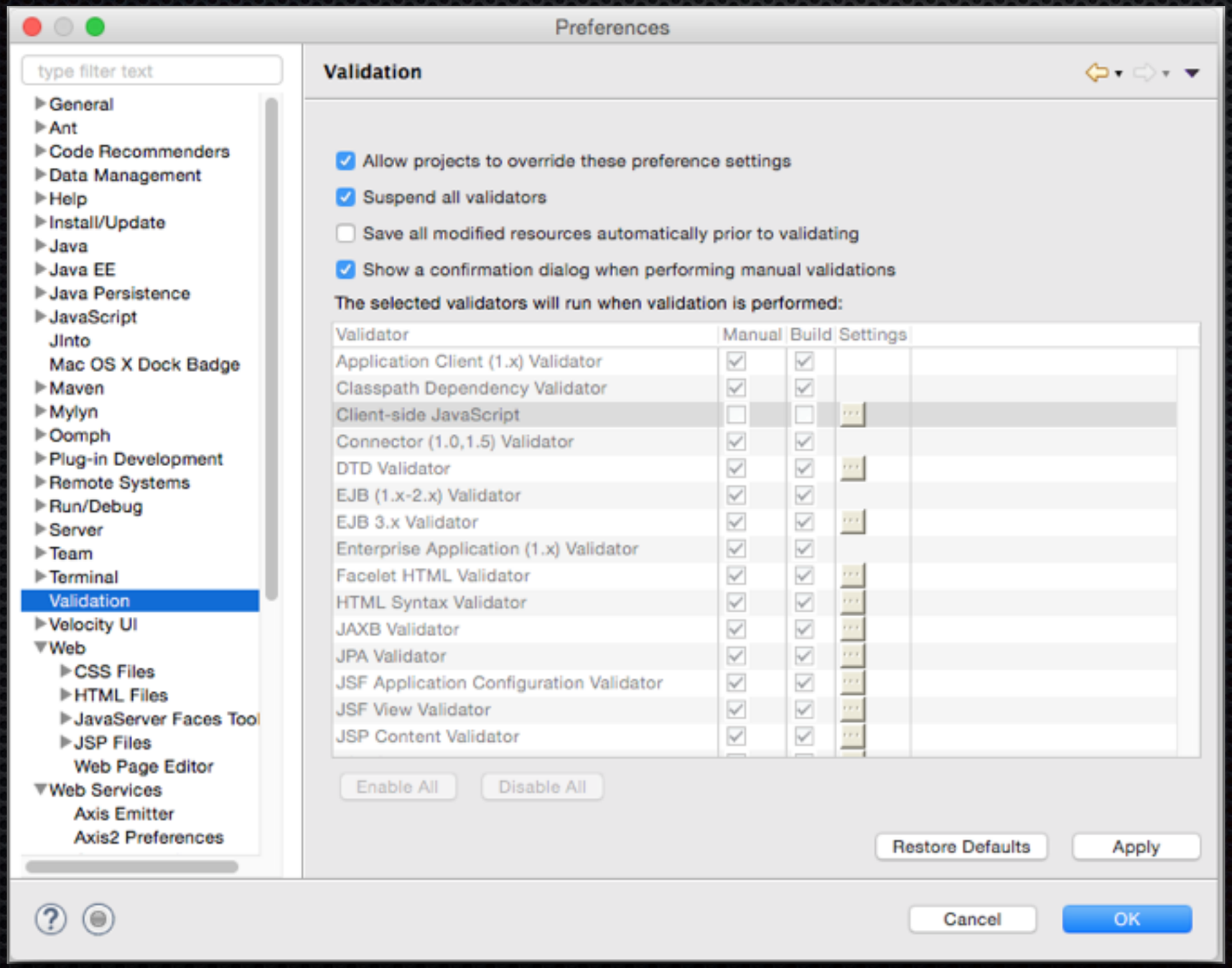
Deactivate Startup Plugins



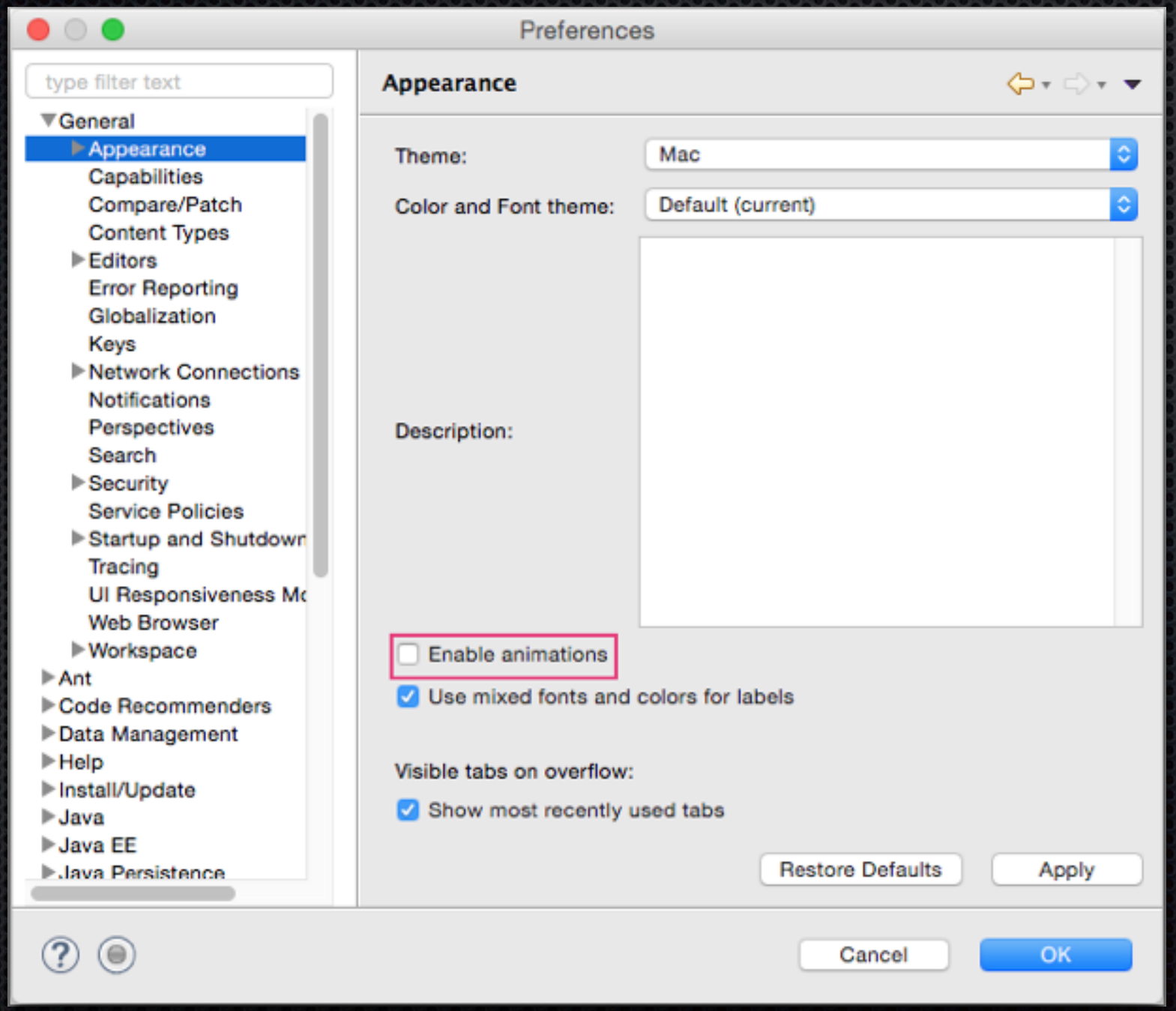
Disable Spell Checking



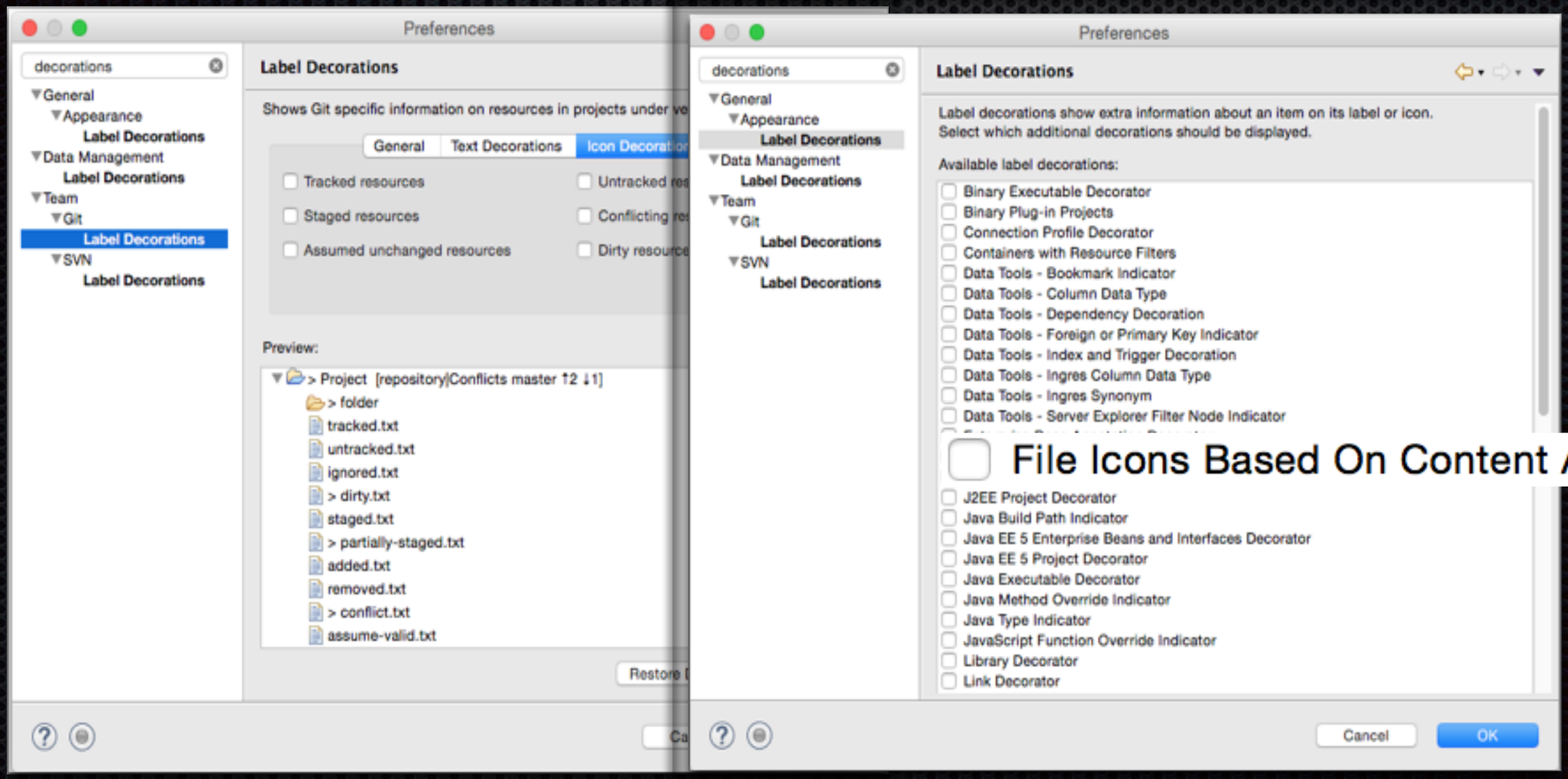
Suspend Validators



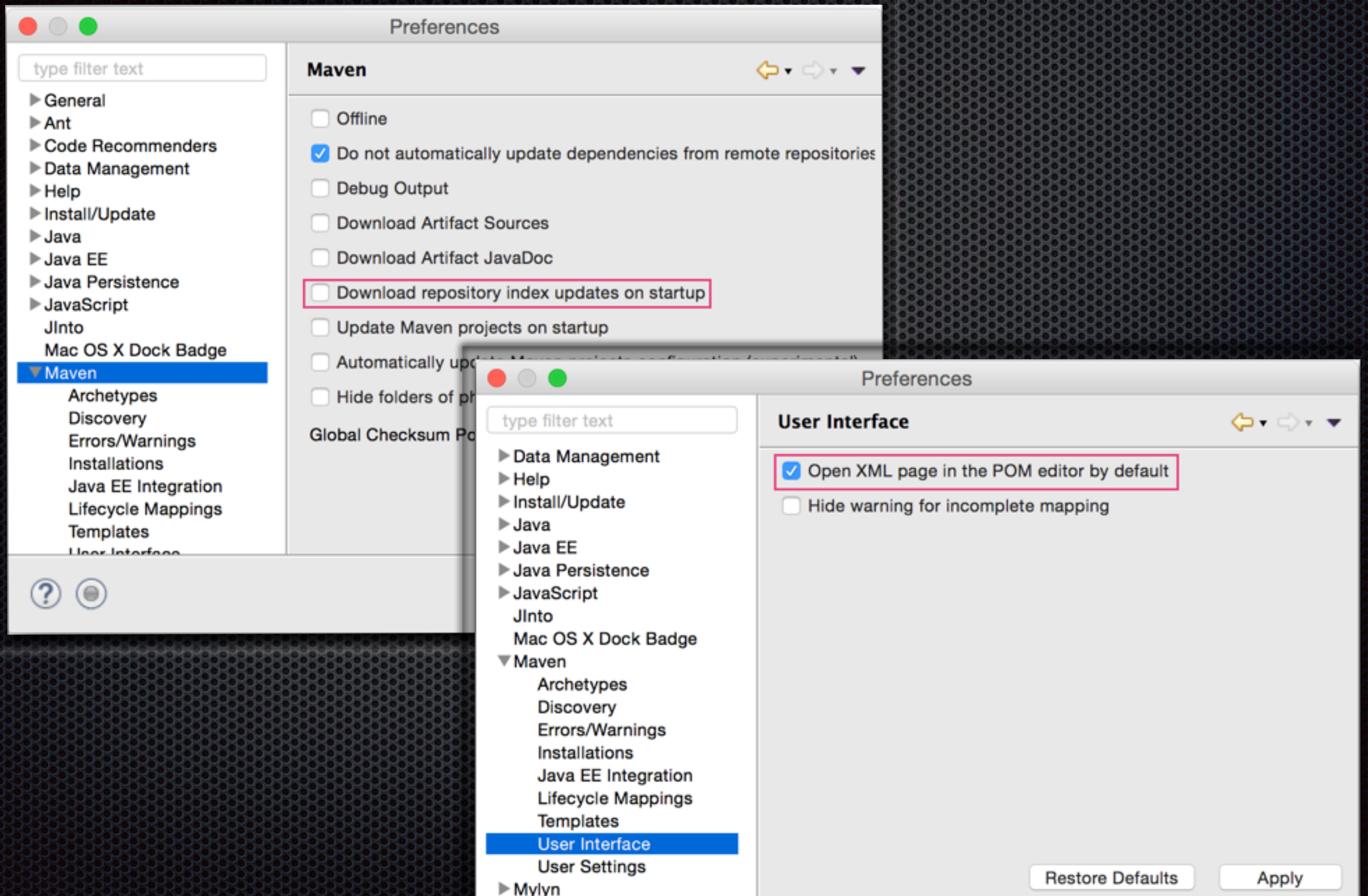
Disable Animations



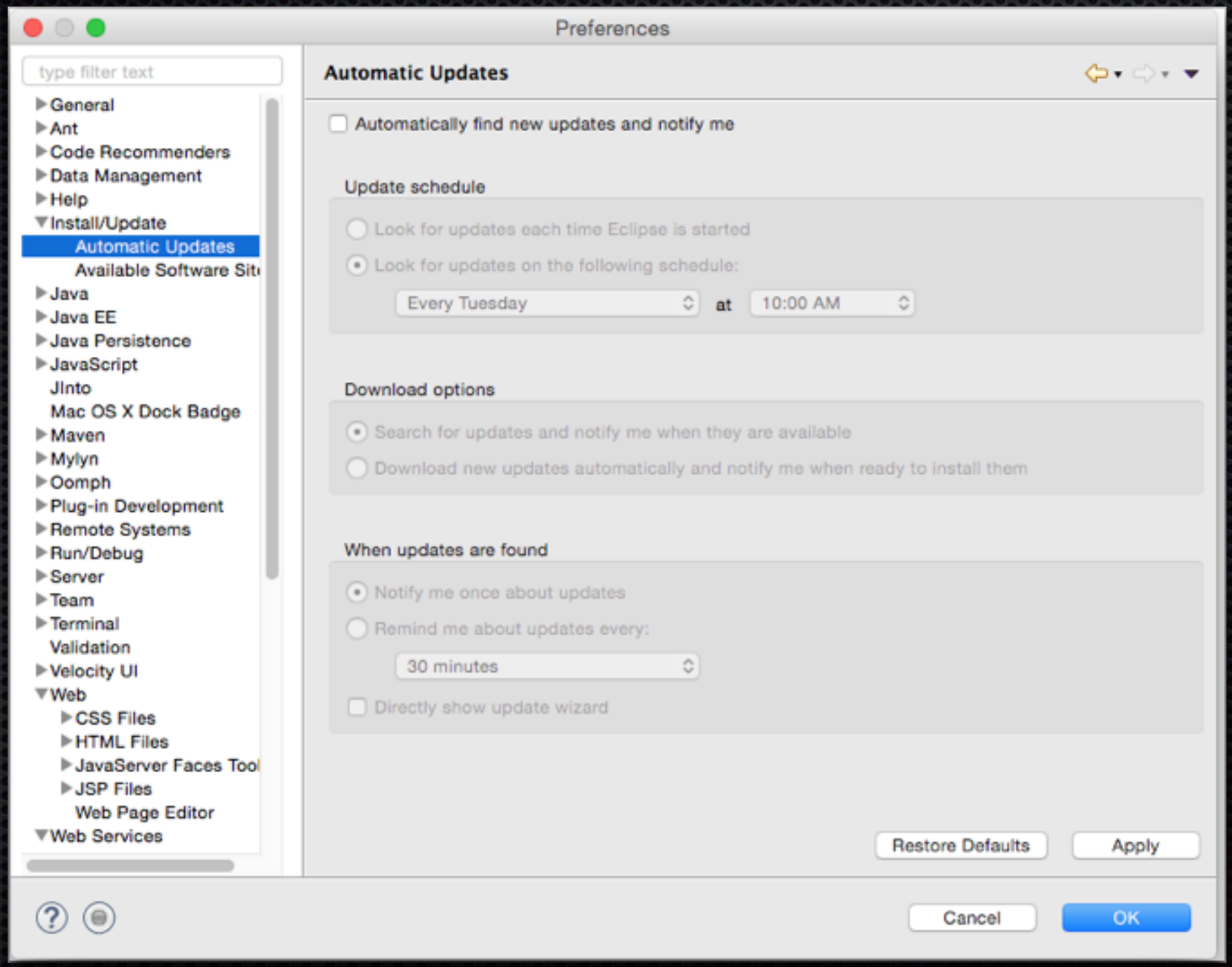
Disable Decorations



Maven


Preferences
type filter text
General
Ant
Code Recommenders
Data Management
Help
Install/Update
Java
Java EE
Java Persistence
JavaScript
JInto
Mac OS X Dock Badge
Maven
Archetypes
Discovery
Errors/Warnings
Installations
Java EE Integration
Lifecycle Mappings
Templates
User Interface
Maven
Offline
Do not automatically update dependencies from remote repositories
Debug Output
Download Artifact Sources
Download Artifact JavaDoc
Download repository index updates on startup
Update Maven projects on startup
Hide folders of ph
Global Checksum Po
Preferences
type filter text
Data Management
Help
Install/Update
Java
Java EE
Java Persistence
JavaScript
JInto
Mac OS X Dock Badge
Maven
Archetypes
Discovery
Errors/Warnings
Installations
Java EE Integration
Lifecycle Mappings
Templates
User Interface
User Settings
Mylyn
User Interface
Open XML page in the POM editor by default
Hide warning for incomplete mapping
Restore Defaults
Apply

Disable Check for Updates



Profile Eclipse Tasks

Questions ?

http://www.stockvault.net/photo/122646/school-boy-

eclipsecon Europe

Ludwigsburg, Germany, 3 - 5 November 2015

Evaluate the sessions at www.eclipsecon.org

+1 0 -1

References

- “7 Tips to Speed Up Eclipse”
  http://www.nicolasbize.com/blog/7-tips-to-speed-up-eclipse/
- „15 Useful Tips - Speed Up Eclipse To An Ultra Fast IDE"
  http://www.fromdev.com/2013/05/Speed-Up-Eclipse.html
- „How to quickly make eclipse faster“
  http://howtodoinjava.com/2014/04/05/how-to-quickly-make-eclipse-faster/
- “Benchmarking G1 and other Java 7 Garbage Collectors”
  http://blog.mgm-tp.com/2013/12/benchmarking-g1-and-other-java-7-garbage-collectors/
- Alex Blewitt: “Eclipse start optimisation”
  http://alblue.bandlem.com/2015/09/eclipse-optimisation-part-2.html
- Lars Vogel: “Eclipse Activator startup sins – Tracing the startup time”
  http://blog.vogella.com/2015/09/16/eclipse-activator-startup-sins-tracing-the-startup-time/