

Using and contributing to Open Source

by Geoffrey De Smet
OptaPlanner lead

Who am I?

- Java developer (graduated in 2003)
- OptaPlanner Founder/Lead
 - Customers and users around the world
 - 1k+ downloads per month
 - 255k lines of code

Reusing code
is better than
writing it
(99% of the time)

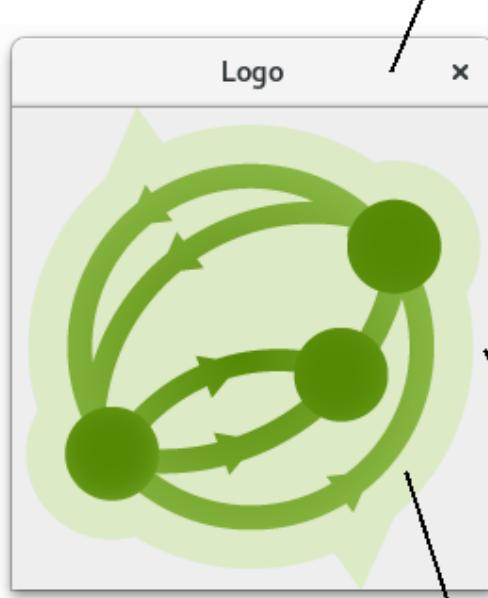


```
import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JLabel;

public class LogoApp extends JFrame {
    ...

    public LogoApp() {
        super("Logo");
        ImageIcon image = new ImageIcon(
            getClass().getResource("logo.png"));
        getContentPane().add(new JLabel(image));
    }
}
```

```
import javax.swing.JFrame;
```



```
import javax.swing.ImageIcon;
```

```
import javax.swing.JLabel;
```

Use other people's code

```
import javax.swing.ImageIcon;  
import javax.swing.JFrame;  
import javax.swing.JLabel;
```

⇒ my dependencies

To reuse

Not to reuse

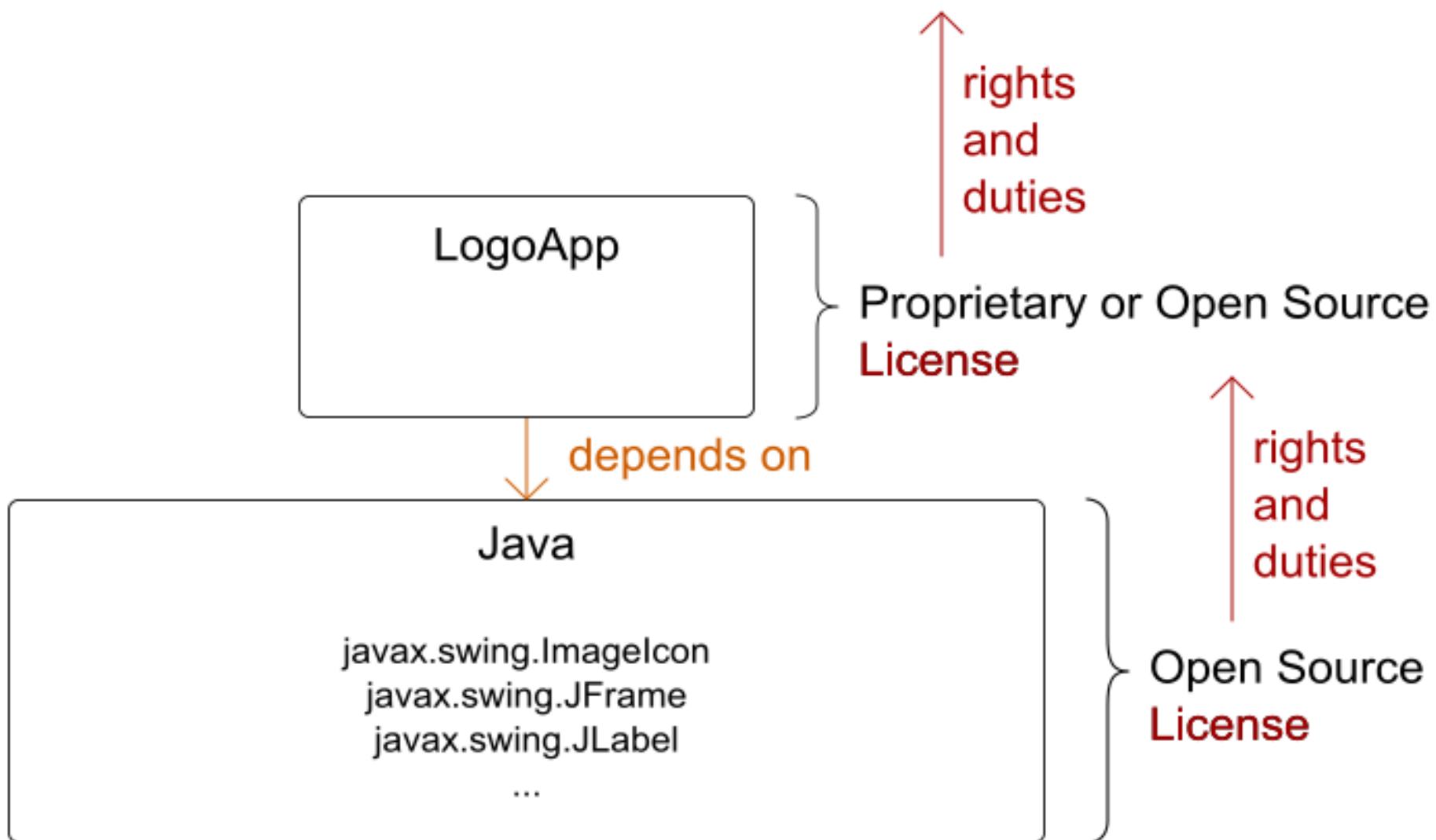
Show an image on the screen

```
import javax.swing.ImageIcon;
```

	To reuse	Not to reuse
Development time	1 minute	2 years
Quality	High Written by specialists Tested by millions	Low Only PNG (no JPG) Exploits, slow, ...
Maintenance time	low (upgrades)	high (rewrites)
Dependencies?	Yes	No

An extra dependency is the lesser evil.

LogoApp depends on Java



WHEN YOU PROGRAM OPEN SOURCE,
YOU'RE PROGRAMMING
COMMUNISM



A REMINDER
from
YOUR FRIENDS AT MICROSOFT

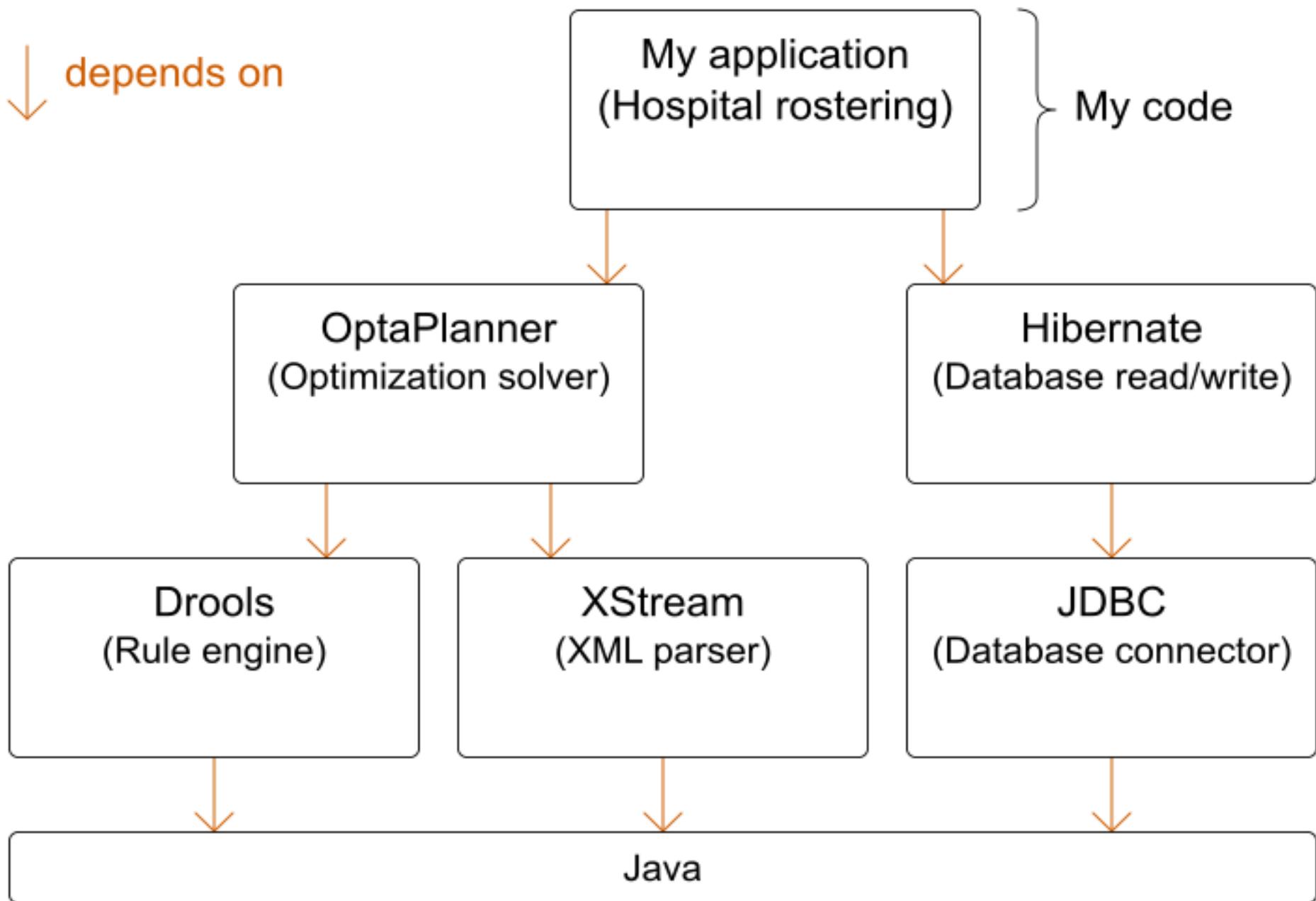
No longer the 90's

- Java is Open Source
 - <http://openjdk.java.net/>
- C++: GNU's GCC is Open Source
 - <https://gcc.gnu.org/>
- .NET Core is Open Source
 - <https://github.com/Microsoft/dotnet>

You're already using Open Source.

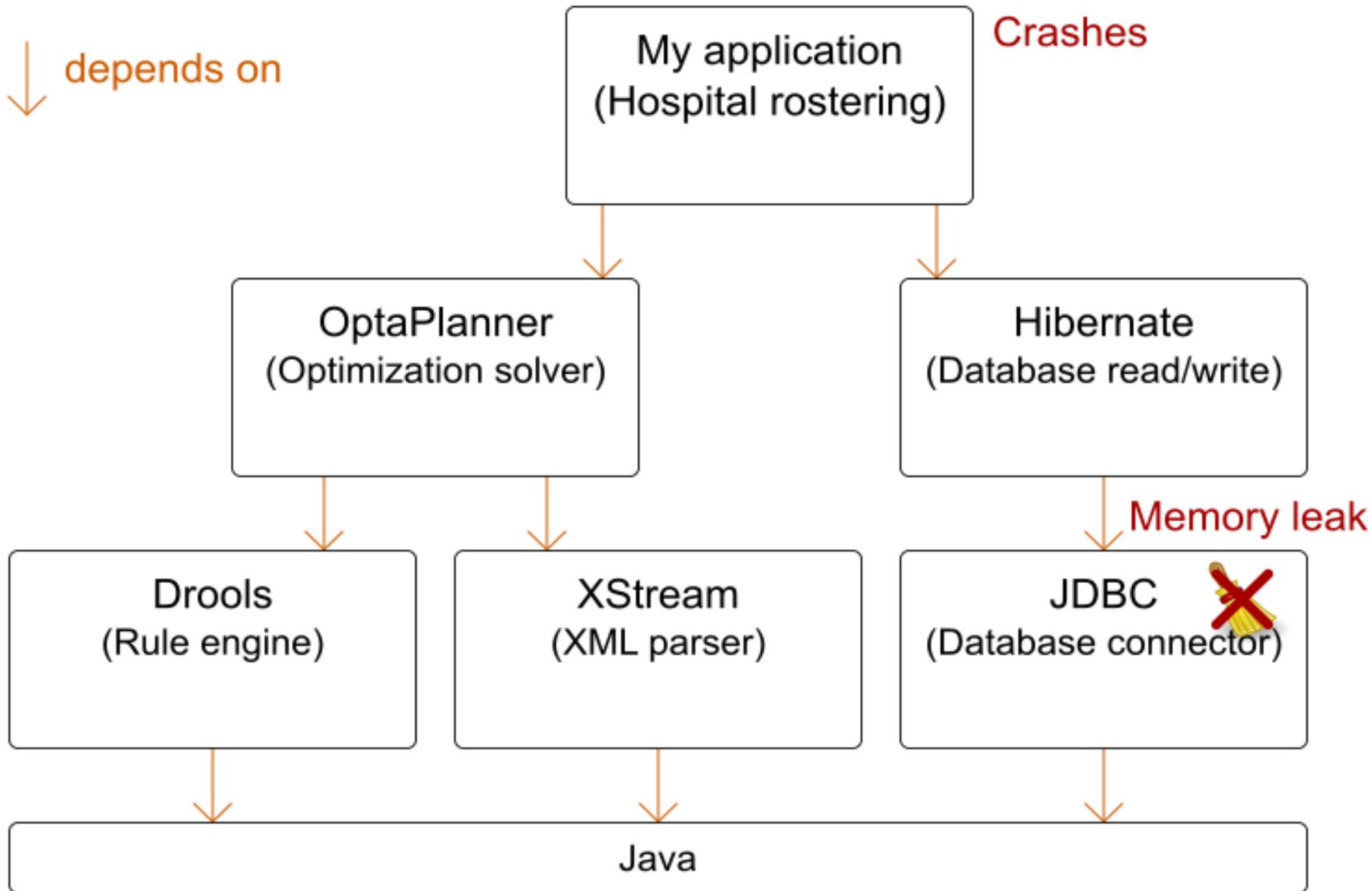
Typical web application has
100+ dependencies
and millions of imports

Dependency Tree



Every dependency
can harm you.

Dependency Tree



Depend only
on **mature** projects

What makes
(Open Source) projects
mature?

What makes (Open Source) projects mature?

1. License

IANAL

Code

Pull requests 7

Projects 0

Pulse

Graphs

Java planning engine: OptaPlanner optimizes business resource scheduling and planning. <http://www.optaplanner.org>

5,669 commits

15 branches

71 releases

60 contributors

Apache-2.0

Branch: master ▾

New pull request

Find file

Clone or download ▾

geOffrey	PLANNER-536 Allow overriding getScore(): extend to non-AbstractSoluti...	...	Latest commit <code>cec03d7</code> 2 days ago
optaplanner-benchmark	Add home page URL to all POMs (#215)		a month ago
optaplanner-core	PLANNER-536 Allow overriding getScore(): extend to non-AbstractSoluti...		2 days ago
optaplanner-distribution	Add home page URL to all POMs (#215)		a month ago
optaplanner-docs	docs: fix the kie-karaf-features reference (#218)		18 days ago
optaplanner-examples	Add home page URL to all POMs (#215)		a month ago
optaplanner-persistence	Add home page URL to all POMs (#215)		a month ago
optaplanner-test	Add home page URL to all POMs (#215)		a month ago
optaplanner-webexamples	Add home page URL to all POMs (#215)		a month ago
.gitattributes	gitattributes: Add pdf to binary files		a year ago
.gitignore	.gitattributes to force unix line endings in this repository		3 years ago
LICENSE-ASL-2.0.txt	JBRULES-2869: Split-off requires license file per repo		6 years ago
README.adoc	Added starter issue info to readme		7 months ago
pom.xml	Add optaplanner-test module		3 months ago
README.adoc			



Download OptaPlanner

6.4.0.Final

Try the examples now:

1. Download the zip and unzip it
2. On Linux/Mac, run [examples/runExamples.sh](#)
On Windows, run [examples/runExamples.bat](#)

Requires Java™ to run.



Read documentation

6.4.0.Final

Read the [Quick Start chapter](#).

What is O

OptaPlanner is a **constraint solver** for business resource planning. Every organization has constrained resources (like time, money, people) to assign a limited set of services to customers. OptaPlanner optimizes business resource allocation to do more with less.

[License](#)[Release notes](#)[Upgrade recipe](#)

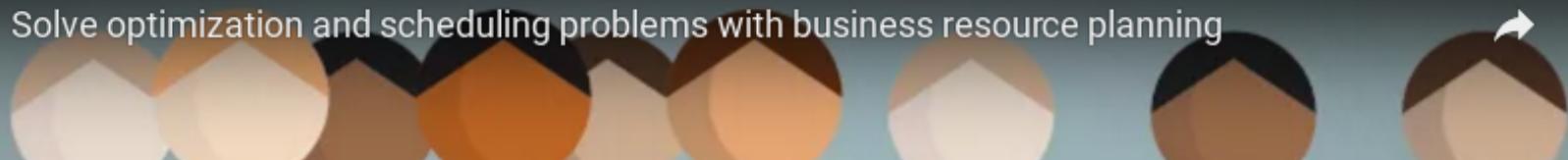
Rostering, Cloud Optimization, Job Scheduling, Bin Packing and many more.

OptaPlanner is a **lightweight, embeddable planning engine**. It enables normal Java™ programmers to solve optimization problems efficiently. Constraints apply on plain domain objects and can reuse existing code. There's no need to input difficult mathematical equations. Under the hood, OptaPlanner combines sophisticated optimization heuristics and metaheuristics (such as Tabu Search, Simulated Annealing and Late Acceptance) with very efficient score calculation.

OptaPlanner is **open source software**, released under [the Apache Software License](#). It is written in 100% pure Java™, runs on any JVM and is available in [the Maven Central repository](#) too.

[Video](#)[Use cases](#)[Usage](#)[Integration](#)[Compatibility](#)[Scalability](#)

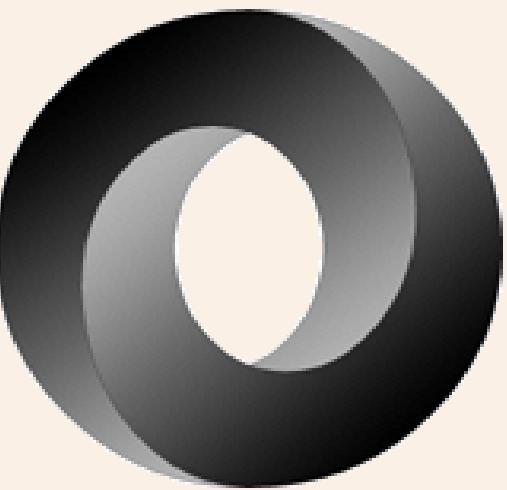
Solve optimization and scheduling problems with business resource planning



What is Open Source?

- Source code included
- Free redistribution
- Right to modify
- No fields of use restrictions
- ... (see opensource.org)

⇒ Formalized by a License



The JSON License

Copyright (c) 2002 JSON.org

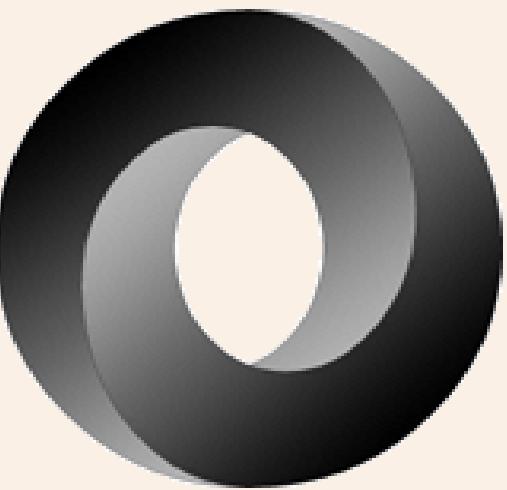
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS





The JSON License

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil. ⇒ Not Open Source

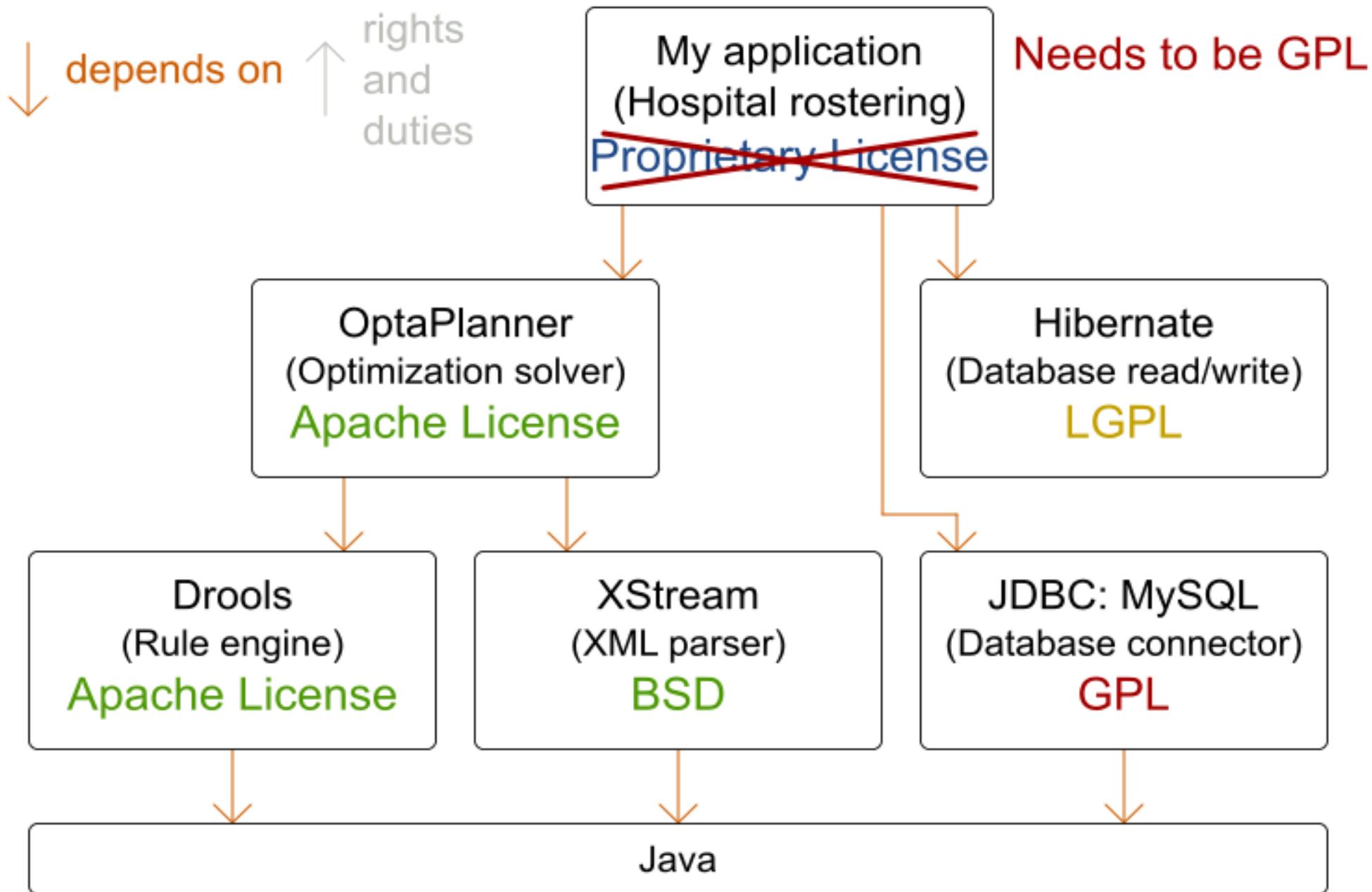
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS

Normal Open Source licenses

- MIT
- BSD
- ASL: Apache License
- MPL: Mozilla Public License
- EPL: Eclipse Public License
- LGPL: Lesser General Public License
- GPL: General Public License
- AGPL

Avoid other Open Source licenses.

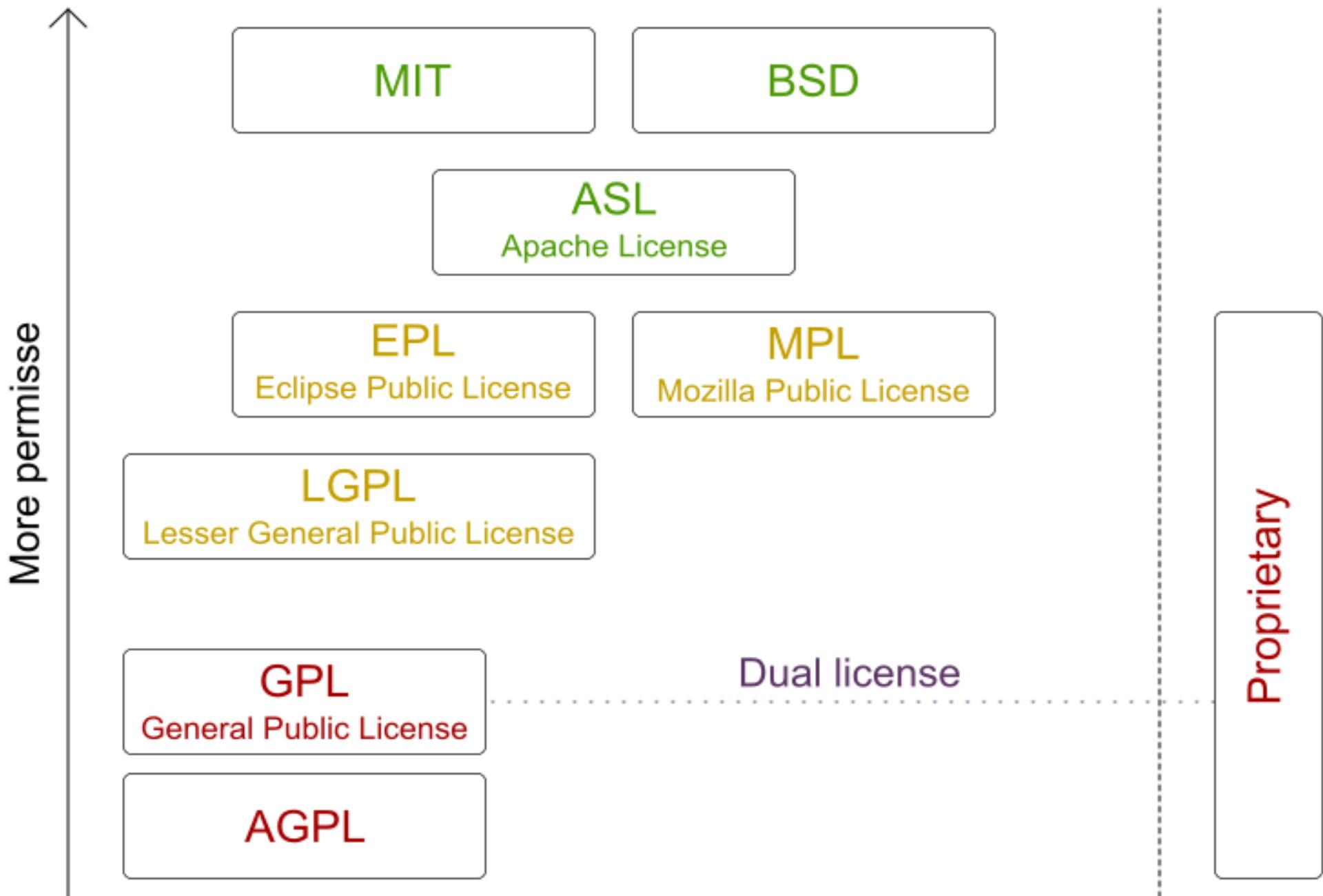
License impact



Make 2 lists of licenses

- Whitelist: approved by legal
- Blacklist: rejected by legal
- Other: ask legal before importing

Commercial friendliness for consumers



FAQ

- Isn't OpenJDK (Java) GPL?
 - **GPL with classpath exception**
 - Does not affect normal Java code
- Isn't GCC (the C++ compiler) GPL?
 - Does not affect your compiled code

I_ANAL

I am not a lawyer

All licenses (open source or proprietary)
include legal uncertainty
until tried in court.

Red Hat includes indemnification for its customers.

What makes (Open Source) projects mature?

2. Distribution and download

Recent stable release



- There is a stable release
 - Stable releases: Final, GA
 - Unstable releases: Alpha, Beta, CR, RC, Milestone
- Released in the last year
 - 8% out of 500 000 tracked open source projects
(according to openhub.net in 2012)

Available on central repository

- Java: in **Maven Central** repository
 - Bad name: Gradle, Ivy and Buildr use it too
- .NET: in NuGet Gallery
- Ruby: in rubygems.org

Similar to Android's Play store, iPhone's App store,
Fedora's rpm repository, Ubuntu's deb repository, ..

search.maven.org / mvnrepository.com

← → × ⓘ mvnrepository.com/search?q=optaplanner

MVNREPOSITORY

optaplanner

Search

Artifacts/Year

Found 39 results

Search in central repositories

1. **OptaPlanner Core** 23 usages
org.optaplanner » optaplanner-core
OptaPlanner solves planning problems. This lightweight, embeddable planning engine implements powerful and scalable algorithms to optimize business resource scheduling and planning. This module contains the planning engine itself.

2. **OptaPlanner Benchmark** 5 usages
org.optaplanner » optaplanner-benchmark
OptaPlanner solves planning problems. This lightweight, embeddable planning engine implements powerful and scalable algorithms to optimize business resource scheduling and planning. This module contains the benchmark toolkit.

3. **OptaPlanner Persistence Common** 6 usages
org.optaplanner » optaplanner-persistence-common
OptaPlanner solves planning problems. This lightweight, embeddable planning engine implements powerful and scalable algorithms to optimize business resource scheduling and planning. This module contains the common persistence integration.

Popular Categories

- Aspect Oriented
- Actor Frameworks
- Application Metrics
- Build Tools
- Bytecode Libraries
- Command Line Parsers
- Cache Implementations
- Cloud Computing
- Code Analyzers
- Collections
- Configuration Libraries

What makes (Open Source) projects mature?

3. Source control

Source Control

- Stored in
 - Git
 - Mercurial
 - Subversion
 - Fork unfriendly ⇒ collaboration unfriendly
- Hosted on
 - **GitHub.com**
 - Bitbucket.org
 - GitLab.com

GitHub commit history

Screenshot of a GitHub commit history page for the repository `droolsjbpm / optaplanner`.

The page shows commits from Sep 19, 2016, to Sep 21, 2016.

Key statistics at the top:

- Watched by 97 users
- Starred by 531 users
- Forked by 300 users

Recent commits (Sep 21, 2016):

- PLANNER-536 Allow overriding getScore(): extend to non-AbstractSoluti...** by **geOffrey** committed 4 days ago. Commit hash: `cec03d7`. Actions: copy, view, diff.
- PLANNER-536 Workbench solver editor: bendable score combobox selectio...** by **mcimbora** committed with **geOffrey** 4 days ago. Commit hash: `ea3bc39`. Actions: copy, view, diff.
- DROOLS-1276 unignore tests** by **geOffrey** committed 4 days ago. Commit hash: `8dc9877`. Actions: copy, view, diff.

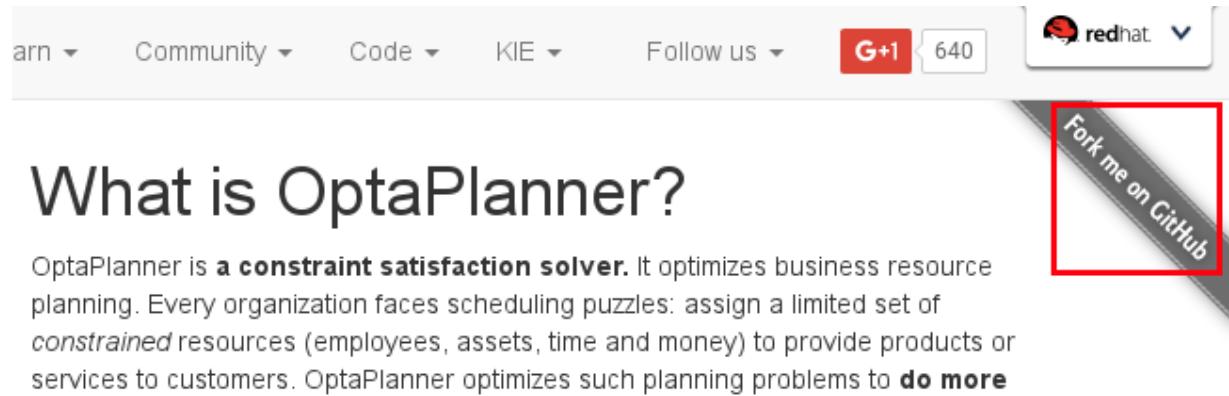
Recent commits (Sep 19, 2016):

- PLANNER-647 javadocs on new BendableScore methods** by **geOffrey** committed 6 days ago. Commit hash: `4830a34`. Actions: copy, view, diff.
- PLANNER-647 Workbench: Include ErraiApp.properties and score marshall...** by **mcimbora** committed with **geOffrey** 6 days ago. Commit hash: `e9819ab`. Actions: copy, view, diff.

Easy to build from source

```
$ git clone https://github.com/droolsjbpm/optaplanner.git  
..  
$ cd optaplanner  
$ mvn clean install -DskipTests  
..
```

Fork it!



The screenshot shows the OptaPlanner website's header with navigation links: "arn", "Community", "Code", "KIE", "Follow us", "G+1 640", and "redhat". Below the header, the main content area has a title "What is OptaPlanner?". A paragraph describes OptaPlanner as a constraint satisfaction solver. To the right of the paragraph is a "Fork me on GitHub" button, which is highlighted with a red rectangular border.

arn ▾ Community ▾ Code ▾ KIE ▾ Follow us ▾ G+1 640 redhat ▾

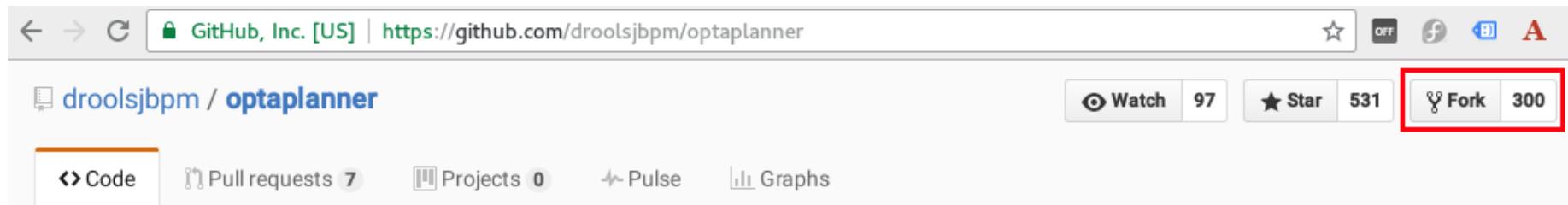
What is OptaPlanner?

OptaPlanner is a **constraint satisfaction solver**. It optimizes business resource planning. Every organization faces scheduling puzzles: assign a limited set of *constrained* resources (employees, assets, time and money) to provide products or services to customers. OptaPlanner optimizes such planning problems to **do more**

My fork = my personal sandbox

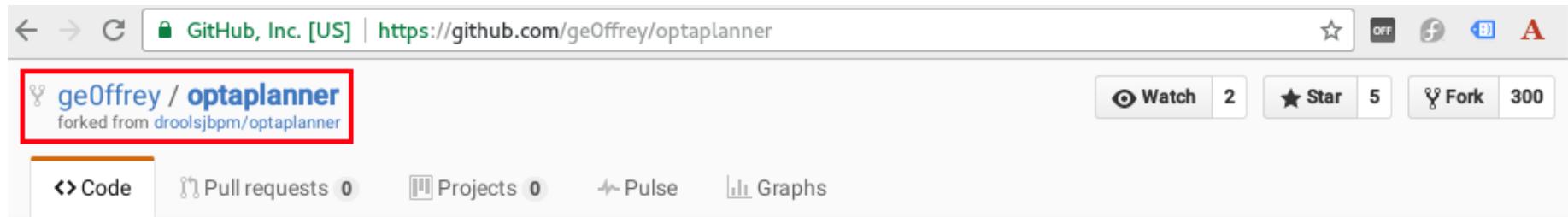
Easy to fork

Blessed (official releases):



A screenshot of a GitHub repository page. The URL in the address bar is <https://github.com/droolsjbpm/optaplanner>. The repository name is **droolsjbpm / optaplanner**. In the top right, there are buttons for Watch (97), Star (531), and Fork (300). The 'Fork' button is highlighted with a red box. Below the header, there are tabs for Code (which is selected), Pull requests (7), Projects (0), Pulse, and Graphs.

My fork:



A screenshot of a GitHub repository page. The URL in the address bar is <https://github.com/geOffrey/optaplanner>. The repository name is **geOffrey / optaplanner**, with a note below it saying "forked from [droolsjbpm/optaplanner](#)". In the top right, there are buttons for Watch (2), Star (5), and Fork (300). The 'Fork' button is highlighted with a red box. Below the header, there are tabs for Code (selected), Pull requests (0), Projects (0), Pulse, and Graphs.

My changes do *not* affect *blessed*.

Easy to pull new commits from *blessed*.

Why share your modifications?

- Reduce maintenance
 - Your modifications are in all future releases
 - No merging with upstream changes
 - No need to do releases
- Get feedback from project committers
- Gain respect (= power in a meritocracy)

How do you share your modifications?

- Patch file
 - Goes stale
 - No review support
- Pull Request
 - Send Pull Request
 - Committers review it
 - Fix review comments
 - Committers merge it in *blessed*

List of Pull Requests

<input type="checkbox"/>	 docs: fix the kie-karaf-features reference ✓	<small>#218 by psiroky was merged 21 days ago</small>	
<input type="checkbox"/>	 PLANNER-647 Workbench: Include ErraiApp.properties and score marshalling mappings in workbench-upstream-model module ✓	<small>#217 by mcimbora was merged 7 days ago</small>	 1
<input type="checkbox"/>	 Reproducer for PLANNER-642: modified vehicle routing example (not fix)	<small>#216 opened on 26 Aug by nbolleboom</small>	 3
<input type="checkbox"/>	 Add home page URL to all POMs ✓	<small>#215 by yurloc was merged on 24 Aug</small>	 2
<input type="checkbox"/>	 PLANNER-580 Make org.optaplanner.core.api.score.Score class hierarchy JavaScript-compile friendly ✓	<small>#214 by mcimbora was merged on 24 Aug</small>	 7
<input type="checkbox"/>	 Benchmark report consistency fix ✓	<small>#213 by yurloc was merged on 20 Aug</small>	 4
<input type="checkbox"/>	 Trivial typo fixes	<small>#212 by bibryam was merged on 8 Aug</small>	 3

Trivial typo fixes #212

Merged geOffrey merged 1 commit into droolsjbpm:master from bibryam:docs on 8 Aug

Conversation 3

Commits 1

Files changed 2

Changes from all commits ▾ 2 files ▾ +2 -2

Unified

Split

Review changes ▾

2 ...cs/src/main/docbook/en-US/Chapter-Planner_configuration/Chapter-Planner_configuration.xml

View

		@@ -10,7 +10,7 @@
10	10	<section xml:id="plannerConfigurationOverview">
11	11	<title>Overview</title>
12	12	
13	-	<para>Solving a planning problem with Planner consists of 5 steps:</para>
	13	+ <para>Solving a planning problem with Planner consists of the following steps:</para>
14	14	
15	15	<orderedlist>
16	16	<listitem>

2 ...anner-docs/src/main/docbook/en-US/Chapter-Score_calculation/Chapter-Score_calculation.xml

View

		@@ -1119,7 +1119,7 @@ end</programlisting>
1119	1119	are initialized (while the already initialized variables do not change). Some optimization algorithms (such
1120	1120	Construction Heuristics and Exhaustive Search) run faster if they have such information.</para>
1121	1121	
1122	-	<para>For the Score (or each <link linkend="scoreLevel">score level</link> separately), specify a
	1122	+ <para>For the Score (or each <link linkend="scoreLevel">score level</link> separately), specify a
1123	1123	trend:</para>
1124	1124	
1125	1125	<itemizedlist>

Review

4  ...anner-benchmark/src/main/java/org/optaplanner/benchmark/impl/report/BenchmarkReport.java Show notes View

⊕		@@ -498,8 +498,8 @@ private void writeAverageCalculateCountPerSecondSummaryChart() {
498	498	}
499	499	XYPlot plot = createScalabilityPlot(seriesList,
500	500	"Problem scale", NumberFormat.getInstance(locale),
501	-	"Average calculate count per second", NumberFormat.getInstance(locale));
502	-	JFreeChart chart = new JFreeChart("Average calculate count summary (higher is better)",
	501	+ "Average calculation count per second", NumberFormat.getInstance(locale));
	502	+ JFreeChart chart = new JFreeChart("Average calculation count summary (higher is better)",

 **geOffrey** on 19 Aug
Some time ago, I renamed "average calculation count per second (999)" to "score calculation speed (999/sec)". Looks like I missed these - we should rename them also.

 **yurloc** on 19 Aug
Ahh, sorry, I forgot to include an important notice: this is for 6.x stream, the fix doesn't apply to master.

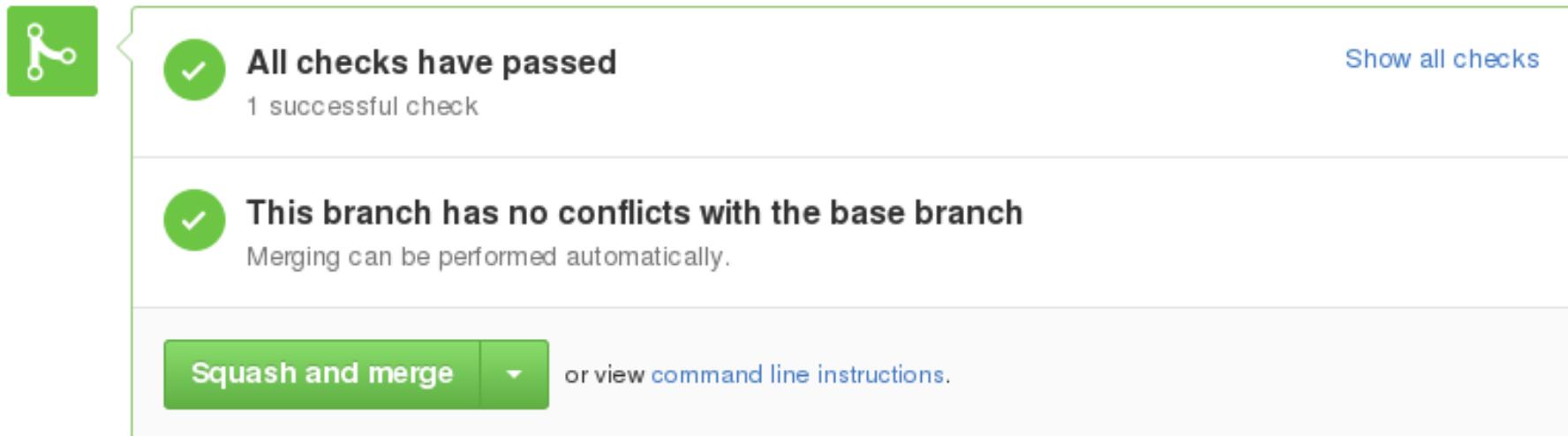
 **geOffrey** on 20 Aug
ah, I hadn't notice that, merging, thanks :)

 Reply...

503	503	JFreeChart.DEFAULT_TITLE_FONT, plot, true);
504	504	averageCalculateCountSummaryChartFile = writeChartToImageFile(chart, "averageCalculateCountSummary");
505	505	}

Merge a Pull Request

Add more commits by pushing to the **test-coverage** branch on **yurloc/optaplanner**.



A screenshot of the GitHub Checks API interface. It shows two green checkmarks indicating success. The first checkmark says "All checks have passed" with "1 successful check". The second checkmark says "This branch has no conflicts with the base branch" with the note "Merging can be performed automatically." Below these, there is a green button labeled "Squash and merge" with a dropdown arrow, and a link to "or view command line instructions".

All checks have passed
1 successful check

This branch has no conflicts with the base branch
Merging can be performed automatically.

Squash and merge ▾ or view command line instructions.

Fork merged to blessed

-
- A vertical timeline of commits from a fork to a blessed branch. The commits are represented by blue dots connected by a red line. The timeline starts at the bottom with a vertical orange line.
- remove xsteam references in test files
 - Merge pull request #177 from oskopek/PLANNER-415
 - PLANNER-415: Added a null check
 - Removed unused variable causing errors on benchmark report generation
 - PLANNER-519 Race condition with Solver.getBestSolution()

Geoffrey De Smet
Geoffrey De Smet
Ondrej Skopek <s
Ondrej Skopek <s
Geoffrey De Smet

What makes (Open Source) projects mature?

4. Business model

Governance

Who controls an Open Source project?

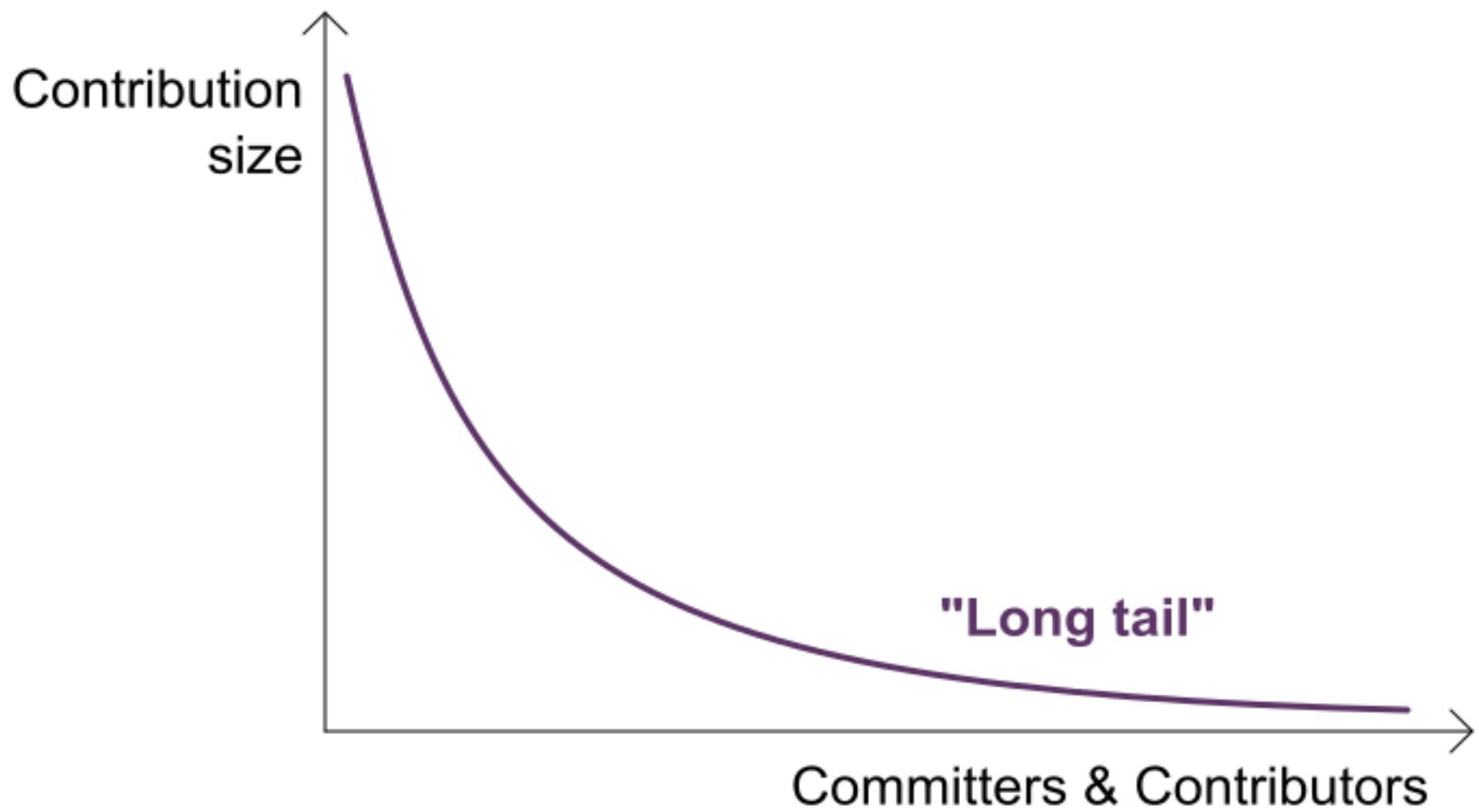
Meritocracy

Profile	Project benefit	Person benefit	How many?	Code written
Committer Write rights	Develops features Writes documentation	Accept/reject PR's Votes in discussions	1-20 per project	90%
Customer Pays for support	Indirectly pays wages of committers	Roadmap influence Prioritized issues Support/certification		
Contributor Contributed code	Bugfixes & improvements for own use	Gains respect Influence in discussions	5%	10%
User Uses project	Word of mouth Reports issues	Well reported issues are considered	95%	0%

Communication

- Issue tracker
- Mailing list
 - Use gmane.org (NNTP or web) to avoid flood
- IRC (90's chat, like Slack)
 - Most Open Source projects are on Freenode
 - Bot mentions new commits/issues ⇒ sparks discussions
 - Video conferencing for complex discussions

Healthy Open Source project



Healthy Open Source project

- Some of the committers needs to be payed to work on it full time.
- Possible to buy support

Governance models

- Multi-company model
 - Apache Foundation
 - Eclipse Community
- One-company model
 - Red Hat
 - Pivotal
 - Oracle
 - Docker Inc.
 - ...

Apache Foundation / Eclipse Community

- Neutral ground
- Infrastructure sponsored
- No dictator
- Voting system:
 - +1: Do the change
 - +0: I don't care
 - -1: I veto the change (requires motivation)
- Politics can kill successful projects
 - Log4J 1.x died when main committer got veto'ed
 - Maven 3 flatlined after dispute with main contributing company (now mostly fixed)

One company model

- Clear, sustainable business model
 - Paid full-time committers
- Fear of fork reduces power abuse
 - OpenOffice got forked as LibreOffice
 - Hudson got forked as Jenkins
- Often benevolent dictator (project lead)
 - Listens to company and community
- Potential conflict of interest
 - Depends on the business model
 - Balancing act for project lead

Open Source business models

- Dual-licensing (GPL/proprietary)
 - MySQL
- Sell proprietary enterprise features
 - IntelliJ
- Sell support/consultancy
 - All Red Hat projects
 - IBM projects
 - Most Apache projects
- Get contributions on not core business software
 - Google projects
 - Facebook projects

What makes (Open Source) projects mature?

5. Everything that makes proprietary
projects mature

Typical quality indicators

- Unit tests
- Test coverage
- Continuous integration
- Documentation

Is Open Source better than proprietary?

Depends on the project!

Summary

Summary

- Reuse is better than writing it yourself
- Have only mature projects in your dependency tree
- Look for maturity indicators
 - Open Source projects can't hide them

Q & A

- Slides** geOffrey.github.io/geOffrey-presentations/
[\(https://geOffrey.github.io/geOffrey-presentations/\)](https://geOffrey.github.io/geOffrey-presentations/)
- OptaPlanner** www.optaplanner.org
[\(http://www.optaplanner.org\)](http://www.optaplanner.org)
- Red Hat** www.redhat.com (<http://www.redhat.com>)
- Feedback**  @GeoffreyDeSmet
[\(https://twitter.com/GeoffreyDeSmet\)](https://twitter.com/GeoffreyDeSmet)