

# Java 7 – State of the Enterprise

Challenges, Adoption Path and Reasons to move forward ...

Automotive Communications Financial Services Government Insurance Life Science  
& Healthcare Travel & Logistics Utilities Automotive Communications Financial  
Services Government Insurance Life Science & Healthcare Travel & Logistics  
Utilities Automotive Communications Financial Services Government  
Insurance Life Science & Healthcare Travel & Logistics Utilities Automotive  
Communications Financial Services Government Insurance Life Science  
& Healthcare Travel & Logistics Utilities Automotive Communications  
Financial Services Government Insurance Life Science & Healthcare Travel  
& Logistics Utilities Automotive Communications Financial Services  
Government Insurance Life Science & Healthcare Travel & Logistics  
Utilities Automotive Communications Financial Services Government  
Insurance Life Science & Healthcare Travel & Logistics Utilities Automotive  
Communications Financial Services Government Insurance Life Science &  
Healthcare Travel & Logistics Utilities Automotive Communications Financial  
Services Government Insurance Life Science & Healthcare Travel & Logistics  
Utilities Automotive Communications Financial Services Government Insurance  
Life Science & Healthcare Travel & Logistics Utilities Automotive Communications



**Jfokus  
2012**

Stockholm Waterfront  
Congress Center  
February 13–15 2012

.consulting .solutions .partnership



The thoughts expressed here are the personal opinions of the author and no official statement of the msg systems ag.

- 1. Introduction**
- 2. Reasons: Why Migrate?**
- 3. Migration Procedures**
- 4. Challenges**
- 5. Adoption Paths**
- 6. Restrictions**
- 7. Real World Answers**





ORACLE  
ACE Director

Scale Exadata  
The First OLTP Database  
With Sun FlashFire Technology



<http://blog.eisele.net>  
<http://twitter.com/myfear>  
markus@eisele.net



2011  
**DOAG**  
Konferenz + Ausstellung

**JavaTECH**  
JOURNAL

*Java*magazin

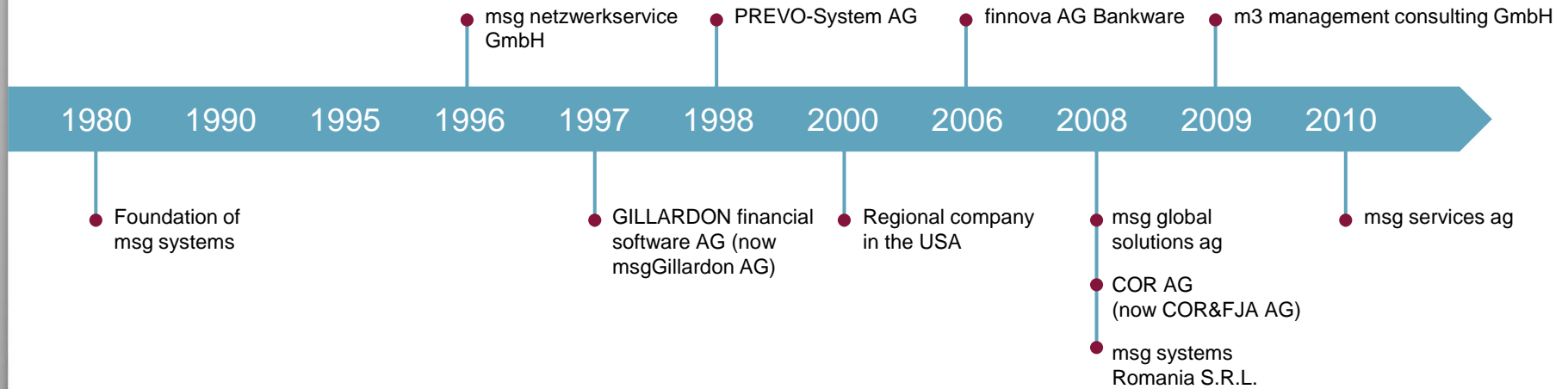
**ODTUG**  
**Kscope11**  
LONG BEACH, CALIFORNIA • JUNE 26-30

**eclipseCON**  
EUROPE 2011

**Java**aktuell  
Das JUG Magazin

**iX**  
MAGAZIN FÜR PROFESSIONELLE  
INFORMATIONSTECHNIK

**Java** mit Integrations-  
SPEKTRUM  
Magazin für professionelle, integrierte und komplexe Java-Systeme

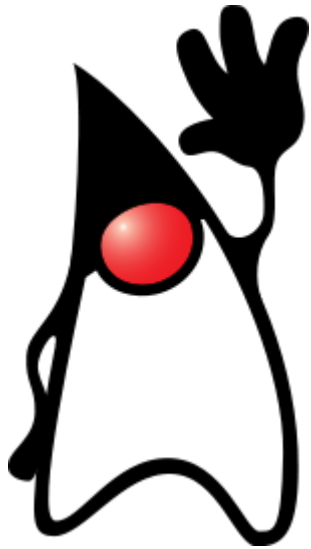


**Industries:**

- Insurance
- Financial Services
- Automotive
- Communications
- Travel & Logistics
- Utilities
- Life Science & Healthcare
- Government

**Individual Solutions:**

- Allianz | AUDI | BG-PHOENICS | BMW Financial Services | BMW Group | Daimler | DER Deutsches Reisebüro | Deutsche Bank | Deutsche Post | Sächsischer Landtag | Versicherungskammer Bayern | VR Leasing



- Java Language
  - Project Coin (JSR-334)
- Class Libraries
  - NIO2 (JSR-203)
  - Fork-Join framework, ParallelArray (JSR-166y)
- Java Virtual Machine
  - The DaVinci Machine project (JSR-292)
  - InvokeDynamic bytecode
- Miscellaneous things

### Features of Java SE 7

- Improved performance, stability and security.
- Enhancements in the Java Plug-in for Rich Internet Applications development and deployment.
- Java Programming language enhancements that enable developers with ease of writing and optimizing the Java code.
- Enhancements in the Java Virtual machine to support Non-Java languages.



<http://www.slideshare.net/myfear/introducing-java-7>

## Why Upgrade? – Old Stuff is getting EOL'ed!

### Java SE Support Roadmap

Release Family	GA Date	EOL Notification	EOL	Support Period
1.4	Feb 2002	Dec 2006	Oct 2008	6 ½ years
5.0	May 2004	Apr 2008	Oct 2009	5 ½ years
6	Dec 2006	Feb 2011	<b>Jul 2012</b>	5 1/2 years



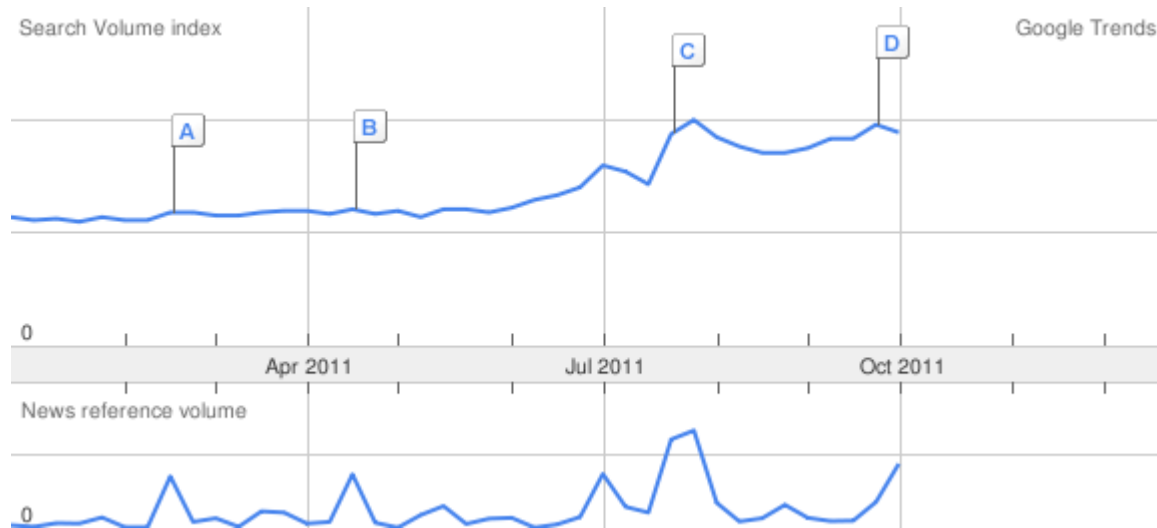
### Example Java for Business Support Roadmap

Release Family	GA Date	EOL Notification	Premier Support until	Extended Support until	Sustaining Period
1.4	Feb 2002	Apr 2008	Feb 2010	Feb 2013	Indefinite
5.0	May 2004	Jun 2009	May 2011	May 2014	Indefinite
6	Dec 2006	Jan 2011	Dec 2013	<b>Dec 2016</b>	Indefinite

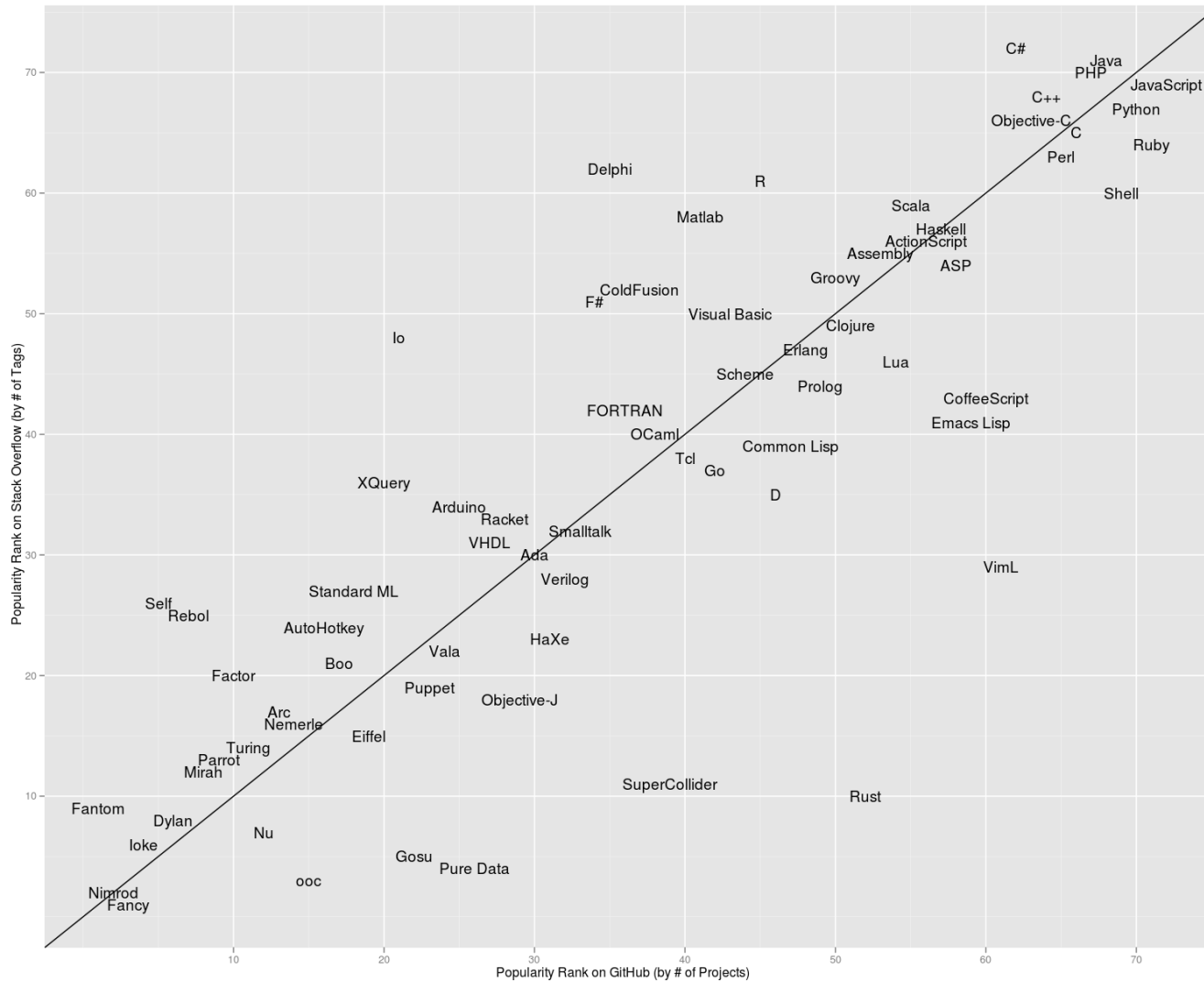
<http://www.oracle.com/technetwork/java/javase/eol-135779.html>



## Why Upgrade? – It's a hype!

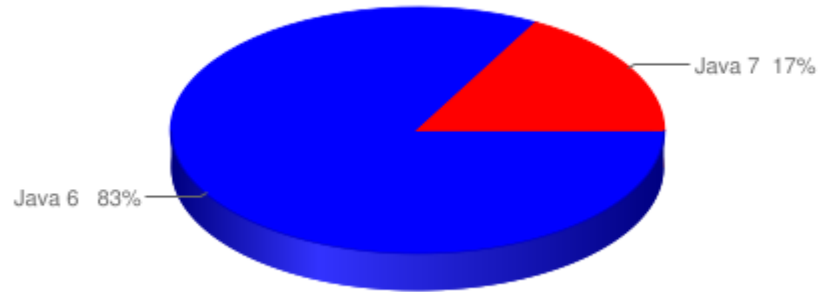


- A** Java 7 Support Slated for JetBrains IDE  
PCWorld - Feb 23 2011
- B** Updated Oracle NetBeans IDE offers Java 7  
Computerworld - Apr 20 2011
- C** **Oracle releases Java 7 Standard Edition**  
**Inquirer - Jul 29 2011**
- D** Eclipse Announces Full Support for Java 7  
San Francisco Chronicle - Sep 29 2011



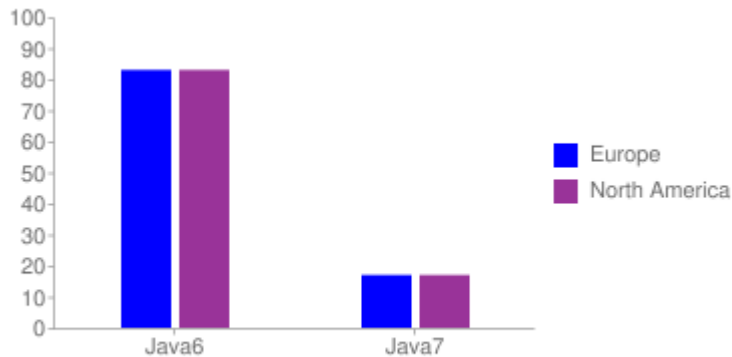
<http://redmonk.com/sograzy/2012/02/08/language-rankings-2-2012/>

Java version distribution, January 2012



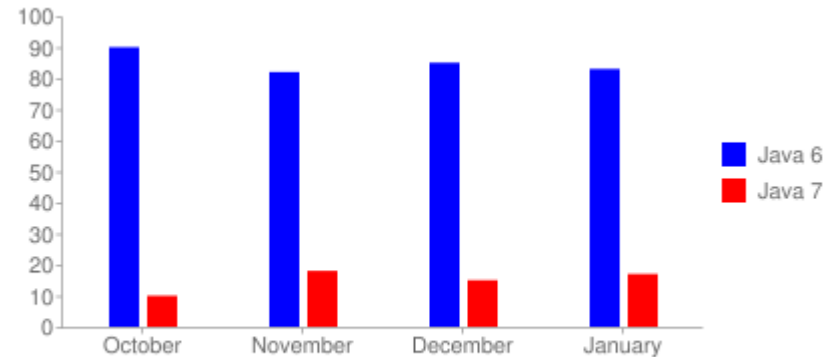
Source: Hosting statistics from Jelastic.com.

Java version market share by region, January 2012



Source: Hosting statistics from Jelastic.com.

Java version market share by time period, Oct 2011 - Jan 2012



Source: Hosting statistics from Jelastic.com.

<http://blog.jelastic.com/2012/02/06/java-version-market-share-january-2012/>

## But: Enterprises think different:

- Maximize Benefit to the Enterprise
- Business Continuity
- Common Use Applications
- Requirements-Based Change
- Responsive Change Management
- Control Technical Diversity
- Interoperability

# Have You Adopted Java 7 Yet?

FEBRUARY 09, 2012 AT 1:30 AM



We have adopted Java 7



We're still on Java 6, with no plans to change



We'll move to Java 7 later this year

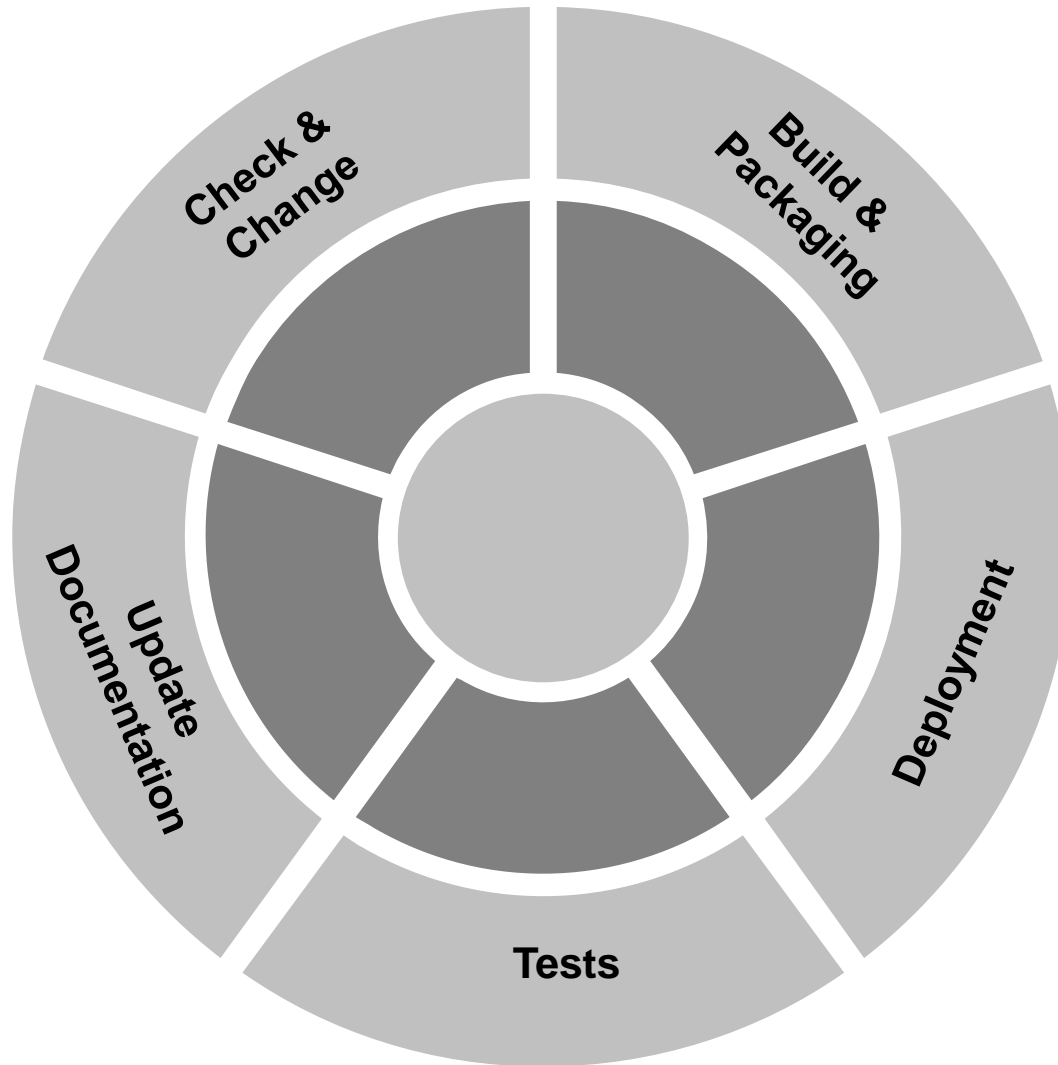


We haven't even moved to Java 6!



Total votes: 888

<http://java.dzone.com/polls/have-you-adopted-java-7-yet>





- **Pre-Flight Check**
  - **Development Configuration**
    - Does your IDE support your version?
    - What about your build tool?
    - Your quality checkers?
    - Your ALM solutions?
  - **Project Dependencies (OSS & Commercial)**
    - Do your libraries run on/support new version?
  - **Your Company Specific Frameworks**
    - They are out there. Check them!
  - **Deprecated Features**
    - Do you use anything that has been deprecated?
    - Your chance to refactor a bit!
  - **Incompabilities**
    - <http://www.oracle.com/technetwork/java/javase/compatibility-417013.html#incompatibilities>
- **Change if necessary**
  - Control what you do!
  - Make a list and track it!
  - Other Projects may need it to estimate efforts!



<http://download.oracle.com/javase/7/docs/webnotes/adoptionGuide/index.html>



- Check your build
  - Change JRE/JDK libraries
  - Change source version level for maven
  - Check special ANT tasks (ftp..)
- Change if necessary
  - All of the above
  - Also a very good time to :
    - change build tools
    - Get rid of old and unused stuff
    - Clean old mess
- Test-drive your “updated” build process carefully.

Java SE 7 is binary-compatible with Java SE 6  
The class file version for Java SE 7 is 51.





- Using Java 7 on the Client is no save solution at the moment!
- <http://www.java.com/> still distributes Java 1.6\_30 to the public.
- If you chose to change your deployments there are some chances to improve it:
  - Change your Applets to use the Deployment Toolkit (codebase)
  - Utilize the new “resource per OS” feature
  - Think about partly signing your JNLP files
  - Some small enhancements to Applet loading (single network requests)

JAVA + YOU,  
DOWNLOAD  
TODAY!

Free Java Download

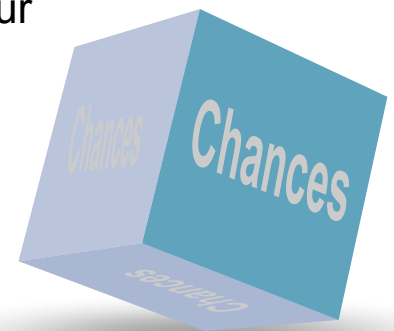




- Excuse: If your tests randomly fail with Java 7
- JUnit uses `Class.getMethods()` to find all tests in a class
  - This list is not explicitly sorted in any order!
  - Until Java 6 the methods were returned in same order as declared in source file! This changed with Java 7.
- Repair your tests to not rely on execution order of `@Test` methods!
- Test. Test. Test.



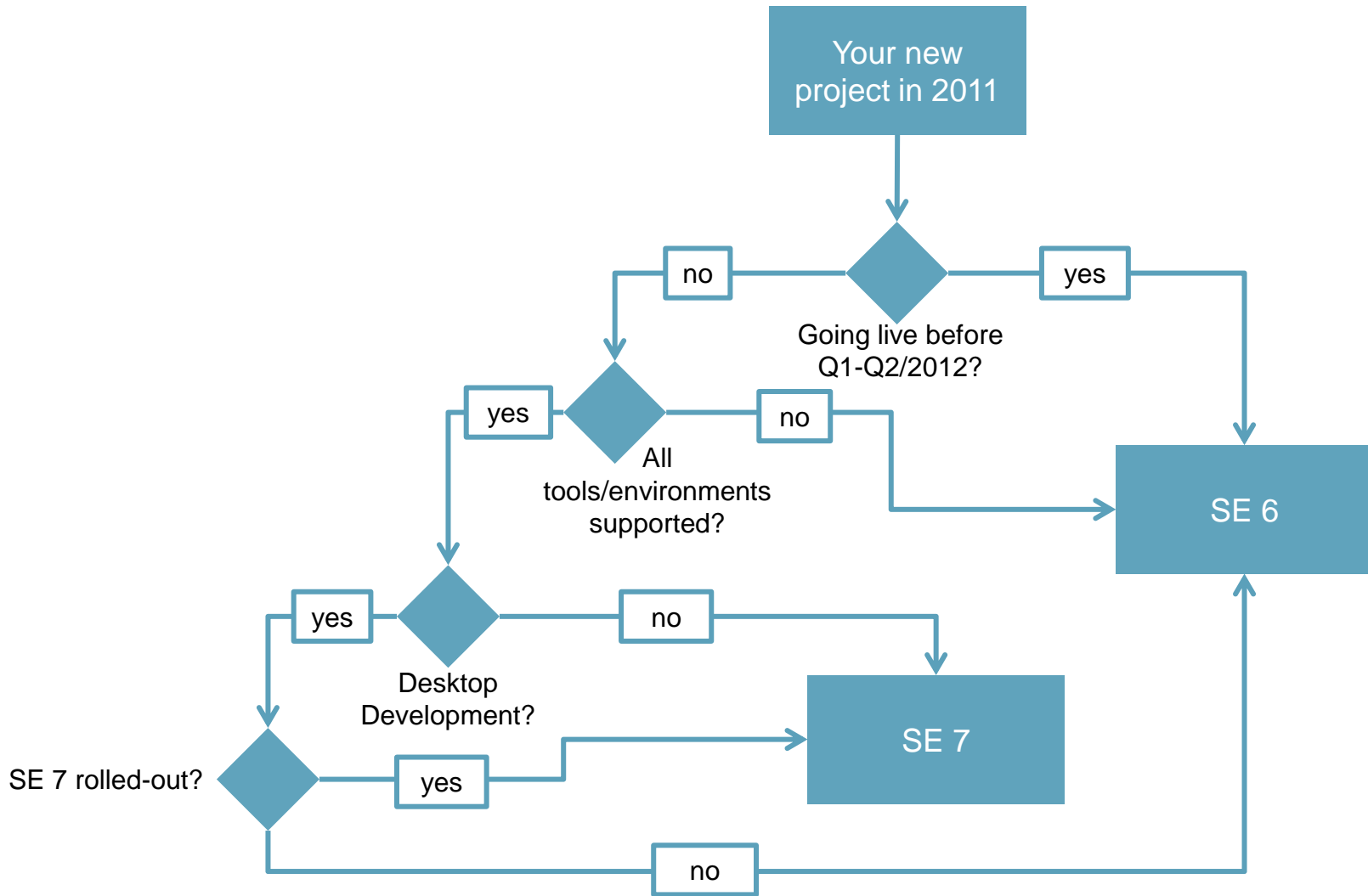
- Why are developers adverse to documenting software systems they've worked on?
  - Code should be the documentation
  - The act of solving a problem is more exciting than writing about it
  - We are lazy.
  - The minute you write a doc it is out of date
  - Nobody is ever going to read it
  - It seemed impossible to organize documentation usefully
  - ....
- Chances are:
  - Write a “Update your environment to Java 7” checklist for your team-members
  - Write a “How expensive it really is to migrate to Java 7” blog post
  - Have a good idea about what to bill the customer for



- From a desktop perspective Java 7 is still considered “for developers“ only.
  - You still can’t download it from java.com
  - And your Java 6 will never ask you for an update.
- Some trust is lost
  - The Java 7.0 release had some ugly bugs around <http://blog.eisele.net/2011/07/dont-use-java-7-are-you-kidding-me.html>
  - We have Java 7u1 out since 19/10/11 which solved them
- Not all commercial products ready until today:
  - WebSphere Application Server V8
- But: Adoption of Java SE 7 is rising:
  - WebLogic 12c

**SE 6 still not broadly adopted**





What to expect if you decide to travel?



```
private static final int default_kostenart = 215879;  
private static final int ZZ_BUFFER_SIZE = 16384;
```



```
// ...
    if (o1.getVehicleFeatureDO().getFeatureType().equals(
        "PAINTWORK")
        ||
o1.getVehicleFeatureDO().getFeatureType().equals(
        "UPHOLSTERY")) {
        return -1;
    } else if
(o1.getVehicleFeatureDO().getFeatureType().equals("OPTION")) {
        return 1;
    }
//...
```

```
String s = o1.getVehicleFeatureDO().getFeatureType();  
switch (s) {  
    case "PAINTWORK": case "UPHOLSTERY"  
        return -1;  
    case "OPTION"  
        return 1;  
}
```

```
final List<SelectItem> items = new ArrayList<SelectItem>();
```

```
Collection<KalkFLAufwandsverlaufBE> saveVerlauf = new  
ArrayList<KalkFLAufwandsverlaufBE>();
```

```
final List<ErrorRec> errorRecList = new ArrayList<ErrorRec>();
```

```
List<ProjektDO> resultList = new ArrayList<ProjektDO>();
```

```
Map<String, String> retVal = new HashMap<String, String>();
```

```
final List<SelectItem> items = new ArrayList<>();  
  
Collection<KalkFLAufwandsverlaufBE> saveVerlauf = new  
ArrayList<>();  
  
final List<ErrorRec> errorRecList = new ArrayList<>();  
  
List<ProjektDO> resultList = new ArrayList<>();  
  
Map<String, String> retVal = new HashMap<>();
```

```
public DateiAnhangDO setDateiContent(DateiAnhangDO dateiAnhang, InputStream
dateiInhalt) throws ValidationException {
    OutputStream out = null;
    try {
        File f = new File(dateiAnhang.getDateiNameSystem());
        out = new FileOutputStream(f);
        byte[] buf = new byte[1024];
        int len;
        while ((len = dateiInhalt.read(buf)) > 0) {
            out.write(buf, 0, len);
        }
    } catch (IOException e) {
        // Error management skipped
        throw AppExFactory.validation(CLAZZ, true, errorRecList);
    } finally {
        try {
            out.close();
        } catch (IOException e) {
            // Error management skipped
            throw AppExFactory.validation(CLAZZ, true, errorRecList);
        }
        try {
            dateiInhalt.close();
        } catch (IOException e) {
            // Error management skipped
            throw AppExFactory.validation(CLAZZ, true, errorRecList);
        }
    }
    return dateiAnhang;
}
```

```
public void setDateiContent(String dateiAnhang, InputStream dateiInhalt) throws
ValidationException {
    try (InputStream in = dateiInhalt; OutputStream out = new
FileOutputStream(new File(dateiAnhang))) {
        byte[] buf = new byte[1024];
        int len;
        while ((len = in.read(buf)) > 0) {
            out.write(buf, 0, len);
        }
    } catch (IOException e) {
        // Error management details skipped
        throw AppExFactory.validation(CLAZZ, true, errorRecList);
    }
    return dateiAnhang;
}
```

```
try {
    final Method method = cls.getMethod("getDefault", new Class[0]);
    final Object obj = method.invoke(cls, new Object[0]);
    return (Enum) obj;
} catch (NoSuchMethodException nsmex) {
    throw new EmergencyException(Enum.class, Level.SEVERE, "getDefault method
not found", nsmex);
} catch (IllegalAccessException iae) {
    throw new EmergencyException(Enum.class, Level.SEVERE, "getDefault method
not accessible", iae);
} catch (InvocationTargetException ite) {
    throw new EmergencyException(Enum.class, Level.SEVERE, "getDefault method
invocation exception", ite);
}
```

```
try {
    final Method method = cls.getMethod("getDefault", new Class[0]);
    final Object obj = method.invoke(cls, new Object[0]);
    return (Enum) obj;
} catch (NoSuchMethodException | IllegalAccessException |
IllegalArgumentException | InvocationTargetException nsmex) {
    throw new EmergencyException(Enum.class, Level.SEVERE, "getDefault
method not found", nsmex);
}
```



```

public boolean createZipFile(String workDir, String zipFileName, ZipOutputStream out,
String subDir) {
boolean zipOk = false;
String outFileName = zipFileName;
FileInputStream in = null;
boolean closeZip = true;
String nflen = "";
try {
if (out == null) {
out = new ZipOutputStream(new FileOutputStream(outFileName));
} else {
closeZip = false;
}
if (subDir != null) {
workDir = workDir + "/" + subDir;
}

// Compress the files
File srcDir = new File(workDir);
File[] files = srcDir.listFiles();
byte[] buf = new byte[1024];
for (int i = 0; i < files.length; i++) {
if (zipFileName.equals(files[i].getName())) {
continue;
}
if (files[i].isDirectory()) {
createZipFile(workDir, zipFileName, out, files[i].getName());
continue;
}
in = new FileInputStream(files[i]);

// Add ZIP entry to output stream.
nflen = files[i].getName();
if (subDir != null) {
nflen = subDir + "/" + nflen;
}
out.putNextEntry(new ZipEntry(nflen));

// Transfer bytes from the file to the ZIP file
int len;
while ((len = in.read(buf)) > 0) {
out.write(buf, 0, len);
}

// Complete the entry
out.closeEntry();
in.close();
zipOk = true;
}

// Complete the ZIP file
} catch (FileNotFoundException e) {
//skipped

} catch (IOException e) {
//skipped

} finally {
try {
if (in != null) {
in.close();
}
} catch (IOException e) {
//skipped

}
try {
if (closeZip && out != null) {
out.close();
}
} catch (IOException e) {
//skipped

}
}
return zipOk;
}

```

Recursive ZIP File Handling  
- ZipOutputStream  
- ZipEntry

Error Handling (condensed ☹)

```

public static void create(String zipFilename, String... filenames)
    throws IOException {

    try (FileSystem zipFileSystem = createZipFileSystem(zipFilename, true)) {
        final Path root = zipFileSystem.getPath("/");

        //iterate over the files we need to add
        for (String filename : filenames) {
            final Path src = Paths.get(filename);

            //add a file to the zip file system
            if(!Files.isDirectory(src)){
                final Path dest = zipFileSystem.getPath(root.toString(),
                                                         src.toString());

                final Path parent = dest.getParent();
                if(Files.notExists(parent)){
                    System.out.printf("Creating directory %s\n", parent);
                    Files.createDirectories(parent);
                }
                Files.copy(src, dest, StandardCopyOption.REPLACE_EXISTING);
            }
            else{
                //for directories, walk the file tree
                Files.walkFileTree(src, new SimpleFileVisitor<Path>(){
                    @Override
                    public FileVisitResult visitFile(Path file,
                                                       BasicFileAttributes attrs) throws IOException {
                        final Path dest = zipFileSystem.getPath(root.toString(),
                                                                file.toString());
                        Files.copy(file, dest, StandardCopyOption.REPLACE_EXISTING);
                        return FileVisitResult.CONTINUE;
                    }

                    @Override
                    public FileVisitResult preVisitDirectory(Path dir,
                                                             BasicFileAttributes attrs) throws IOException {
                        final Path dirToCreate = zipFileSystem.getPath(root.toString(),
                                                                      dir.toString());

                        if(Files.notExists(dirToCreate)){
                            System.out.printf("Creating directory %s\n", dirToCreate);
                            Files.createDirectories(dirToCreate);
                        }
                        return FileVisitResult.CONTINUE;
                    }
                });
            }
        }
    }
}

```

```

private static FileSystem createZipFileSystem(String
zipFilename, boolean create) throws IOException {
    // convert the filename to a URI
    final Path path = Paths.get(zipFilename);
    final URI uri = URI.create("jar:file:" +
path.toUri().getPath());

    final Map<String, String> env = new HashMap<>();
    if (create) {
        env.put("create", "true");
    }
    return FileSystems.newFileSystem(uri, env);
}

```

- Nice, new features
- Moving old stuff to new stuff isn't an automatism
- Most relevant ones:
  - Multi-Catch probably the most used one
  - Diamond Operator
- Maybe relevant:
  - NIO.2
  - Fork/Join
- Not relevant: (at least to the projects I looked at!)
  - Better integer literals
  - String in switch case
  - Varargs Warnings
  - InvokeDynamik

“The best thing about a boolean is  
even if you are wrong,  
you are only off by a bit.”

(Anonymous)

<http://www.devtopics.com/101-great-computer-programming-quotes/>

**Thank you for your attention**

**Markus Eisele**

Principal IT Architect

Phone: +49 89 96101-0  
markus.eisele@msg-systems.com

[www.msg-systems.com](http://www.msg-systems.com)



.consulting .solutions .partnership

