The Apache Lucene Infrastructure: What's going on in development behind the scenes?

Uwe Schindler

Apache Lucene Committer & PMC Chair

uschindler@apache.org

http://www.thetaphi.de, http://blog.thetaphi.de

@ThetaPh1

SD DataSolutions GmbH, Wätjenstr. 49, 28213 Bremen, Germany

Tel: +49 421 40889785-0, http://www.sd-datasolutions.de

Leading the Wave of Open Source

My Background

- **Committer** and **PMC chair** of **Apache Lucene and Solr** - main focus is on development of Lucene Core.
- Implemented **fast numerical search** and maintaining the **new attribute-based text analysis API**. Well known as *Generics and Sophisticated Backwards Compatibility Policeman*.
- Working as **consultant** and software architect at **SD DataSolutions GmbH** in Bremen, Germany. The main task is maintaining PANGAEA (Publishing Network for Geoscientific & Environmental Data) where I implemented the portal's geo-spatial retrieval functions with Apache Lucene Core and Elasticsearch.

Agenda

- Motivation
- The Apache Lucene infrastructure
- Tools
- Policeman Jenkins
- Automated release testing

MOTIVATION

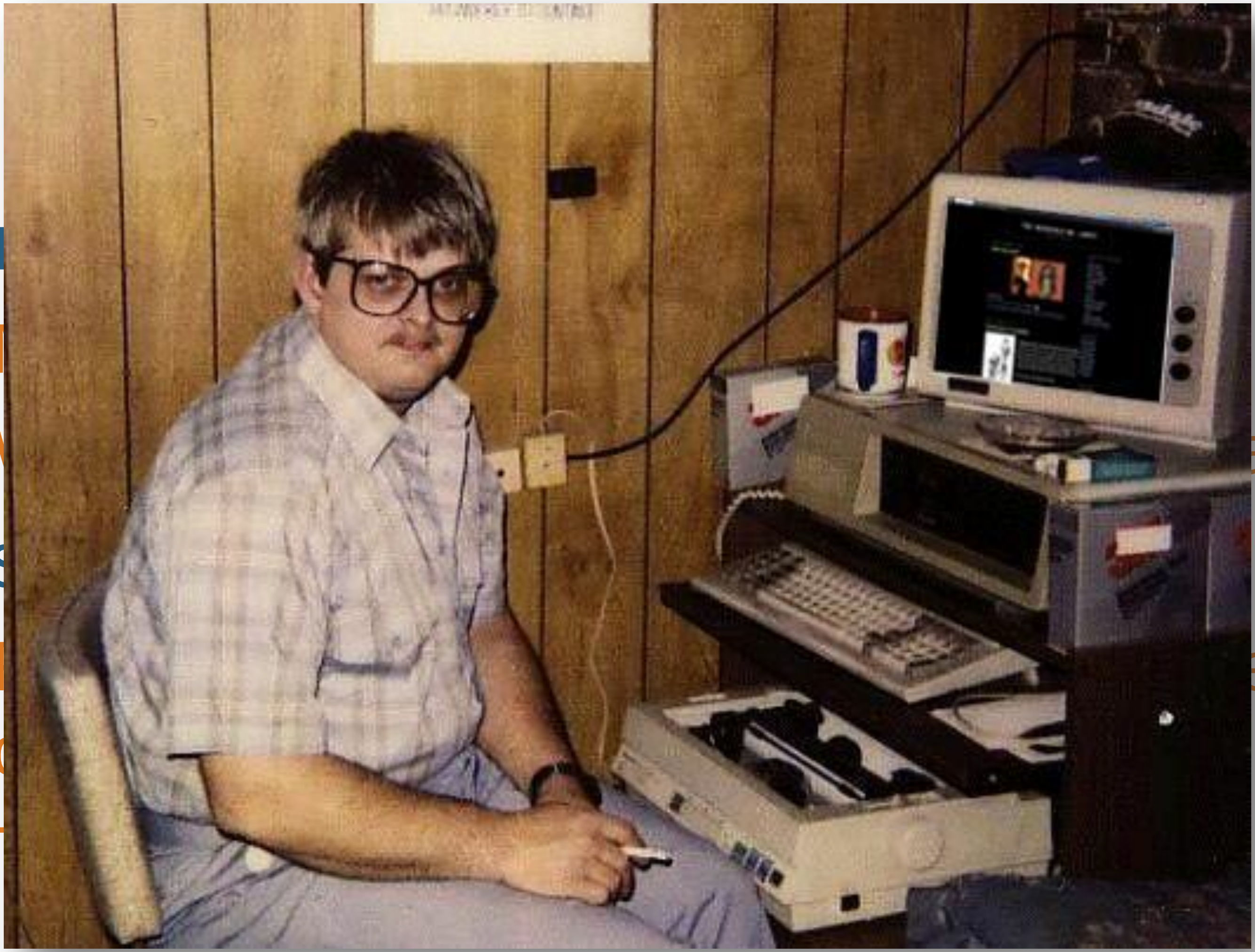
Motivation

- Release early, release often!
  - Lot’s of new features in **LUCENE** and **SOLR** evolving
  - Users should get them as soon as possible
- Sister projects like Elasticsearch
  - are very active
  - share committers
  - rely on new features and bug fixes available in time

Problems

- Burden for release manager
  - Many small details to take care of
  - Wiki page about release got larger and larger!
- Risks with non-automated checks
  - RM or voting PMC members miss to carefully check all stuff
  - Time intensive

- Bu
  - I
  - V arger!
- Ris
  - I efully c
  - T



THE APACHE LUCENE INFRASTRUCTURE

Build system

- **Apache Ant**
- **Apache Ivy** for dependencies

- Multi-Module structure
  – Inter-module deps not yet ideal
- Why not **Apache Maven**?
  – More flexibility with **Ant** for release process!

Custom Plugins in the Ant build system

TOOLS

```
if ("hiho".equals(
  myString.toLowerCase()
))
```

```
(GregorianCalendar) Calendar.getInstance()
```

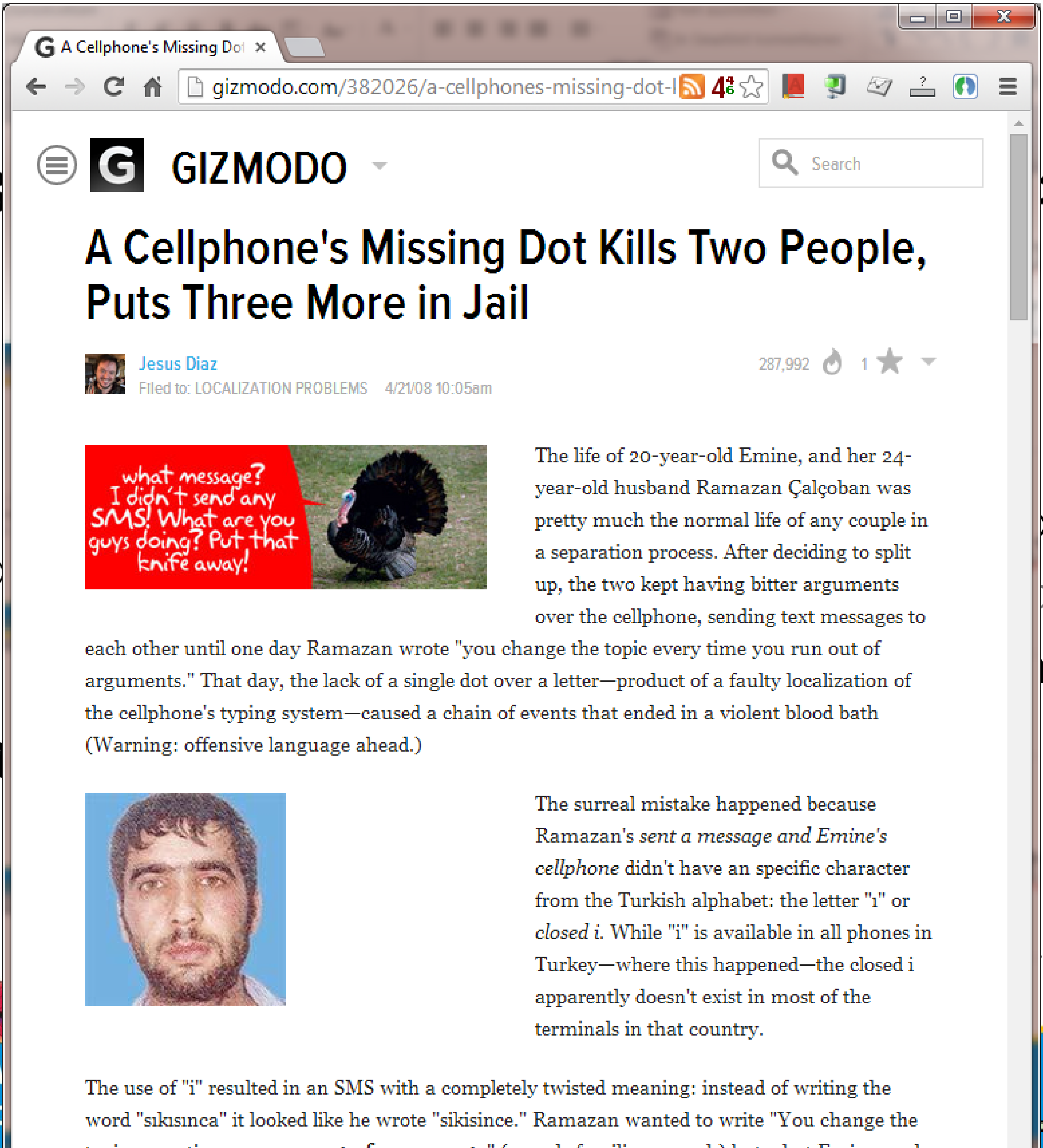
```
} catch (Exception e) {
  // Eclipse autogenerat
  e.printStackTrace();
}
```

```
new InputStreamReader(is)
```

if ("
mys
))
(Grego

e()
n e) {
generat
ce();
s)

Leading the W
of Open Sourc



A Cellphone's Missing Dot

gizmodo.com/382026/a-cellphones-missing-dot-l

GIZMODO

Search

A Cellphone's Missing Dot Kills Two People, Puts Three More in Jail

Jesus Diaz
Filed to: LOCALIZATION PROBLEMS 4/21/08 10:05am

287,992 1

what message? I didn't send any SMS! What are you guys doing? Put that knife away!

The life of 20-year-old Emine, and her 24-year-old husband Ramazan Çalçoban was pretty much the normal life of any couple in a separation process. After deciding to split up, the two kept having bitter arguments over the cellphone, sending text messages to each other until one day Ramazan wrote "you change the topic every time you run out of arguments." That day, the lack of a single dot over a letter—product of a faulty localization of the cellphone's typing system—caused a chain of events that ended in a violent blood bath (Warning: offensive language ahead.)

The surreal mistake happened because Ramazan's *sent a message and Emine's cellphone* didn't have an specific character from the Turkish alphabet: the letter "ı" or *closed i*. While "i" is available in all phones in Turkey—where this happened—the closed i apparently doesn't exist in most of the terminals in that country.

The use of "i" resulted in an SMS with a completely twisted meaning: instead of writing the word "sıkısınca" it looked like he wrote "sikisince." Ramazan wanted to write "You change the

```
if ("hiho".equals(
  myString.toLowerCase()
))
```

```
(GregorianCalendar) Calendar.getInstance()
```

```
} catch (Exception e) {
  // Eclipse autogenerat
  e.printStackTrace();
}
```

```
new InputStreamReader(is)
```

Forbidden-APIs

- Keep Lucene free of „unsafe“ APIs:
  - Locale sensitive calls using system default
  - Use of platform‘s default charset
  - Same applies to timezones and default use of non-Gregorian calendars (e.g., Thai).
- Other no-gos:
  - Printing to System.out/err
  - Creating threads without name

Forbidden-APIs

- Why not PMD / FindBugs?
  - Incomplete
  - Slow
  - Hard to add custom “forbidden” signatures
- Just a “simple” tool to analyze bytecode and trigger on method signatures
  - Also works with classes in general and annotations
  - Compatible to Java 8: Lambdas supported!

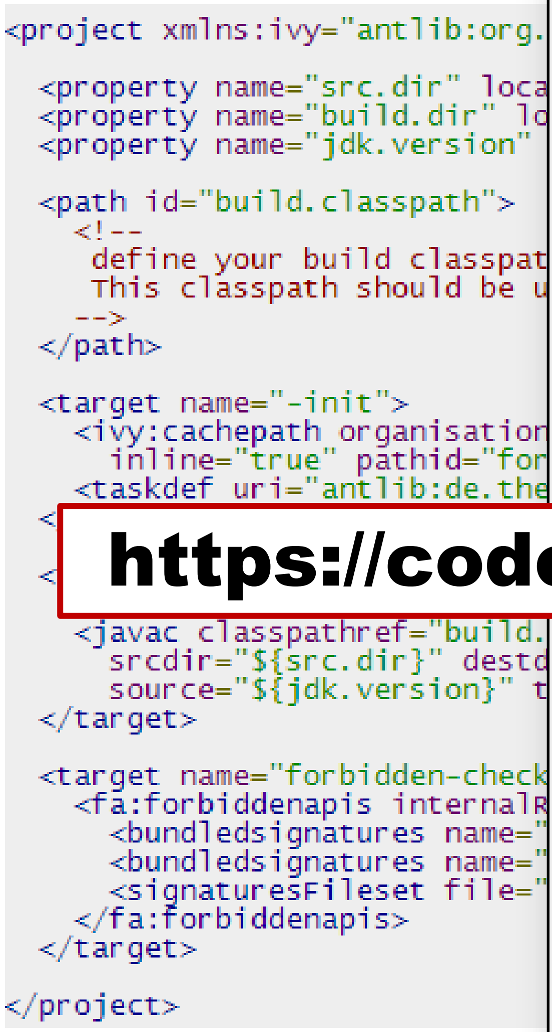
Forbidden-APIs

- Used by other projects, too:
  - Elasticsearch (with Maven)
- Available through Maven Central
  - Ant plugin (e.g., used via Ivy)
  - Maven Mojo
  - Various signature files included: “unsafe”, JDK deprecated methods, `System.out`

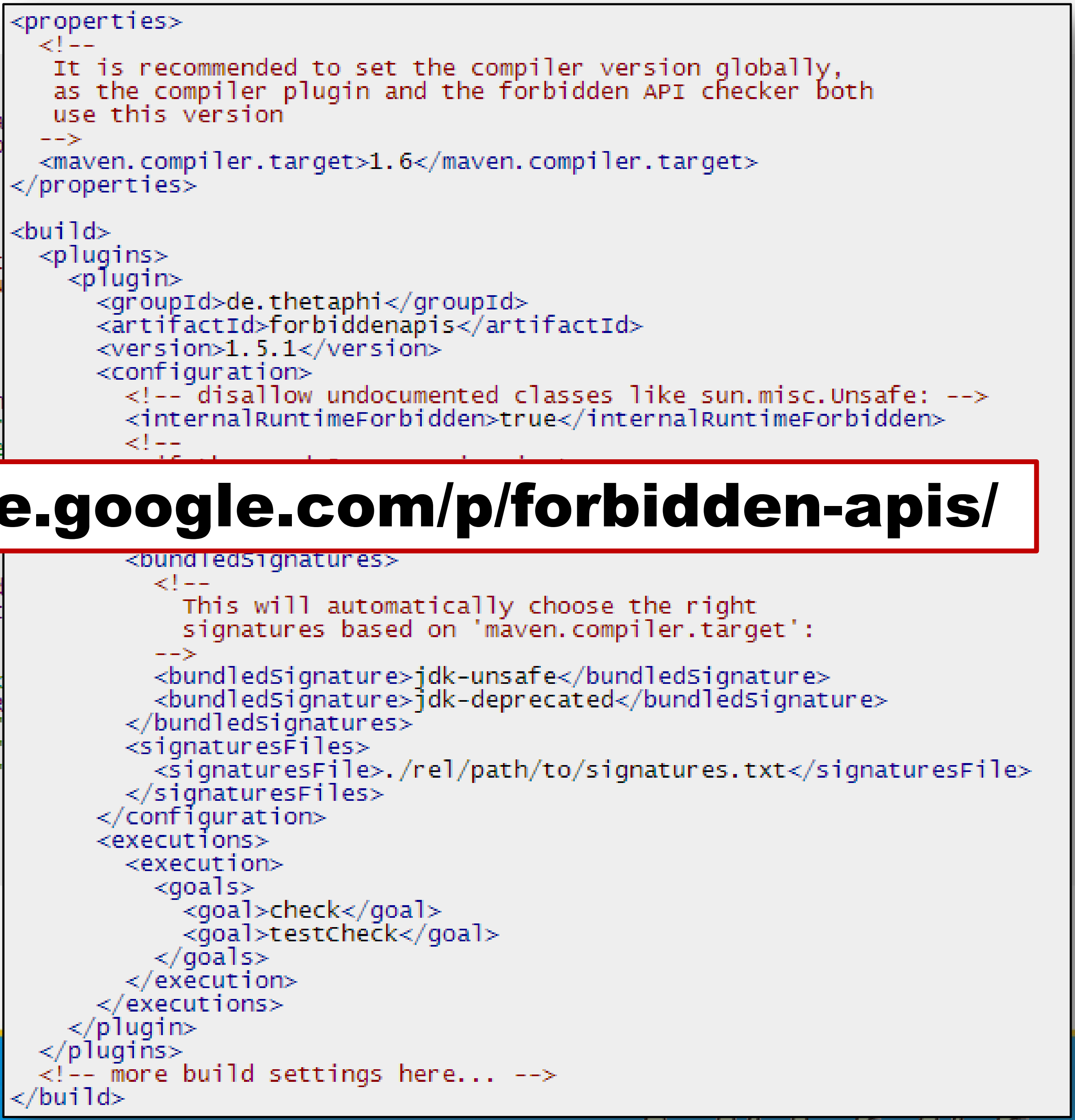
```
<project xmlns:ivy="antlib:org.apache.ivy.ant" xmlns:fa="antlib:de.thetaphi.forbiddenapis">

  <property name="src.dir" location="..."/>
  <property name="build.dir" location="..."/>
  <property name="jdk.version" value="1.6"/>

  <path id="build.classpath">
    <!--
     define your build classpath here, so all referenced JAR files can be found.
     This classpath should be used by javac and the forbidden API checker.
    -->
  </path>

  <target name="-init">
    <ivy:cachepath organisation="de.thetaphi" module="forbiddenapis" revision="1.5.1"
      inline="true" pathid="forbiddenapis.classpath"/>
    <taskdef uri="antlib:de.thetaphi.forbiddenapis" classpathref="forbiddenapis.classpath"/>
  </target>

  <target name="compile" depends="-init">
    <mkdir dir="${build.dir}"/>
    <javac classpathref="build.classpath"
      srcdir="${src.dir}" destdir="${build.dir}"
      source="${jdk.version}" target="${jdk.version}"/>
  </target>

  <target name="forbidden-checks" depends="compile">
    <fa:forbiddenapis internalRuntimeForbidden="true" classpathref="build.classpath" dir="${build.dir}">
      <bundledsignatures name="jdk-unsafe-${jdk.version}"/>
      <bundledsignatures name="jdk-deprecated-${jdk.version}"/>
      <signaturesFileset file="path/to/signatures.txt"/>
    </fa:forbiddenapis>
  </target>

</project>
```

```
<project xmlns:ivy="antlib:org.

  <property name="src.dir" loca
  <property name="build.dir" lo
  <property name="jdk.version"

  <path id="build.classpath">
    <!--
     define your build classpat
     This classpath should be u
    -->
  </path>

  <target name="-init">
    <ivy:cachepath organisation
      inline="true" pathid="for
    <taskdef uri="antlib:de.the
  </target>

  <target name="compile" depend
    <mkdir dir="${build.dir}"/>
    <javac classpathref="build.
      srcdir="${src.dir}" destd
      source="${jdk.version}" t
  </target>

  <target name="forbidden-check
    <fa:forbiddenapis internalR
      <bundledsignatures name="
      <bundledsignatures name="
      <signaturesFileset file="
    </fa:forbiddenapis>
  </target>

</project>
```

```
<properties>
  <!--
   It is recommended to set the compiler version globally,
   as the compiler plugin and the forbidden API checker both
   use this version
  -->
  <maven.compiler.target>1.6</maven.compiler.target>
</properties>

<build>
  <plugins>
    <plugin>
      <groupId>de.thetaphi</groupId>
      <artifactId>forbiddenapis</artifactId>
      <version>1.5.1</version>
      <configuration>
        <!-- disallow undocumented classes like sun.misc.Unsafe: -->
        <internalRuntimeForbidden>true</internalRuntimeForbidden>
        <!--
          if the used Java version is too new,
          don't fail, just do nothing:
        -->
        <failOnUnsupportedJava>false</failOnUnsupportedJava>
        <bundledSignatures>
          <!--
            This will automatically choose the right
            signatures based on 'maven.compiler.target':
          -->
          <bundledSignature>jdk-unsafe</bundledSignature>
          <bundledSignature>jdk-deprecated</bundledSignature>
        </bundledSignatures>
        <signaturesFiles>
          <signaturesFile>./rel/path/to/signatures.txt</signaturesFile>
        </signaturesFiles>
      </configuration>
      <executions>
        <execution>
          <goals>
            <goal>check</goal>
            <goal>testCheck</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
  <!-- more build settings here... -->
</build>
```



```
<project xmlns:ivy="antlib:org.

  <property name="src.dir" loca
  <property name="build.dir" lo
  <property name="jdk.version"

  <path id="build.classpath">
    <!--
     define your build classpat
     This classpath should be u
    -->
  </path>

  <target name="-init">
    <ivy:cachepath organisation
      inline="true" pathid="for
    <taskdef uri="antlib:de.the
  <
  
  <

    <javac classpathref="build.
      srcdir="${src.dir}" destd
      source="${jdk.version}" t
  </target>

  <target name="forbidden-check
    <fa:forbiddenapis internalR
      <bundledsignatures name="
      <bundledsignatures name="
      <signaturesFileset file="
    </fa:forbiddenapis>
  </target>

</project>
```



```
<properties>
  <!--
   It is recommended to set the compiler version globally,
   as the compiler plugin and the forbidden API checker both
   use this version
  -->
  <maven.compiler.target>1.6</maven.compiler.target>
</properties>

<build>
  <plugins>
    <plugin>
      <groupId>de.thetaphi</groupId>
      <artifactId>forbiddenapis</artifactId>
      <version>1.5.1</version>
      <configuration>
        <!-- disallow undocumented classes like sun.misc.Unsafe: -->
        <internalRuntimeForbidden>true</internalRuntimeForbidden>
        <!--

        <bundledSignatures>
          <!--
           This will automatically choose the right
           signatures based on 'maven.compiler.target':
          -->
          <bundledSignature>jdk-unsafe</bundledSignature>
          <bundledSignature>jdk-deprecated</bundledSignature>
        </bundledSignatures>
        <signaturesFiles>
          <signaturesFile>./rel/path/to/signatures.txt</signaturesFile>
        </signaturesFiles>
      </configuration>
      <executions>
        <execution>
          <goals>
            <goal>check</goal>
            <goal>testCheck</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
  <!-- more build settings here... -->
</build>
```

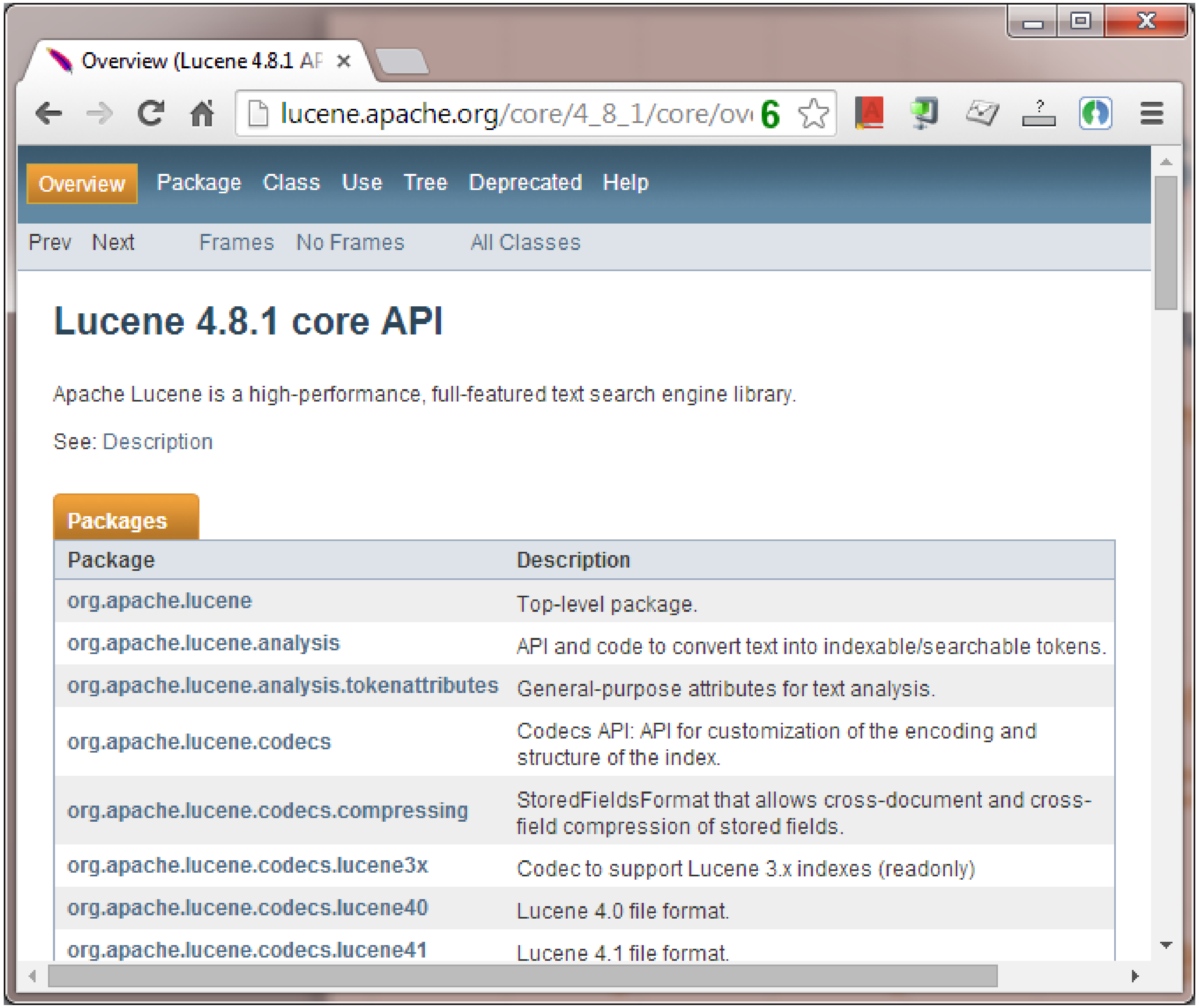
https://code.google.com/p/forbidden-apis/

IPA
Nokia
RPL-1.5
EUPL-1.1
LPPL-1.3c
AFL-3.0
Multics
BSL-1.0
EFL-2.0
LGPL-2.1
Motosoto
APL-1.0
Entessa
NPOSL-3.0
NGPL
MS-RL
SimPL-2.0
ECL-2.0
LGPL-3.0
WXwindows
Naumen
CATOSL-1.1
SPL-1.0
PHP-3.0
CUA-OPL-1.0
AGPL-3.0
Sleepycat
OCLC-2.0
GPL-3.0
Frameworx-1.0
Artistic-2.0
NCSA
BSD-3-Clause
APSL-2.0
OFL-1.1
Apache-2.0
QPL-1.0
RPSL-1.0
MS-PL
CDDL-1.0
EUDatagrid
Watcom-1.0
NTP
RSCPL
OSL-3.0
ZPL-2.0
AAL
BSD-2-Clause
EPL-1.0
MirOS
OGTSL
Zlib
NASA-1.3
CPAL-1.0
ISC
LPL-1.02
MPL-2.0
IPL-1.0
Xnet
GPL-2.0
VSL-1.0
HPND
W3C

License checker

- Checks that every dependency has a corresponding license file and SHA1 checksum available for shipping in the distribution
- As a side-effect checks that **Maven** build does not fetch additional transitive dependencies
- Used in addition to **Apache Rat**

JAVA DOCS



Lucene 4.8.1 core API

Apache Lucene is a high-performance, full-featured text search engine library.

See: Description

Packages

Package	Description
org.apache.lucene	Top-level package.
org.apache.lucene.analysis	API and code to convert text into indexable/searchable tokens.
org.apache.lucene.analysis.tokenattributes	General-purpose attributes for text analysis.
org.apache.lucene.codecs	Codecs API: API for customization of the encoding and structure of the index.
org.apache.lucene.codecs.compressing	StoredFieldsFormat that allows cross-document and cross-field compression of stored fields.
org.apache.lucene.codecs.lucene3x	Codec to support Lucene 3.x indexes (readonly)
org.apache.lucene.codecs.lucene40	Lucene 4.0 file format.
org.apache.lucene.codecs.lucene41	Lucene 4.1 file format.

Javadocs checker

- Javadocs should be up-to-date!
  - Unfortunately programmer's tend to forget about them
- 2 steps validation:

Javadocs checker

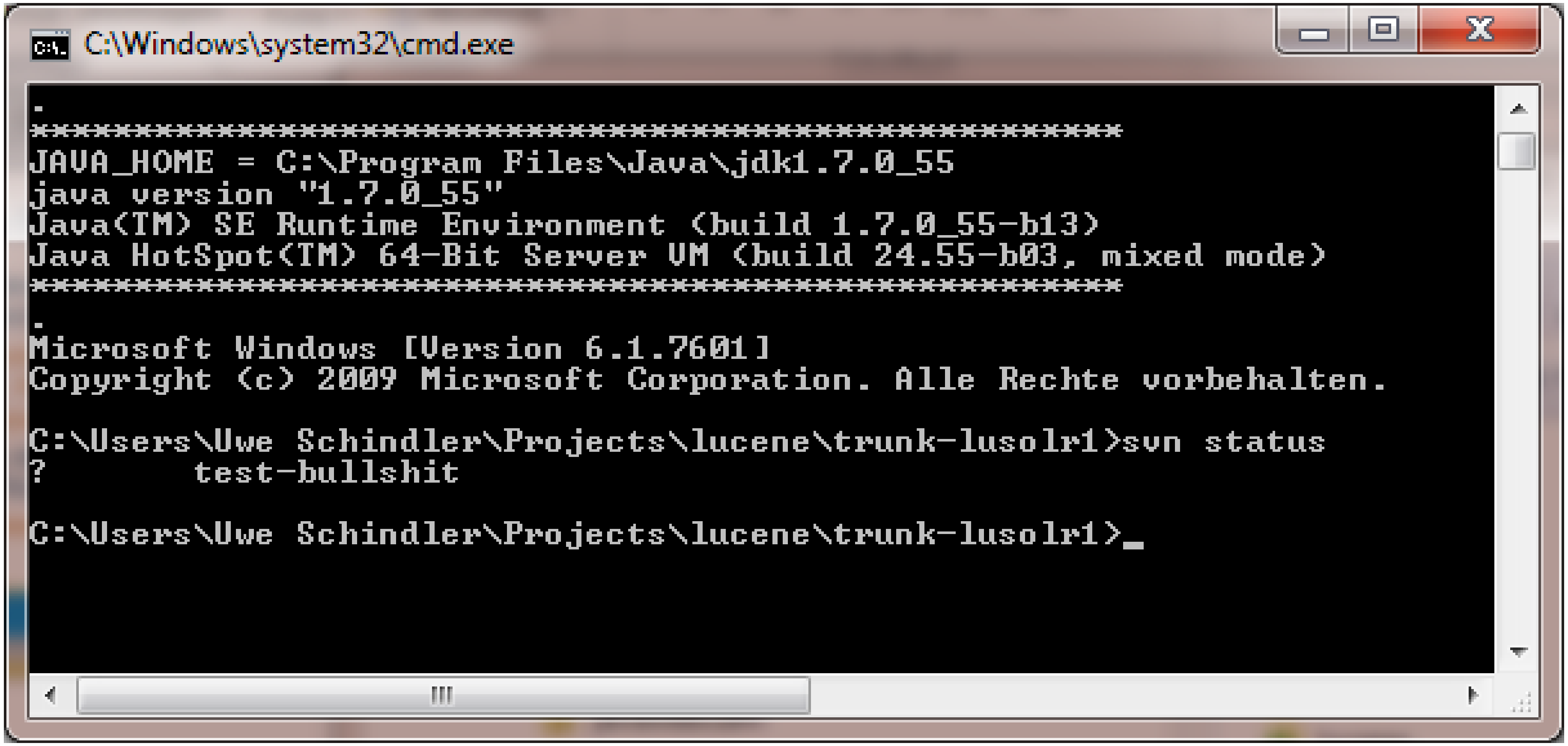
- Javadocs should be up-to-date!
  - Unfortunately programmer's tend to forget about them
- 2 steps validation:
  - Use Eclipse (`ecj`) compiler as additional validation step: This compiler allows to fail on incorrect Javadocs

Javadocs checker

- Javadocs should be up-to-date!
  - Unfortunately programmer's tend to forget about them
- 2 steps validation:
  - Use Eclipse (`ecj`) compiler as additional validation step: This compiler allows to fail on incorrect Javadocs
  - Python script that checks links, e.g. between modules

Java 8 Future: Javadocs

- Java 8 has new `-Xdoclint` feature in `javac` and `javadocs`!
  - even more strict than our checks (disallows XHTML, only HTML4)
  - lacks some checks we currently do
- Currently disabled until Javadocs are made HTML4 only!
  - If Ant detects Java 8, pass `-Xdoclint:none`



```
.
************************************************************
JAVA_HOME = C:\Program Files\Java\jdk1.7.0_55
java version "1.7.0_55"
Java(TM) SE Runtime Environment (build 1.7.0_55-b13)
Java HotSpot(TM) 64-Bit Server VM (build 24.55-b03, mixed mode)
************************************************************

.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Uwe Schindler\Projects\lucene\trunk-lusolr1>svn status
?       test-bullshit

C:\Users\Uwe Schindler\Projects\lucene\trunk-lusolr1>_
```

Test Runner

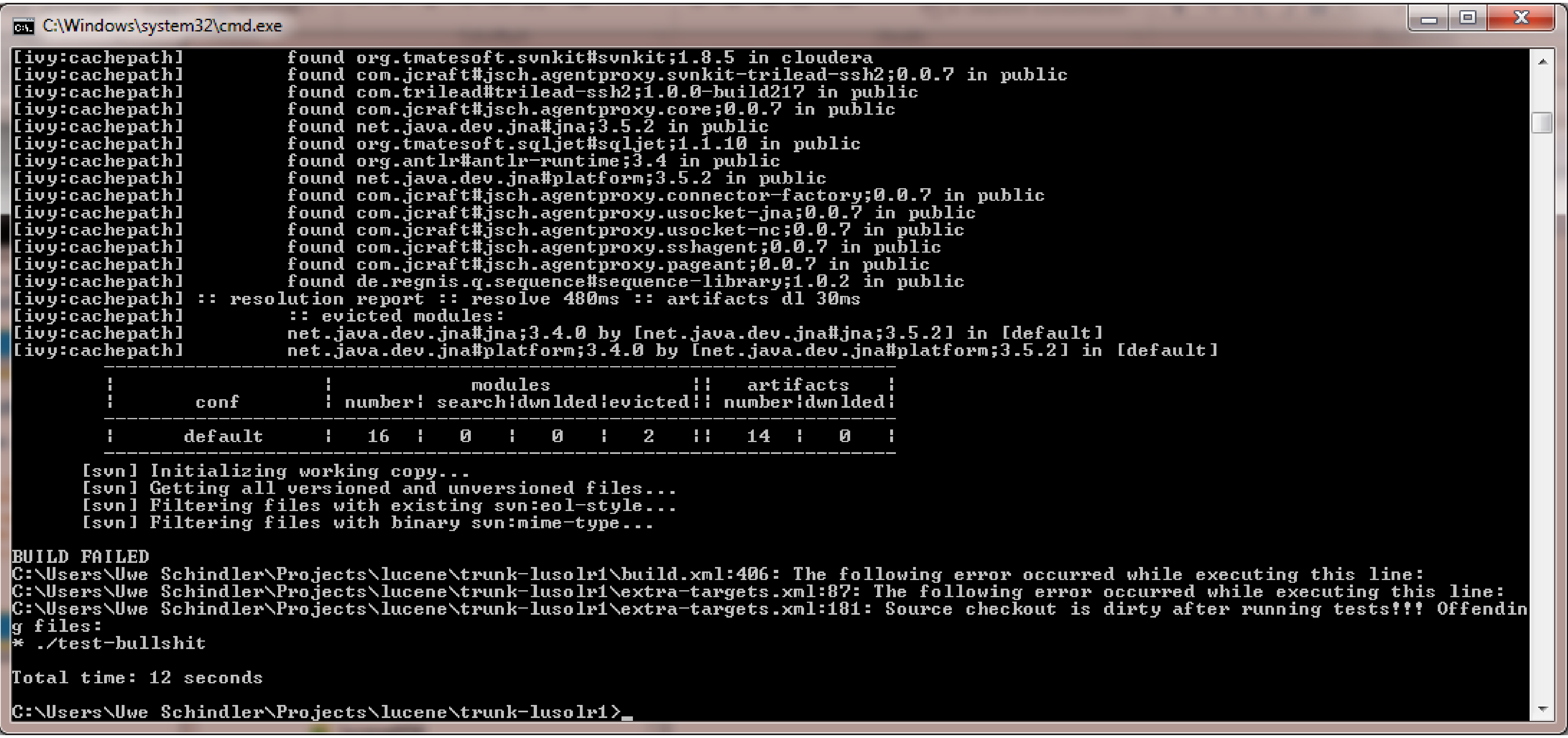
SUBVERSION ISSUES

SVN Working Copy

- Precommit and Jenkins consistency checks:
  - working copy should not be dirty after running tests *(leftover files)*
  - All files need correct MIME-Type SVN properties
  - `svn:eol-style` is checked

SVN Working Copy

- Precommit and Jenkins consistency checks:
  - working copy should not be dirty after running tests *(leftover files)*
  - All files need correct MIME-Type SVN properties
  - `svn:eol-style` is checked
- **Ant** `<groovy/>` script using **SVNKit**



C:\Windows\system32\cmd.exe

```
[ivy:cachepath]         found org.tmatesoft.svnkit#svnkit;1.8.5 in cloudera
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.svnkit-trilead-ssh2;0.0.7 in public
[ivy:cachepath]         found com.trilead#trilead-ssh2;1.0.0-build217 in public
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.core;0.0.7 in public
[ivy:cachepath]         found net.java.dev.jna#jna;3.5.2 in public
[ivy:cachepath]         found org.tmatesoft.sqljet#sqljet;1.1.10 in public
[ivy:cachepath]         found org.antlr#antlr-runtime;3.4 in public
[ivy:cachepath]         found net.java.dev.jna#platform;3.5.2 in public
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.connector-factory;0.0.7 in public
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.usocket-jna;0.0.7 in public
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.usocket-nc;0.0.7 in public
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.sshagent;0.0.7 in public
[ivy:cachepath]         found com.jcraft#jsch.agentproxy.pageant;0.0.7 in public
[ivy:cachepath]         found de.regnis.q.sequence#sequence-library;1.0.2 in public
[ivy:cachepath] :: resolution report :: resolve 480ms :: artifacts dl 30ms
[ivy:cachepath]         :: evicted modules:
[ivy:cachepath]         net.java.dev.jna#jna;3.4.0 by [net.java.dev.jna#jna;3.5.2] in [default]
[ivy:cachepath]         net.java.dev.jna#platform;3.4.0 by [net.java.dev.jna#platform;3.5.2] in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
        |      default     |   16  |   0   |   0   |   2   ||   14  |   0   |
        ---------------------------------------------------------------------
      [svn] Initializing working copy...
      [svn] Getting all versioned and unversioned files...
      [svn] Filtering files with existing svn:eol-style...
      [svn] Filtering files with binary svn:mime-type...

BUILD FAILED
C:\Users\Uwe Schindler\Projects\lucene\trunk-lusolr1\build.xml:406: The following error occurred while executing this line:
C:\Users\Uwe Schindler\Projects\lucene\trunk-lusolr1\extra-targets.xml:87: The following error occurred while executing this line:
C:\Users\Uwe Schindler\Projects\lucene\trunk-lusolr1\extra-targets.xml:181: Source checkout is dirty after running tests!!! Offendin
g files:
* ./test-bullshit

Total time: 12 seconds

C:\Users\Uwe Schindler\Projects\lucene\trunk-lusolr1>_
```

Leading the Wave
of Open Source

maven

Maven build

- Apache Lucene and Solr use primarily **Apache Ant** to build from source
- Optional, limited **Maven** build system

Maven build

- Apache Lucene and Solr use primarily **Apache Ant** to build from source
- Optional, limited **Maven** build system

- Apache Maven POMs are generated by additional Ant task
- Same applies for **Eclipse, IntelliJ, Netbeans** projects

24/7 randomized testing of many JVMs

POLICEMAN JENKINS

Randomization everywhere

- **Apache Lucene & Solr** use randomization while testing:
  - Random codec settings
  - Random Lucene directory implementation
  - Random locales, default charsets,…
  - Random indexing data

Randomization everywhere

- **Apache Lucene & Solr** use randomization while testing:
  - Random codec settings
  - Random Lucene directory implementation
  - Random locales, default charsets,…
  - Random indexing data
- **Reproducible:**
  - Every test gets an initial random seed
  - Printed on test execution & included in stack traces

Leading the Wave of Open Source

Randomize your tests and it will blow your socks off!

Dawid Weiss

(yesterday)

Missing parts

- JVM randomization
  - Oracle JDK 7, Oracle JDK 8
  - IBM J9 7
  - Preview releases

Missing parts

- JVM randomization
  - Oracle JDK 7, Oracle JDK 8
  - IBM J9 7
  - Preview releases
- JVM settings randomization
  - Garbage collector
  - Bitness: 32 / 64 bits
  - Server / Client VM
  - Compressed OOPs *(ordinary object pointer)*

Missing parts

- JVM randomization
  - Oracle JDK 7, Oracle JDK 8
  - IBM J9 7
  - Preview releases
- JVM settings randomization
  - Garbage collector
  - Bitness: 32 / 64 bits
  - Server / Client VM
  - Compressed OOPs *(ordinary object pointer)*
- Platform
  - Linux, Windows, MacOS X, FreeBSD,…

Possibilities

- Define each Jenkins job with a different JVM:
  - Duplicates
  - Hard to maintain
  - Multiplied by additional JVM settings like GC, server/client, or OOP size

Possibilities

- Define each Jenkins job with a different JVM:
  - Duplicates
  - Hard to maintain
  - Multiplied by additional JVM settings like GC, server/client, or OOP size
- Make Jenkins server set build / environment variables with a (pseudo-)randomization script:
  - `$JAVA_HOME` → passed to Apache Ant
  - `$TEST_JVM_ARGS` → passed to test runner

Plugins needed

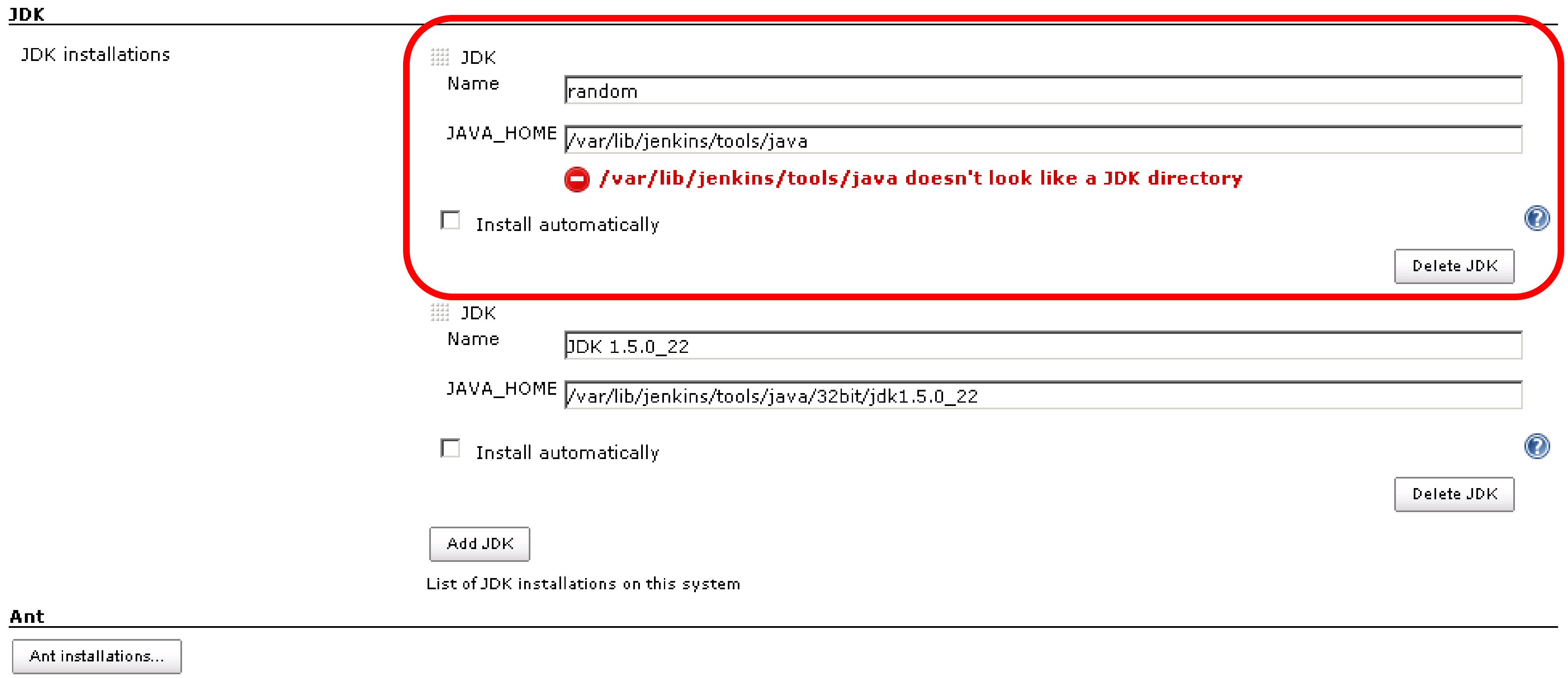
- Environment Injector Plugin
  - Executes Groovy script to do the actual work
  - Sets some build environment variables: `$JAVA_HOME, $TEST_JVM_ARGS, $JAVA_DESC`

Plugins needed

- Environment Injector Plugin
  - Executes Groovy script to do the actual work
  - Sets some build environment variables: `$JAVA_HOME, $TEST_JVM_ARGS, $JAVA_DESC`
- Jenkins Description Setter Plugin / Jenkins Email Extension Plugin
  - Add JVM details / settings to build description and e-mails

Global Jenkins settings

- Extra JDK config in Jenkins (called “random”):
  - pointing to *dummy* directory *(we can use the base directory containing all our JDKs)*
  - Assigned to every job that needs a randomly choosen virtual machine



JDK

JDK installations

JDK
Name: random
JAVA_HOME: /var/lib/jenkins/tools/java
/var/lib/jenkins/tools/java doesn't look like a JDK directory
☐ Install automatically
Delete JDK

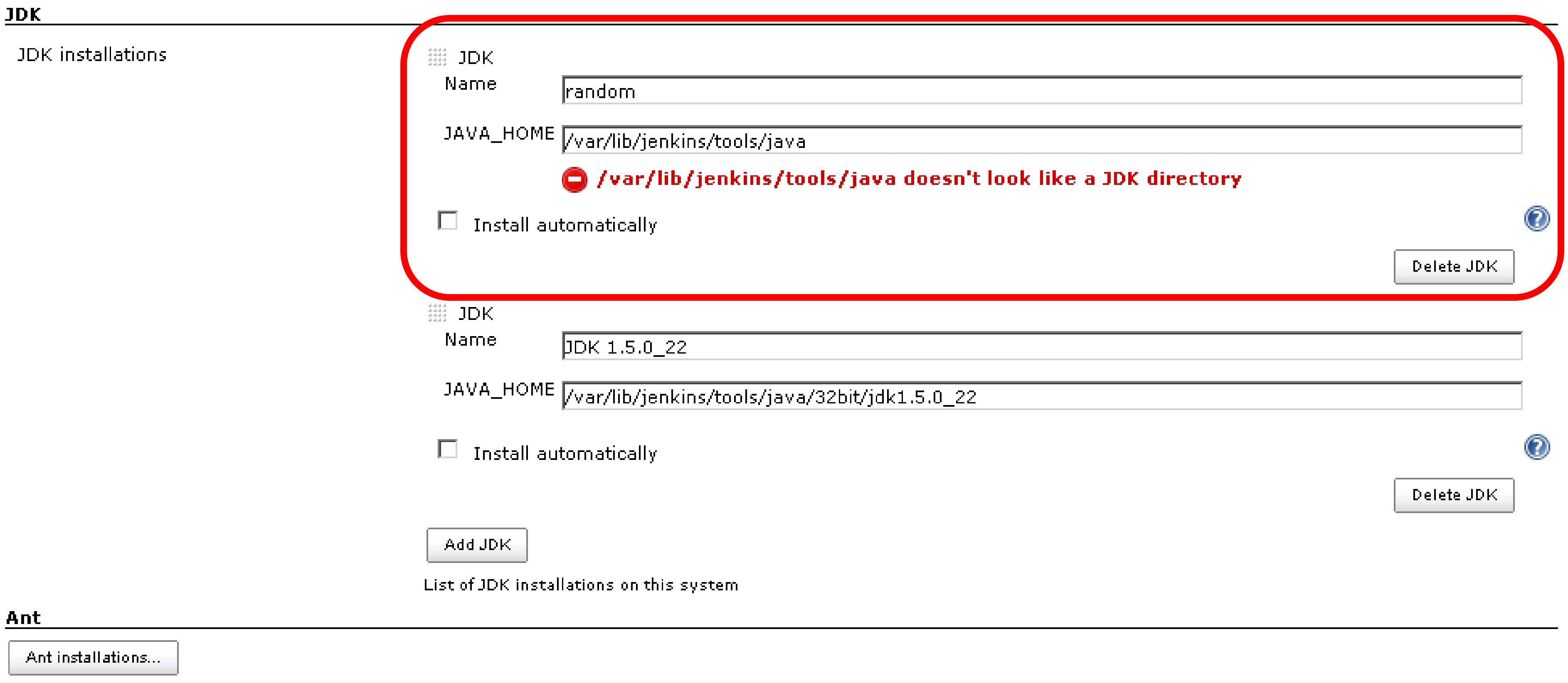
JDK
Name: JDK 1.5.0_22
JAVA_HOME: /var/lib/jenkins/tools/java/32bit/jdk1.5.0_22
☐ Install automatically
Delete JDK

Add JDK

List of JDK installations on this system

Ant

Ant installations...



JDK

JDK installations

JDK

Name: random

JAVA_HOME: /var/lib/jenkins/tools/java

/var/lib/jenkins/tools/java doesn't look like a JDK directory

Install automatically

Delete JDK

JDK

Name: JDK 1.5.0_22

JAVA_HOME: /var/lib/jenkins/tools/java/32bit/jdk1.5.0_22

Install automatically

Delete JDK

Add JDK

List of JDK installations on this system

Ant

Ant installations...

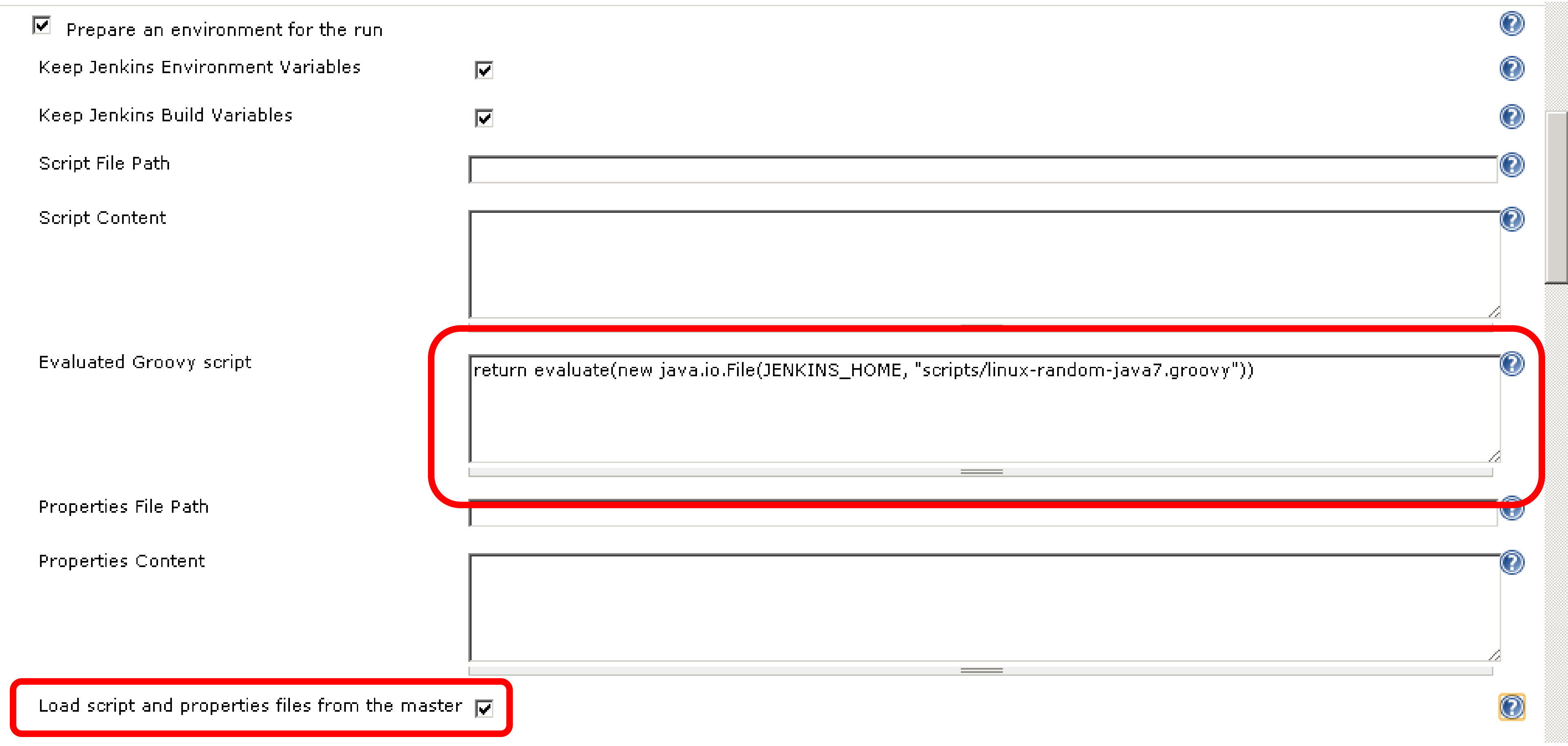
The warning displayed by Jenkins doesn't matter!

Job Config

- Standard **free style build** with plugins activated
  - Calls Groovy script file with main logic (sets `$JAVA_HOME` randomly,…)
  - List of JVM options as a „config file“
  - Job‘s JDK version set to „random“
  - Apache Ant configuration automatically gets `$JAVA_HOME` and test runner gets extra options via build properties

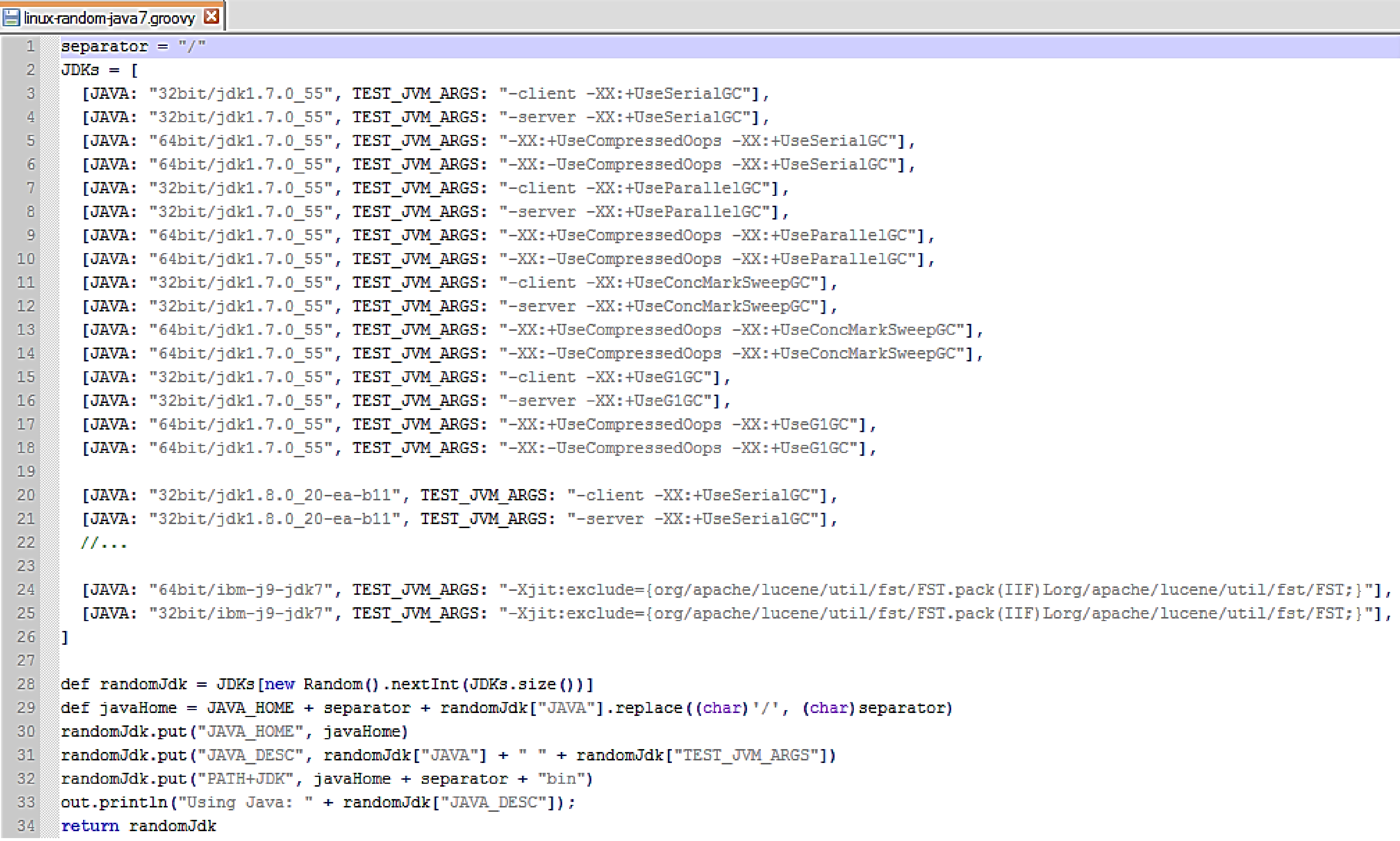
Job Config

- Standard **free style build** with plugins activated
  - Calls Groovy script file with main logic (sets `$JAVA_HOME` randomly,…)
  - List of JVM options as a „config file“
  - Job‘s JDK version set to „random“
  - Apache Ant configuration automatically gets `$JAVA_HOME` and test runner gets extra options via build properties
- Should work with Maven builds, too!

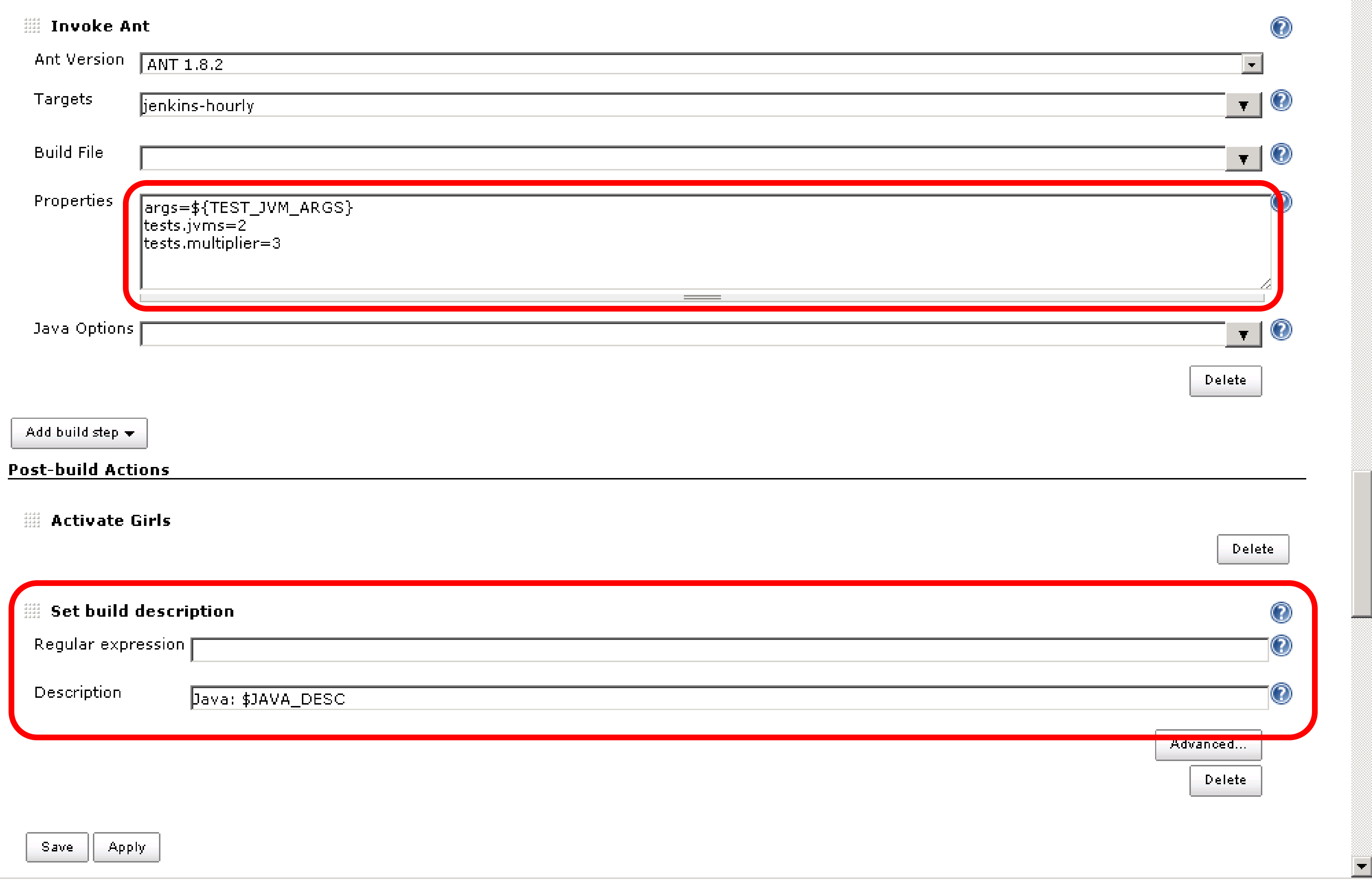


[✓] Prepare an environment for the run

Keep Jenkins Environment Variables	✓
Keep Jenkins Build Variables	✓
Script File Path	
Script Content	
Evaluated Groovy script	`return evaluate(new java.io.File(JENKINS_HOME, "scripts/linux-random-java7.groovy"))`
Properties File Path	
Properties Content	
Load script and properties files from the master	✓



```
separator = "/"
JDKs = [
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-client -XX:+UseSerialGC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-server -XX:+UseSerialGC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:+UseCompressedOops -XX:+UseSerialGC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:-UseCompressedOops -XX:+UseSerialGC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-client -XX:+UseParallelGC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-server -XX:+UseParallelGC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:+UseCompressedOops -XX:+UseParallelGC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:-UseCompressedOops -XX:+UseParallelGC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-client -XX:+UseConcMarkSweepGC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-server -XX:+UseConcMarkSweepGC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:+UseCompressedOops -XX:+UseConcMarkSweepGC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:-UseCompressedOops -XX:+UseConcMarkSweepGC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-client -XX:+UseG1GC"],
  [JAVA: "32bit/jdk1.7.0_55", TEST_JVM_ARGS: "-server -XX:+UseG1GC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:+UseCompressedOops -XX:+UseG1GC"],
  [JAVA: "64bit/jdk1.7.0_55", TEST_JVM_ARGS: "-XX:-UseCompressedOops -XX:+UseG1GC"],

  [JAVA: "32bit/jdk1.8.0_20-ea-b11", TEST_JVM_ARGS: "-client -XX:+UseSerialGC"],
  [JAVA: "32bit/jdk1.8.0_20-ea-b11", TEST_JVM_ARGS: "-server -XX:+UseSerialGC"],
  //...

  [JAVA: "64bit/ibm-j9-jdk7", TEST_JVM_ARGS: "-Xjit:exclude={org/apache/lucene/util/fst/FST.pack(IIF)Lorg/apache/lucene/util/fst/FST;}"],
  [JAVA: "32bit/ibm-j9-jdk7", TEST_JVM_ARGS: "-Xjit:exclude={org/apache/lucene/util/fst/FST.pack(IIF)Lorg/apache/lucene/util/fst/FST;}"],
]

def randomJdk = JDKs[new Random().nextInt(JDKs.size())]
def javaHome = JAVA_HOME + separator + randomJdk["JAVA"].replace((char)'/', (char)separator)
randomJdk.put("JAVA_HOME", javaHome)
randomJdk.put("JAVA_DESC", randomJdk["JAVA"] + " " + randomJdk["TEST_JVM_ARGS"])
randomJdk.put("PATH+JDK", javaHome + separator + "bin")
out.println("Using Java: " + randomJdk["JAVA_DESC"]);
return randomJdk
```



Invoke Ant

Ant Version: ANT 1.8.2

Targets: jenkins-hourly

Build File:

Properties:

```
args=${TEST_JVM_ARGS}
tests.jvms=2
tests.multiplier=3
```

Java Options:

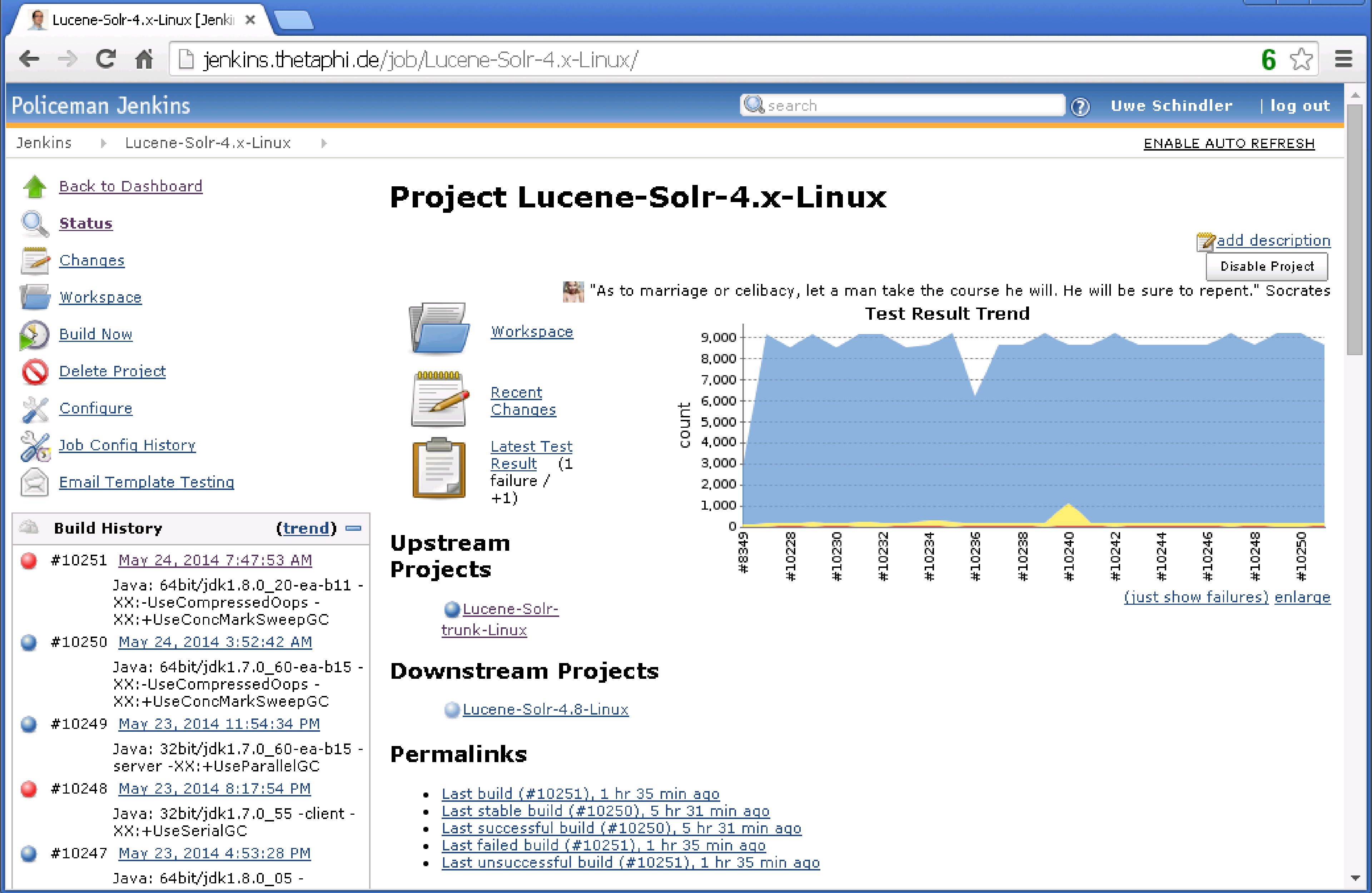
Post-build Actions

Activate Girls

Set build description

Regular expression:

Description: Java: $JAVA_DESC



Lucene-Solr-4.x-Linux [Jenki

jenkins.thetaphi.de/job/Lucene-Solr-4.x-Linux/

Policeman Jenkins | search | Uwe Schindler | log out

Jenkins › Lucene-Solr-4.x-Linux › ENABLE AUTO REFRESH

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now
- Delete Project
- Configure
- Job Config History
- Email Template Testing

Build History (trend)

● #10251 May 24, 2014 7:47:53 AM
Java: 64bit/jdk1.8.0_20-ea-b11 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC

● #10250 May 24, 2014 3:52:42 AM
Java: 64bit/jdk1.7.0_60-ea-b15 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC

● #10249 May 23, 2014 11:54:34 PM
Java: 32bit/jdk1.7.0_60-ea-b15 -server -XX:+UseParallelGC

● #10248 May 23, 2014 8:17:54 PM
Java: 32bit/jdk1.7.0_55 -client -XX:+UseSerialGC

● #10247 May 23, 2014 4:53:28 PM
Java: 64bit/jdk1.8.0_05 -

Project Lucene-Solr-4.x-Linux

add description
Disable Project

"As to marriage or celibacy, let a man take the course he will. He will be sure to repent." Socrates

- Workspace
- Recent Changes
- Latest Test Result (1 failure / +1)

Test Result Trend

(just show failures) enlarge

Upstream Projects

Lucene-Solr-trunk-Linux

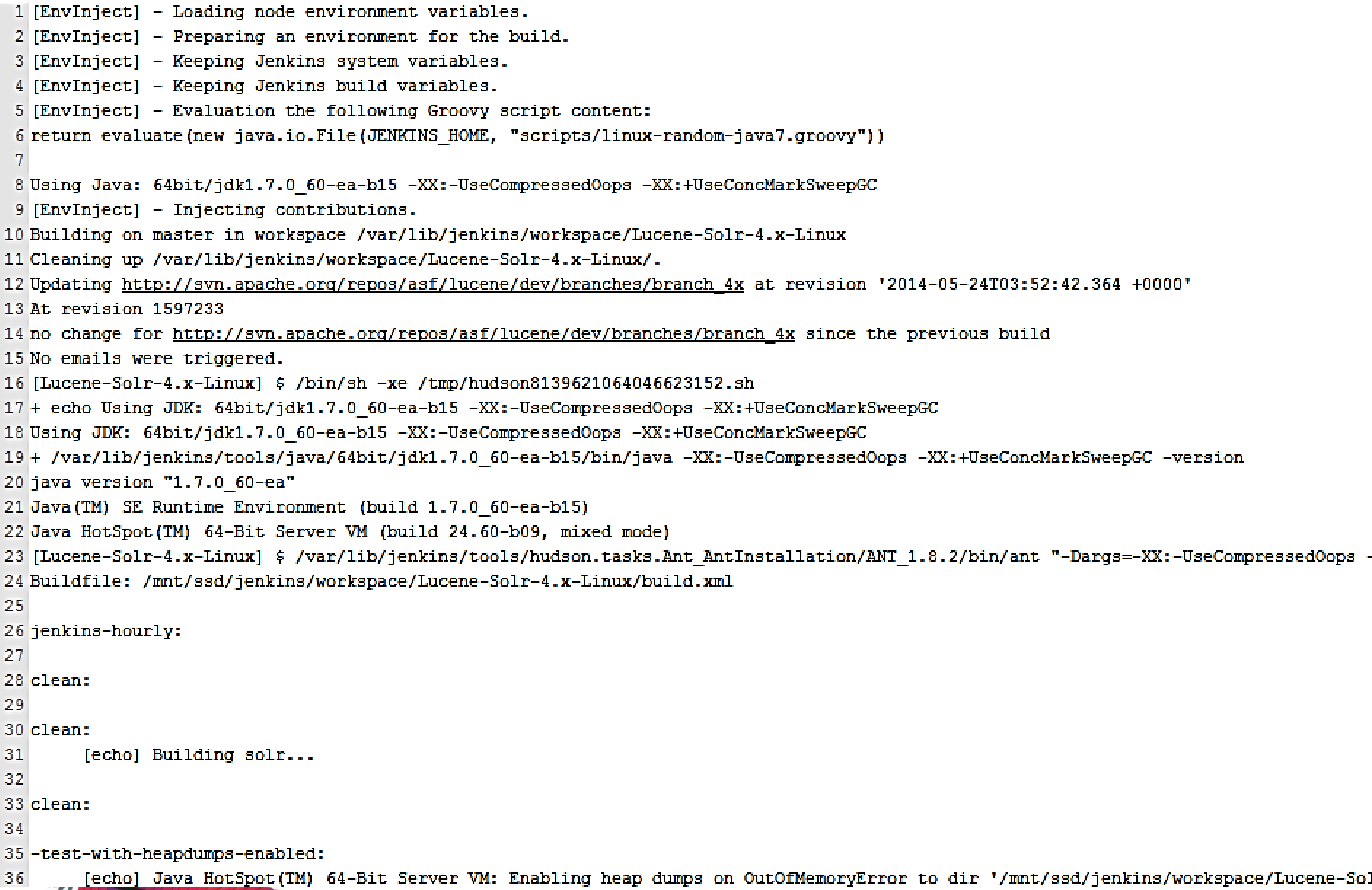
Downstream Projects

Lucene-Solr-4.8-Linux

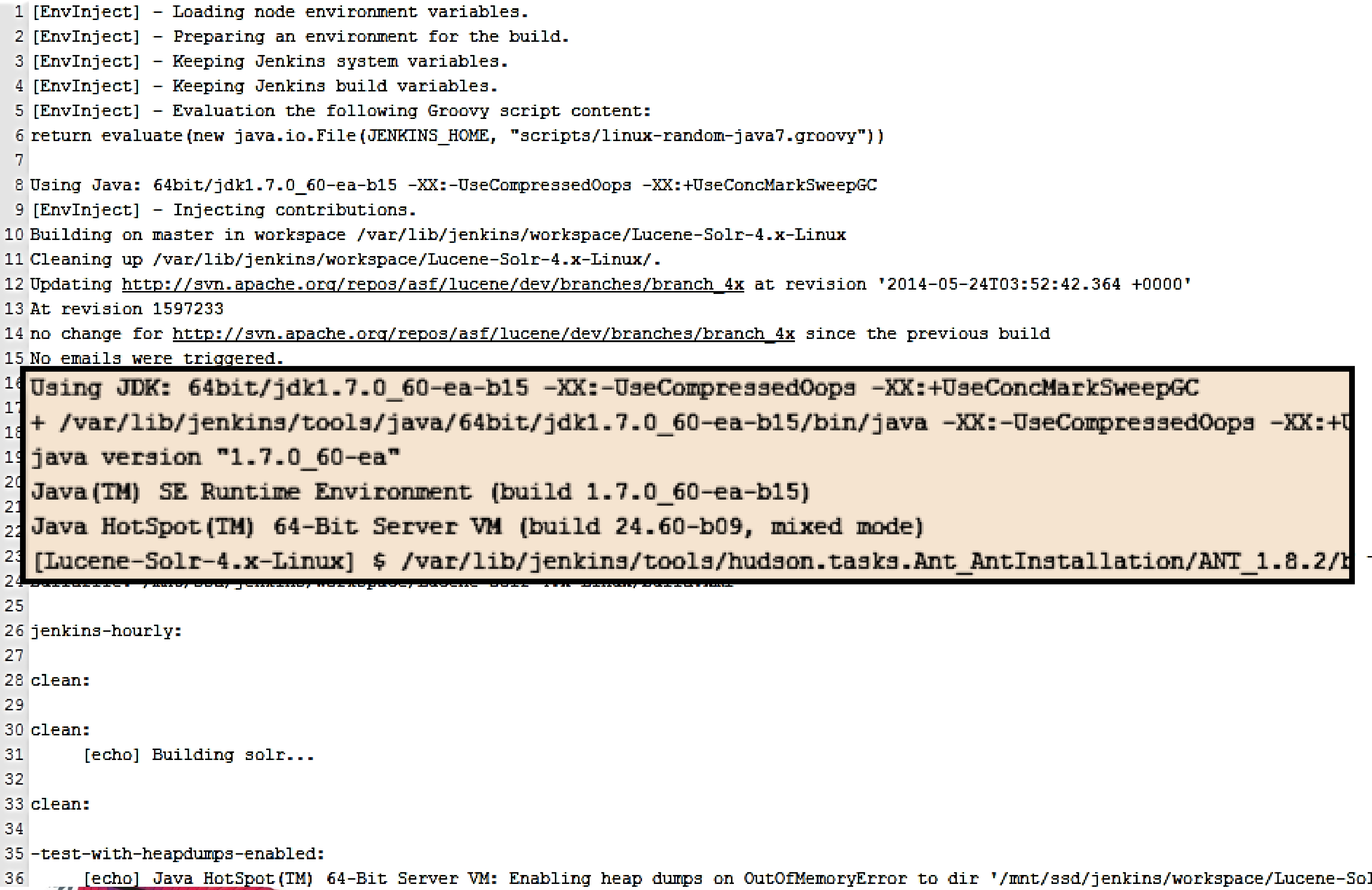
Permalinks

- Last build (#10251), 1 hr 35 min ago
- Last stable build (#10250), 5 hr 31 min ago
- Last successful build (#10250), 5 hr 31 min ago
- Last failed build (#10251), 1 hr 35 min ago
- Last unsuccessful build (#10251), 1 hr 35 min ago

Leading the Wave of Open Source



```
[EnvInject] - Loading node environment variables.
[EnvInject] - Preparing an environment for the build.
[EnvInject] - Keeping Jenkins system variables.
[EnvInject] - Keeping Jenkins build variables.
[EnvInject] - Evaluation the following Groovy script content:
return evaluate(new java.io.File(JENKINS_HOME, "scripts/linux-random-java7.groovy"))

Using Java: 64bit/jdk1.7.0_60-ea-b15 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC
[EnvInject] - Injecting contributions.
Building on master in workspace /var/lib/jenkins/workspace/Lucene-Solr-4.x-Linux
Cleaning up /var/lib/jenkins/workspace/Lucene-Solr-4.x-Linux/.
Updating http://svn.apache.org/repos/asf/lucene/dev/branches/branch_4x at revision '2014-05-24T03:52:42.364 +0000'
At revision 1597233
no change for http://svn.apache.org/repos/asf/lucene/dev/branches/branch_4x since the previous build
No emails were triggered.
[Lucene-Solr-4.x-Linux] $ /bin/sh -xe /tmp/hudson8139621064046623152.sh
+ echo Using JDK: 64bit/jdk1.7.0_60-ea-b15 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC
Using JDK: 64bit/jdk1.7.0_60-ea-b15 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC
+ /var/lib/jenkins/tools/java/64bit/jdk1.7.0_60-ea-b15/bin/java -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC -version
java version "1.7.0_60-ea"
Java(TM) SE Runtime Environment (build 1.7.0_60-ea-b15)
Java HotSpot(TM) 64-Bit Server VM (build 24.60-b09, mixed mode)
[Lucene-Solr-4.x-Linux] $ /var/lib/jenkins/tools/hudson.tasks.Ant_AntInstallation/ANT_1.8.2/bin/ant "-Dargs=-XX:-UseCompressedOops -
Buildfile: /mnt/ssd/jenkins/workspace/Lucene-Solr-4.x-Linux/build.xml

jenkins-hourly:

clean:

clean:
     [echo] Building solr...

clean:

-test-with-heapdumps-enabled:
     [echo] Java HotSpot(TM) 64-Bit Server VM: Enabling heap dumps on OutOfMemoryError to dir '/mnt/ssd/jenkins/workspace/Lucene-Sol
```



```
[EnvInject] - Loading node environment variables.
[EnvInject] - Preparing an environment for the build.
[EnvInject] - Keeping Jenkins system variables.
[EnvInject] - Keeping Jenkins build variables.
[EnvInject] - Evaluation the following Groovy script content:
return evaluate(new java.io.File(JENKINS_HOME, "scripts/linux-random-java7.groovy"))

Using Java: 64bit/jdk1.7.0_60-ea-b15 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC
[EnvInject] - Injecting contributions.
Building on master in workspace /var/lib/jenkins/workspace/Lucene-Solr-4.x-Linux
Cleaning up /var/lib/jenkins/workspace/Lucene-Solr-4.x-Linux/.
Updating http://svn.apache.org/repos/asf/lucene/dev/branches/branch_4x at revision '2014-05-24T03:52:42.364 +0000'
At revision 1597233
no change for http://svn.apache.org/repos/asf/lucene/dev/branches/branch_4x since the previous build
No emails were triggered.
Using JDK: 64bit/jdk1.7.0_60-ea-b15 -XX:-UseCompressedOops -XX:+UseConcMarkSweepGC
+ /var/lib/jenkins/tools/java/64bit/jdk1.7.0_60-ea-b15/bin/java -XX:-UseCompressedOops -XX:+U
java version "1.7.0_60-ea"
Java(TM) SE Runtime Environment (build 1.7.0_60-ea-b15)
Java HotSpot(TM) 64-Bit Server VM (build 24.60-b09, mixed mode)
[Lucene-Solr-4.x-Linux] $ /var/lib/jenkins/tools/hudson.tasks.Ant_AntInstallation/ANT_1.8.2/b

jenkins-hourly:

clean:

clean:
     [echo] Building solr...

clean:

-test-with-heapdumps-enabled:
     [echo] Java HotSpot(TM) 64-Bit Server VM: Enabling heap dumps on OutOfMemoryError to dir '/mnt/ssd/jenkins/workspace/Lucene-Sol
```

python
SMOKE DETECTOR TESTER
A CONVENIENT TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE. CONTENTS UNDER PRESSURE
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

AUTOMATED RELEASE TESTING

Release Workflow

- Release Manager (RM) creates artifacts
- RM does initial testing
- Project Management Committee (PMC) votes for artifacts *(72hrs)*
- RM publishes artifacts and javadocs

Release Building

- All Apache Ant checks *(like previously presented)*
- Python script creates release and uploads to staging area
- Runs "smoke tester"

Smoke Tester

python™
SMOKE
DETECTOR
TESTER
A CONVENIENT
TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE. CONTENTS UNDER PRESSURE.
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

Smoke Tester

- Python™ powered

python™
SMOKE DETECTOR TESTER
A CONVENIENT TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE. CONTENTS UNDER PRESSURE.
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

Smoke Tester

- Python™ powered
- Convenient use for release manager and PMC

python™
SMOKE DETECTOR TESTER
A CONVENIENT TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE. CONTENTS UNDER PRESSURE
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

Smoke Tester

- Python™ powered
- Convenient use for release manager and PMC
- Includes functional testing ☺

python™
SMOKE DETECTOR TESTER
A CONVENIENT TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE CONTENTS UNDER PRESSURE
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

Smoke Tester

- Python™ powered
- Convenient use for release manager and PMC
- Includes functional testing ☺
- Takes approx. one hour

python™
SMOKE DETECTOR TESTER
A CONVENIENT TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE. CONTENTS UNDER PRESSURE
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

Smoke Tester

- Python™ powered
- Convenient use for release manager and PMC
- Includes functional testing ☺
- Takes approx. one hour
- **Uses all your CPU and burns package contents!**

python™
SMOKE DETECTOR TESTER
A CONVENIENT TRUE FUNCTIONAL TEST
HOMEOWNER'S SIZE MODEL 1H
DANGER: EXTREMELY FLAMMABLE CONTENTS UNDER PRESSURE
CONTAINS ISO-BUTANE AND PROPANE
SEE BACK PANEL FOR ADDITIONAL WARNING
NET WT. 1 OZ. (28g)

Continuous Nightly

- **Smoke testing** runs nightly as **Jenkins Job**
- Preview releases downloadable:
  - https://builds.apache.org/job/Lucene-Artifacts-4.x/
  - https://builds.apache.org/job/Solr-Artifacts-4.x/

Thank You!