

**Institute of Systems Science
National University of Singapore**

**GRADUATE CERTIFICATE
INTELLIGENT REASONING SYSTEMS**

Workshop Project (Continuous Assessment) Guide

Subject: *Reasoning Systems*

Agenda : Course Assessment & Grading

➡ EEP & MTech Stackable

- **Paper Assessment** on last lecture day
 - [Individual] 50 marks] 1 hour open book exam (course level)
- **Workshop Project Deliverables** due last lecture day
 - [Individual] 25 marks] A runnable standalone bespoke hybrid reasoning system
 - [Individual] 25 marks] A project report with relevant attachments, including
 - System Design / Knowledge Models
 - Use/Test cases

➡ MTech Thru-Train

- **Paper Assessment** on last lecture day
 - [Individual] 50 marks] 1 hour open book exam (course level)
- **Workshop Project Deliverables** [due last lecture day + 14]
 - [Group] 20 marks] A runnable standalone bespoke hybrid reasoning system
 - [Group] 20 marks] A project report with relevant attachments, including
 - System Design / Knowledge Models
 - System Development & Implementation in tools, e.g. KIE suite
 - System User Guide
 - [Group] 10 marks] A 5 minute video presentation, covering
 - System Design & Use Case Demo

- **Source impactful real life business scenario for workshop project.**
- **Conduct comprehensive research and reference reading.**
- **Make your own reasonable assumptions where necessary.**

Workshop 1 Guide

WORKSHOP SEARCH REPRESENTATION

- **Search Modelling & Representation**

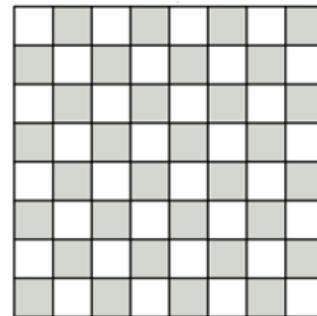
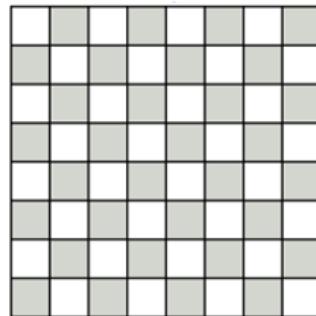
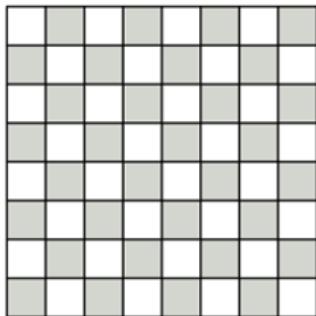
- Pen & Paper Planning
- Robot Navigation
- Vehicle Route Planning (VRP)

- **KIE OptaPlanner Tutorial**

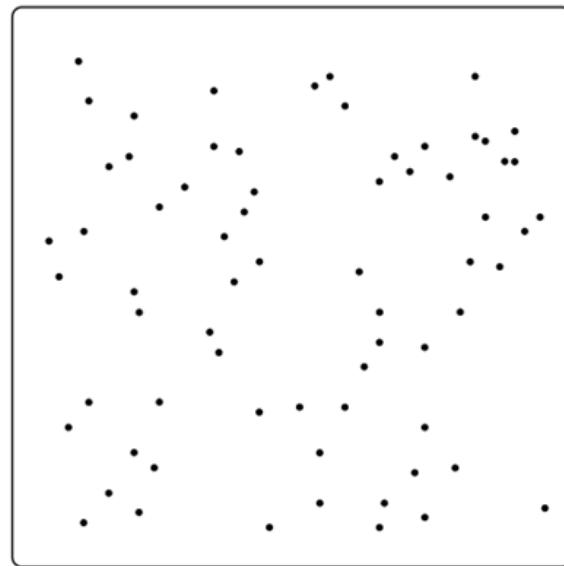
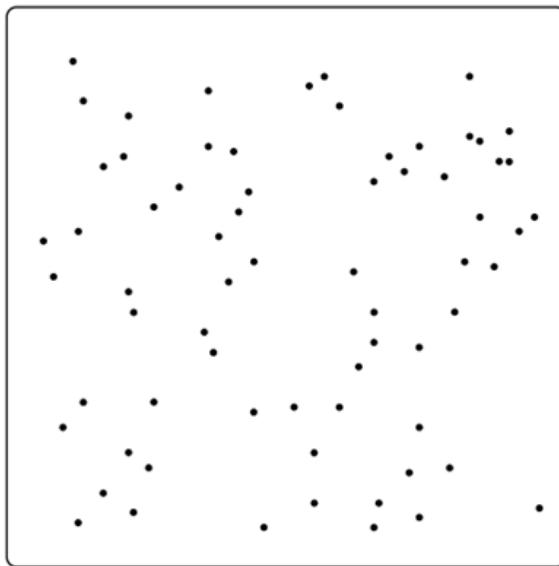
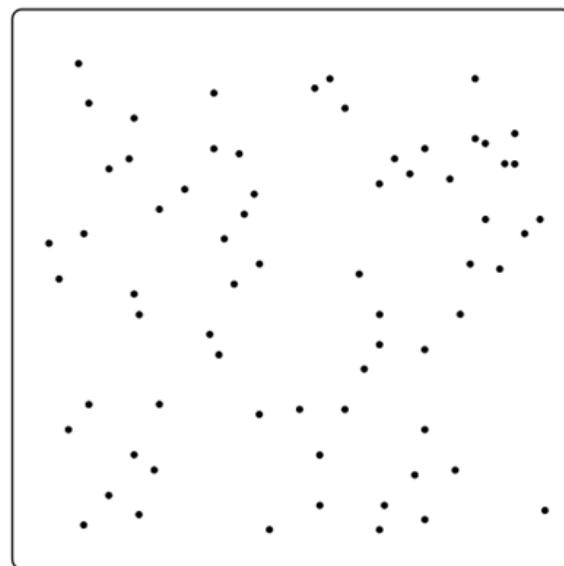
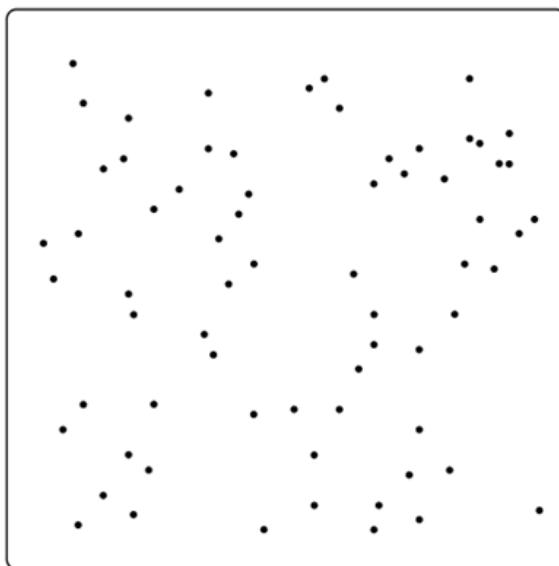
- Optimizing Vehicle Route Planning (VRP)
- Optimizing Europe Travelling Sales Person (TSP)

Workshop 1.1 [Individual]

1) Place 8 queens on this chessboard so no 2 queens can attack each other.

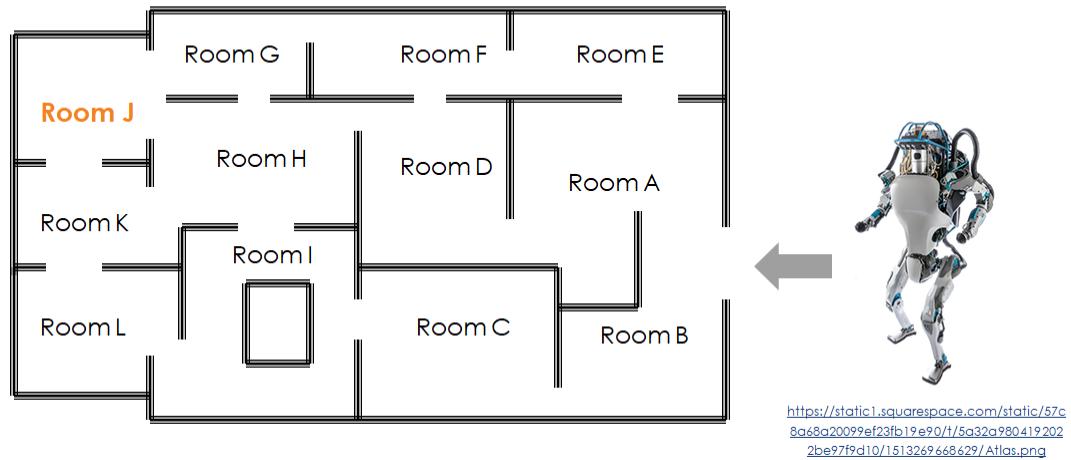


2) Draw the shortest line that connects all dots and returns to its origin.



Workshop 1.2 [Individual]

- **Robotics: How to rapidly navigate to Room J ?**



Design and draft robot navigation search representation here:

Write down DFS search order:

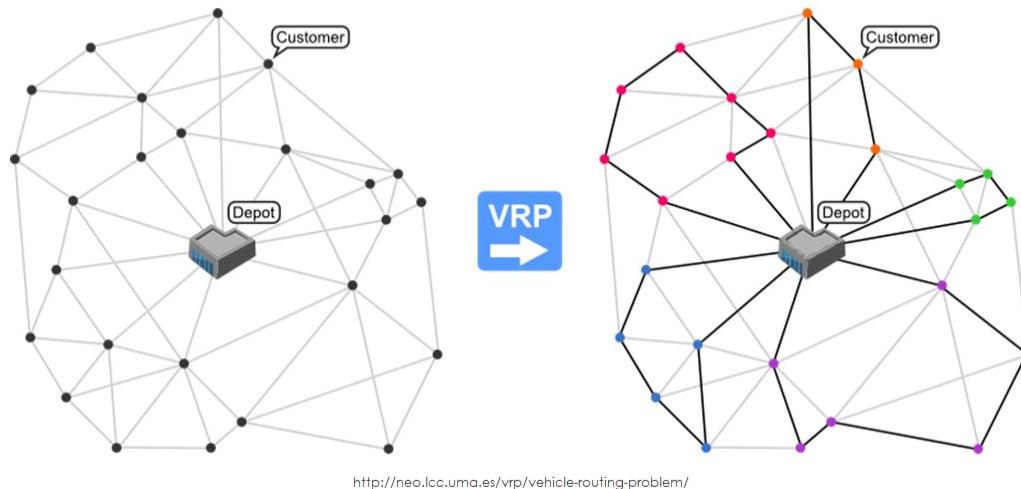
Write down BFS search order:

Construct reasonable heuristics, then design heuristic search strategy to enhance above DFS/BFS brute force search:

Workshop 1.3 [Group]

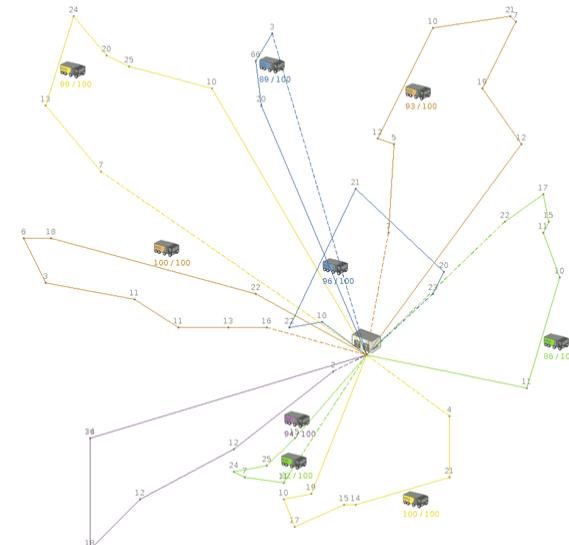
Form groups: 4-6 learners per group.

Vehicle Route Problem



Optimizing Vehicle Route Planning

- We are a logistic company owning a warehouse and 9 delivery trucks. This morning we received 54 customer orders, with different load demand, and different locations. Our truck's maximum load capacity is 100 TVs.
- We want to delivery all customer orders using fewer gasoline. Hence, we'd like to have shortest distance of combined truck delivery routes.

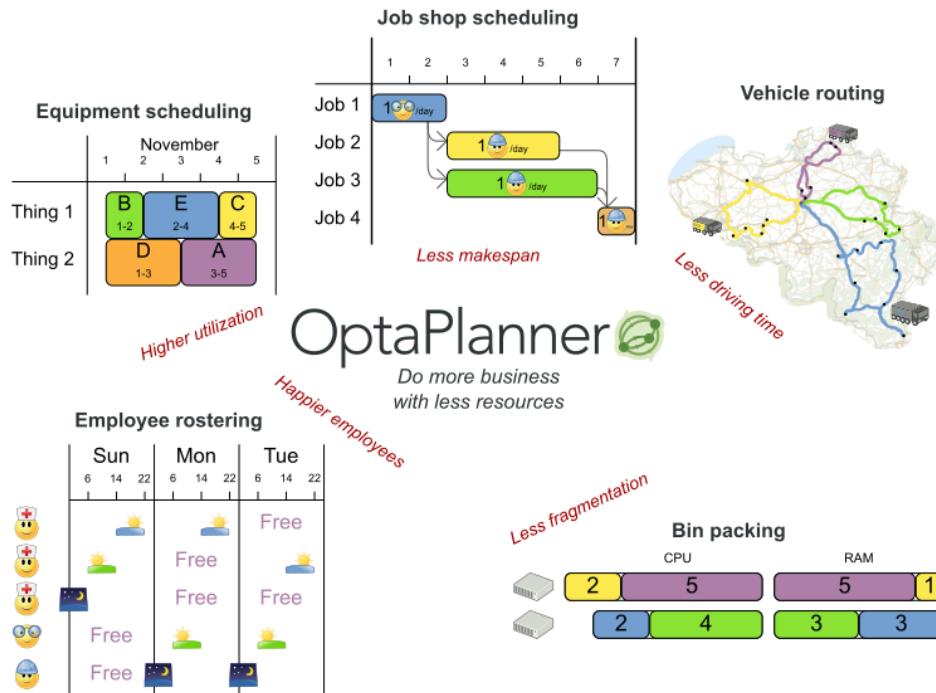


Design and draft VRP search representation here:

Strategy to avoid looping (revisiting same customer):

Workshop 1.4 [Individual]

KIE OptaPlanner Tutorial: Installation; KIE Workbench; IDE plug-in; Case Study



The screenshot shows the KIE GROUP website with the OptaPlanner section highlighted. The page includes the KIE GROUP logo, navigation links for DROOLS, OPTAPLANNER, JBPM, and APPFORMER, and a Red Hat logo.

OPTAPLANNER

OptaPlanner is a lightweight, embeddable planning engine. It enables normal Java™ programmers to solve optimization problems efficiently. It is also compatible with other JVM languages (such as Kotlin and Scala).

DROOLS

Drools is a business rule management system with a forward-chaining and backward-chaining inference based rules engine, allowing fast and reliable evaluation of business rules and complex event processing.

[Read more →](#)

OPTAPLANNER

OptaPlanner is a constraint solver that optimizes use cases such as employee rostering, vehicle routing, task assignment and cloud optimization.

[Read more →](#)

JBPM

jBPM is a flexible Business Process Management suite allowing you to model your business goals by describing the steps that need to be executed to achieve those goals.

[Read more →](#)

APPFORMER

AppFormer is a low code platform to develop modern applications. It's a powerful tool for developers that can easily build applications by mashing up components and connect them to other Red Hat modules and software.

We make building apps looks easy.

[Read more →](#)

Above: OptaPlanner Installation
<https://www.optaplanner.org/>

ECLIPSE FOUNDATION

Members Working Groups Projects More [Download](#)

Home / Downloads / Packages / Release / Eclipse 2018-09 / R

Eclipse Installer Eclipse Packages Eclipse Developer Builds ▾

Eclipse 2018-09 R Packages

 JRebel for Eclipse IDE See Java Code Changes Instantly. Save Time. Reduce Stress. Finish Projects Faster!	 	 Payara Server 5 Innovative, cloud-native & optimized for production deployments Download for Free
 Eclipse IDE for Eclipse Committers 311 MB 177,028 DOWNLOADS Package suited for development of Eclipse itself at Eclipse.org; based on the Eclipse Platform adding PDE, Git, Marketplace Client, source code and developer documentation. Click here to file a bug against Eclipse Platform. Click here to file a bug against Eclipse Git team provider.		Windows 32-bit 64-bit Mac Cocoa 64-bit Linux 32-bit 64-bit
 Eclipse IDE for C/C++ Developers 223 MB 125,268 DOWNLOADS An IDE for C/C++ developers with Mylyn integration.		Windows 32-bit 64-bit Mac Cocoa 64-bit Linux 32-bit 64-bit
 Eclipse IDE for Java and DSL Developers 338 MB 88,054 DOWNLOADS The essential tools for Java and DSL developers, including a Java & Xtend IDE, a DSL Framework (Xtext), a Git client, XML Editor, and Maven integration.		Windows 32-bit 64-bit Mac Cocoa 64-bit Linux 32-bit 64-bit
 Eclipse IDE for Java Developers 189 MB 352,300 DOWNLOADS The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven and Gradle integration.		Windows 32-bit 64-bit Mac Cocoa 64-bit Linux 32-bit 64-bit
 Eclipse IDE for JavaScript and Web Developers 172 MB 20,511 DOWNLOADS The essential tools for any JavaScript developer, including JavaScript, HTML, CSS, XML languages support, Git client, and Mylyn.		Windows 32-bit 64-bit Mac Cocoa 64-bit Linux 32-bit 64-bit
 Eclipse IDE for Java EE Developers 339 MB 374,239 DOWNLOADS Tools for Java developers creating Java EE and Web applications, including a Java IDE, tools for Java EE, JPA, JSF, Mylyn, EGit and others.		Windows 32-bit 64-bit Mac Cocoa 64-bit Linux 32-bit 64-bit
 Eclipse Modeling Tools 437 MB 8,208 DOWNLOADS		

Get Eclipse IDE 2018-09
Install your favorite desktop IDE packages.

[Download 32 bit](#) [Download 64 bit](#)

[Download Packages](#) | [Need Help?](#)

RELATED LINKS

- Compare & Combine Packages
- New and Noteworthy
- Install Guide
- Documentation
- Updating Eclipse
- Forums

MORE DOWNLOADS

- Other builds
- Eclipse 2018-09 (4.9)
- Eclipse Photon (4.8)
- Eclipse Oxygen (4.7)
- Eclipse Neon (4.6)
- Eclipse Mars (4.5)
- Eclipse Luna (4.4)
- Eclipse Kepler (4.3)
- Older Versions

HINT
You will need a Java runtime environment (JRE) to use Eclipse

Above: Install Eclipse IDE
<https://www.eclipse.org/downloads/packages/>

stack overflow Search... 

Home PUBLIC Stack Overflow Tags Users Jobs TEAMS + Create Team

509 (Edit 2016-10-12: Many Eclipse downloads from <https://eclipse.org/downloads/eclipse-packages/> have M2Eclipse included already. As of Neon both the Java and the Java EE packages do - look for "Maven support")

Maven Eclipse plugin installation step by step:

1. Open Eclipse IDE
2. Click Help -> Install New Software...
3. Click Add button at top right corner
4. At pop up: fill up Name as "M2Eclipse" and Location as "<http://download.eclipse.org/technology/m2e/releases>" or <http://download.eclipse.org/technology/m2e/milestones/1.0>
5. Now click OK

After that installation would be started.

Another way to install Maven plug-in for Eclipse:

1. Open Eclipse
2. Go to Help -> Eclipse Marketplace
3. Search by Maven
4. Click "Install" button at "Maven Integration for Eclipse" section
5. Follow the instruction step by step

After successful installation do the followings in Eclipse:

1. Go to Window -> Preferences
2. Observe, Maven is enlisted at left panel

Finally,

1. Click on an existing project
2. Select Configure -> Convert to Maven Project

share improve this answer edited Oct 12 '16 at 9:05 Thorbjørn Ravn Andersen 56.8k ● 23 ● 142 ● 286 answered Nov 30 '12 at 6:26 Ripon Al Wasim 25.3k ● 30 ● 123 ● 149

4 I have tired these steps but not install, I got some error which detail below.... Cannot complete the install because one or more required items could not be found. Software being installed: m2e - Maven Integration for Eclipse (includes Incubating components) 1.5.0.20140606-0033 (org.eclipse.m2e.feature.feature.group 1.5.0.20140606-0033) – Krunal Patel Oct 6 '14 at 6:31 @Krunal Patel: have you tried from Marketplace? – Ripon Al Wasim Oct 13 '14 at 6:21 I have the same Cannot complete the install because one or more required items could not be found. Software being installed: m2e - slf4j over logback logging (Optional) 1.5.0.20140606-0033 (org.eclipse.m2e.logback.feature.feature.group 1.5.0.20140606-0033) - didn't see the marketplace option under help. – Diego Oct 21 '14 at 0:44 4 Eclipse juno doesnt support maven 1.5..still they have not added capability of 1.5 so try 1.4 or 1.3 download.eclipse.org/technology/m2e/releases/1.3 download.eclipse.org/technology/m2e/releases/1.4 – Alvin Mar 3 '15 at 6:47 Thanks for such a great description but I am not able to install maven either way. Error saying that "Cannot satisfy dependency:" and "annot complete the install because one or more required items could not be found.". I am not sure what's the issue. Please help. – Kushal Jayswal Nov 19 '15 at 12:19 show 4 more comments

2 Exception in java.lang.N
1 How do I import Java file if I my project Path?
4 java applic eclipse
0 How To Cre
1 creating a s in eclipse
1 No suitable driver
1 How to use with an Ecli dependenc
see more linked que

Related

1197 How can yo
1152 Must Over after import
1788 *Debug cer Android plu
582 Force mave
849 How to add project?
794 How to solv by lifecycle Maven Buil
17 Installing AI
9 Maven plug
8 How do yo Eclipse?
1 Setting up e with Maven

Hot Network Q

Coworker reviewi
Are there any oth with mental health
I'm an Australian Iran; can I go to C
Search for exact
Adding size of file
Why do the antec spawn?

Above: Install Maven in Eclipse

<https://stackoverflow.com/questions/8620127/maven-in-eclipse-step-by-step-installation>

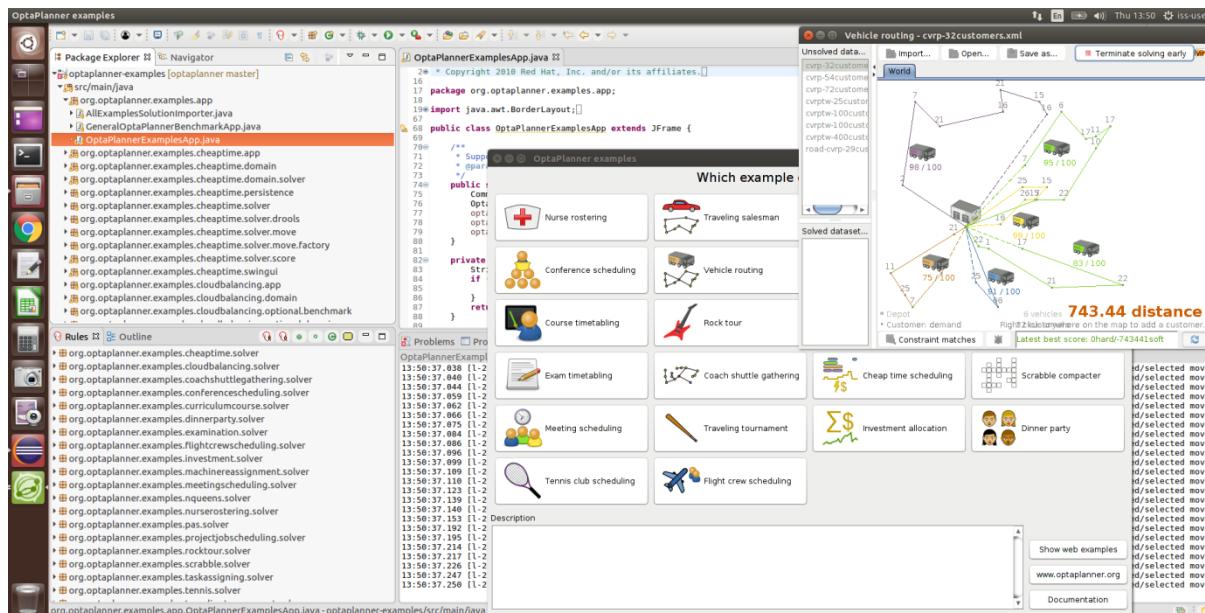
The screenshot shows the official Drools download page. It features a navigation bar with links for Home, Download, Learn, Get Help, Source, Services, and KIE. Below the navigation is a note about the license: "License: ASL 2.0". The main content is a table with columns for Name, Description, and Download. The rows include:

- Drools Engine**: Described as the rule engine and Drools Fusion does complex event processing (CEP). Distribution zip contains binaries, examples, sources and javadocs. Download links: Distribution ZIP.
- Drools and jBPM integration**: Described as Drools and jBPM integration with third party project like Spring. Distribution zip contains binaries, examples and sources. Download link: Distribution ZIP.
- Drools Workbench**: Described as the web application and repository to govern Drools and jBPM assets. Documentation link: documentation. Download links: WildFly 14 WAR, EAP 7 WAR.
- Drools and jBPM tools**: Described as Eclipse plugins and support for Drools, jBPM and Guvnor functionality. Distribution zip contains binaries and sources. Download link: Distribution ZIP.
- KIE Execution Server**: Described as Standalone execution server that can be used to remotely execute rules using REST, JMS or Java interface. Distribution zip contains WAR files for all supported containers. Download link: Distribution ZIP.

Above: Install KIE plug-in for Eclipse
<https://www.drools.org/download/download.html>

The screenshot shows the GitHub repository page for `kiegroup / optaplanner`. The top navigation bar includes links for Why GitHub?, Business, Explore, Marketplace, Pricing, and Sign in/Sign up. The repository header shows 154 stars and 473 forks. The main content area displays the repository structure under `optaplanner / optaplanner-examples /`, including files like `data`, `src`, `.gitignore`, and `pom.xml`, along with their commit history. At the bottom, there are links for Contact GitHub, Pricing, API, Training, Blog, and About.

Above: Download OptaPlanner source code and example cases
<https://github.com/kiegroup/optaplanner>



Above: Import OptaPlanner example cases as Maven project in Eclipse

[OptaPlanner](http://www.optaplanner.org) Home Download Learn Get Help Team Services [Follow](#) [GitHub](#)

What is OptaPlanner?

OptaPlanner is a **constraint solver**. It optimizes business resource planning use cases, such as [Vehicle Routing](#), [Employee Rostering](#), [Cloud Optimization](#), [Task Assignment](#), [Conference Scheduling](#), [Job Scheduling](#), [Bin Packing](#) and many more. Every organization faces such scheduling puzzles: assign a limited set of constrained resources (employees, assets, time and money) to provide products or services. OptaPlanner delivers more efficient plans to improve service quality and reduce costs.

OptaPlanner is a **lightweight, embeddable planning engine**. It enables normal Java™ programmers to solve optimization problems efficiently. It is also compatible with other JVM languages (such as Kotlin and Scala). Constraints apply on plain domain objects and can reuse existing code. There's no need to input them as 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.

[Download OptaPlanner 7.14.0.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`

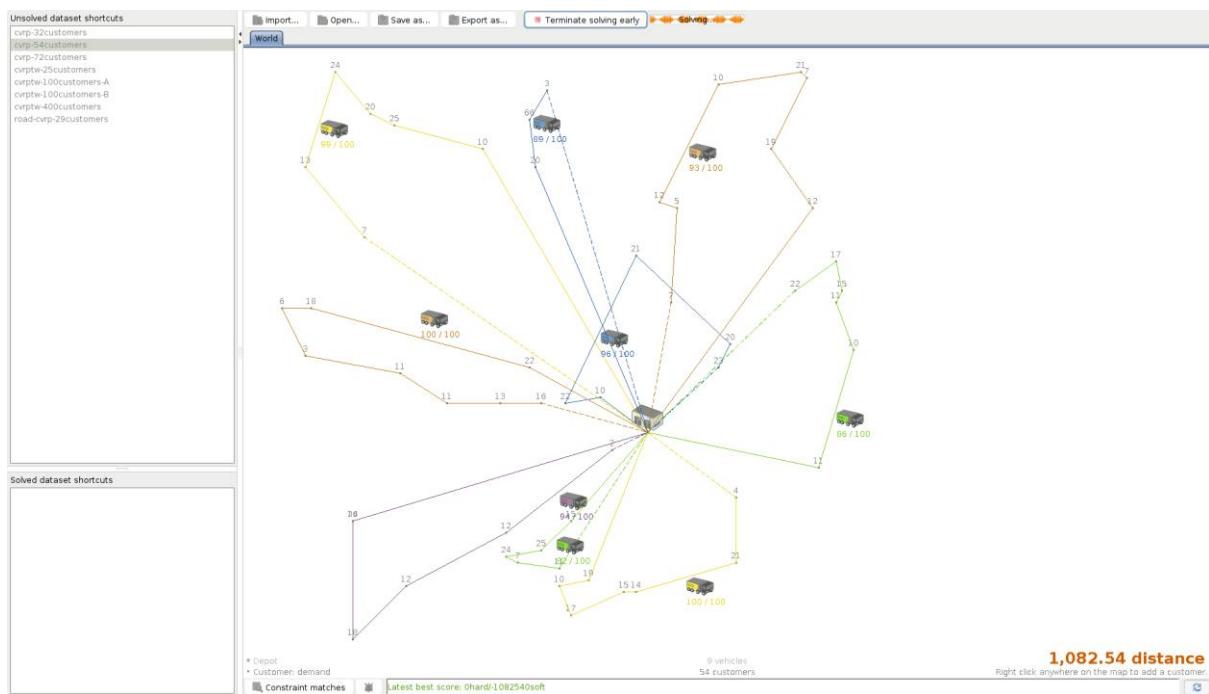
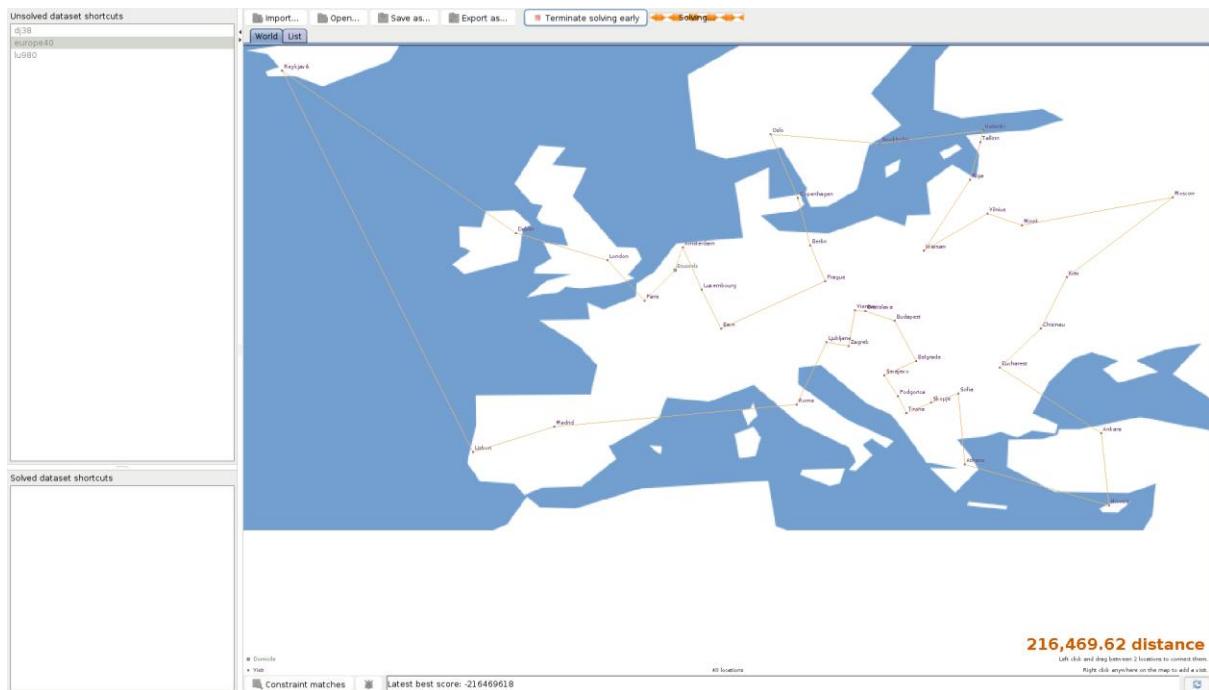
Requires Java™ to run.

[Read documentation 7.14.0.Final](#)

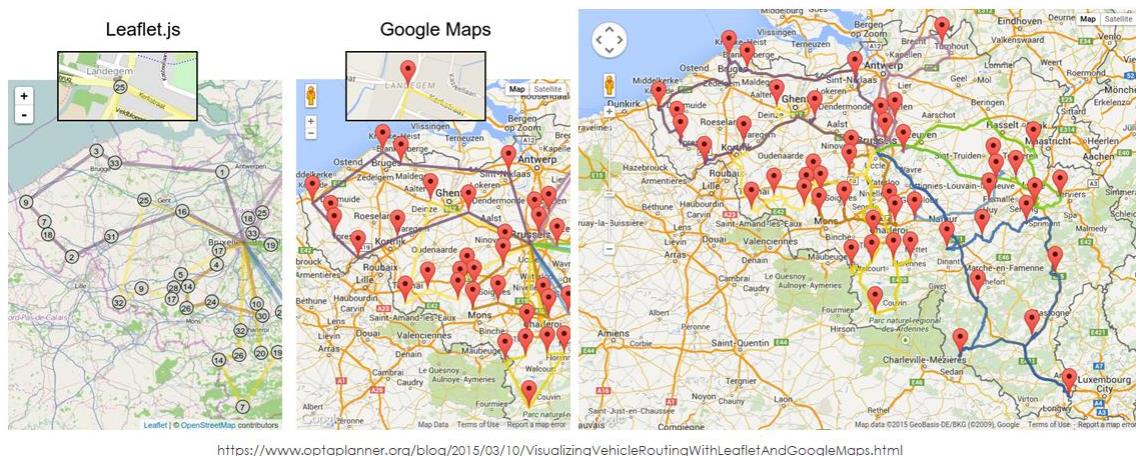
Read the [Quick Start](#) chapter.

Video Use cases Usage Integration Compatibility Scalability

Above: Standalone OptaPlanner
[https://www.optaplanner.org/](http://www.optaplanner.org/)



Visualizing Vehicle Routing with Leaflet and Google Maps



VRP Reference

- <http://www.optaplanner.org/learn/useCases/vehicleRoutingProblem.html>
- <http://www.optaplanner.org/learn/slides/optaplanner-presentation/index.html#/2>
- <https://www.optaplanner.org/blog/2015/03/10/VisualizingVehicleRoutingWithLeafletAndGoogleMaps.html>

ANNEX 1 WORKSHOP PROJECT CANDIDATE

Workshop Project Candidate One

Hybrid Airport Gate Assignment System (HAGAS)

The Airport Gate Assignment Problem: Scheduling Algorithms and Simulation Approach
Ahmed Thanyan AL-Sultan

The rapid development of airlines has made airports busier and more complicated. The assignment of schedule to available gates is a major issue for daily airline operations. We consider the over-constrained airport gate assignment problem (AGAP) where the number of flights exceeds the number of available gates, and where the objectives are to minimize the number of ungated flights and the total walking distance or connection times. The procedures used in this project are to create a mathematical model formulation to identify decision variables to identify, constraints and objective functions. In addition, we will consider in the AGAP the size of each gate in the terminal and also the towing process for the aircraft. We will use a greedy algorithm and a Tabu search meta-heuristic to solve the problem and compare it with other scheduling methods. Actual and forecasted data will be simulated in the experiment. The greedy algorithm minimizes ungated flights while providing initial feasible solutions that allow flexibility in seeking good solutions, especially in case when flight schedules are dense in time. Experiments conducts give good results. The distance a passenger has to walk in any airport to reach various key areas, including departure gates, baggage belts and connecting flights provide for an important performance measure for the quality of any airport. While certain walking distances are fixed, others are dynamic. In particular, the distances traversed by passengers from check-in counters to gates and from gate to gate, in the case of transfer or connecting passengers, change according to how scheduled flights are assigned to gates. This allows for the ground handling agents and airlines, together with airport authorities, to dynamically assign airport gates to scheduled flights so as to minimize walking distances while, consequently, minimizing connection times. Which flight to gate assignment policy to be used so as to achieve such minimum times can be derived at the start of such planning day based on published flights schedules and booked passenger loads. The airport gate assignment problem (AGAP) seeks to find feasible flight to gate assignments so that total passenger connection times and walking distances is minimized. Distances that are taken into account are those from check-in to gates in the case of embarking or originating passengers, from gates to baggage claim areas (check-out) in the case of disembarking or destination passengers and from gate to gate in the case of transfer or connecting passengers. In the over-constrained case, where the number of aircraft exceeds the number of available gates, we include the distance from the apron or tarmac area to the terminal for aircraft assigned to these areas.

...

Reference

The Airport Gate Assignment Problem: Scheduling Algorithms and Simulation Approach,
Ahmed Thanyan AL-Sultan, Graduate School of environmental science, March 2012
http://ousar.lib.okayama-u.ac.jp/files/public/4/48534/20160528091554614463/K0004584_honbun.pdf

Workshop Project Candidate Two

OptaPlanner Application Implementation in KIE Workbench & Server

Objective:

Construct a useful business reasoning system using KIE product suite, incorporating an OptaPlanner solver as an embedded optimization engine/task for automated machine reasoning.

Choose one OptaPlanner example application from below **ONLY**:

- Cloud balancing
- Course timetabling
- Vehicle routing with time windows
- Project job scheduling
- Exam timetabling
- Nurse rostering
- Cheap time scheduling
- Flight crew scheduling

System Requirements:

- Make use of KIE BRMS (Drools) & BPMS (jBPM) capability;
- Define at least one practical business enhancement/requirement based on OptaPlanner example;
- Convert/migrate original Maven/Eclipse OptaPlanner project with enhancements into KIE (jBPM) Workbench project;
- Develop a (web-based) User Interface;
- Use Restful API between User Interface and KIE Server;
- Deploy the developed system onto KIE Server for use;

ANNEX 2 PROJECT CODE EXPORT & IMPORT USING KIE WORKBENCH

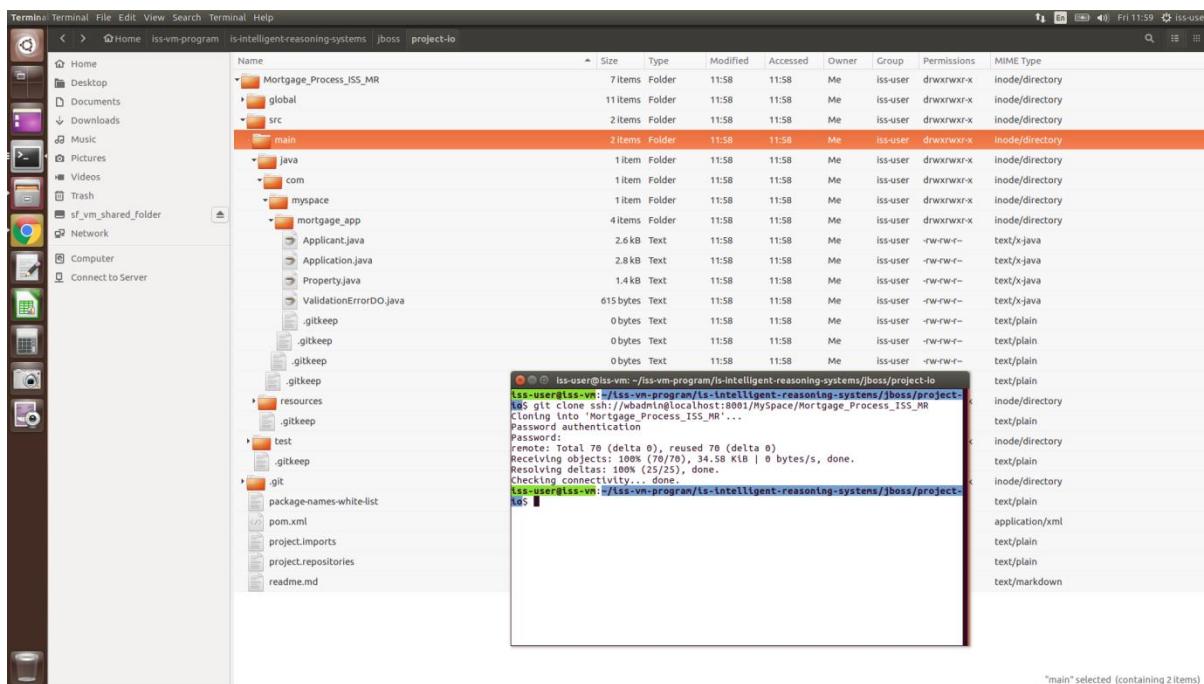
Example: export KIE project **Mortgage_Process_ISS_MR** from work space **MySpace**

Review project settings to obtain project URL

The screenshot shows the KIE Workbench interface with the 'Settings' tab selected for the 'Mortgage_Process_ISS_MR' project. The left sidebar lists project categories: General Settings, Dependencies, KIE bases, External Data Objects, Validation, Deployments, and Persistence. The 'General Settings' section contains fields for Name (Mortgage_Process_ISS_MR), Description (Getting started loan approval process in BPMN2, decision table, business rules, and forms.), URL (ssh://localhost:8001/MySpace/Mortgage_Process_ISS_MR), and Group ID (mortgage-process). There are also checkboxes for Disable GAV conflict check and Allow child GAV edition. The 'Artifact ID' field is set to Mortgage_Process_ISS_MR and the 'Version' field is set to 1.0.0-SNAPSHOT.

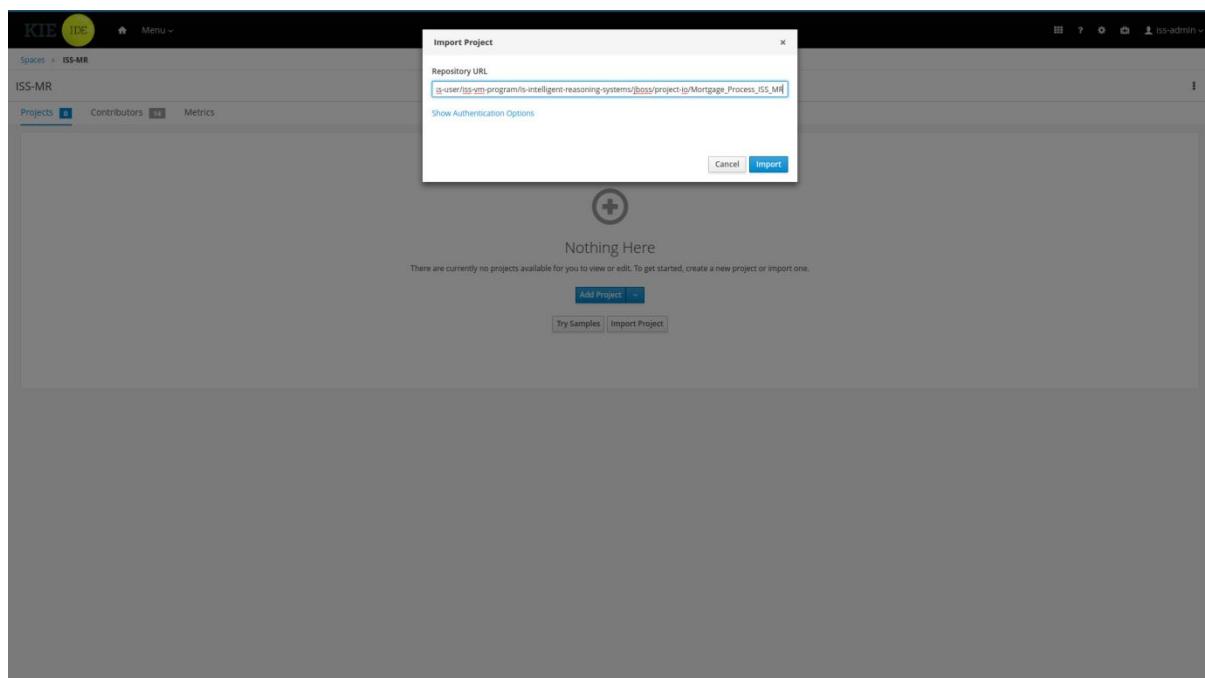
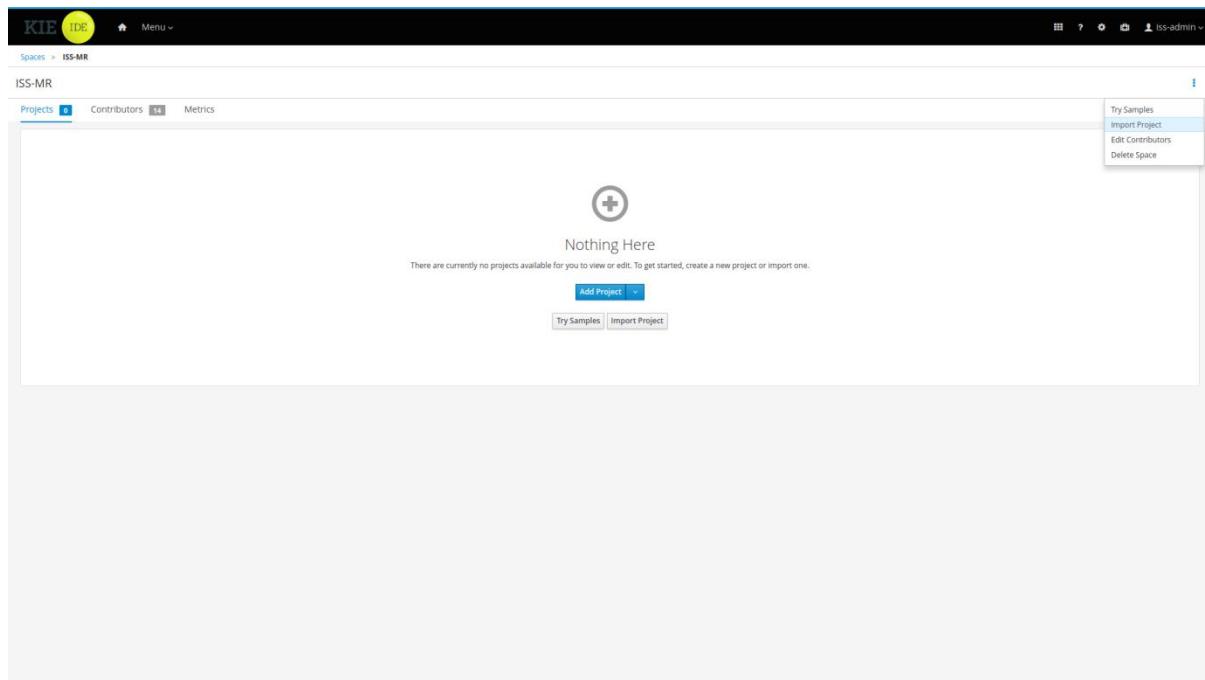
Export project from KIE Workbench

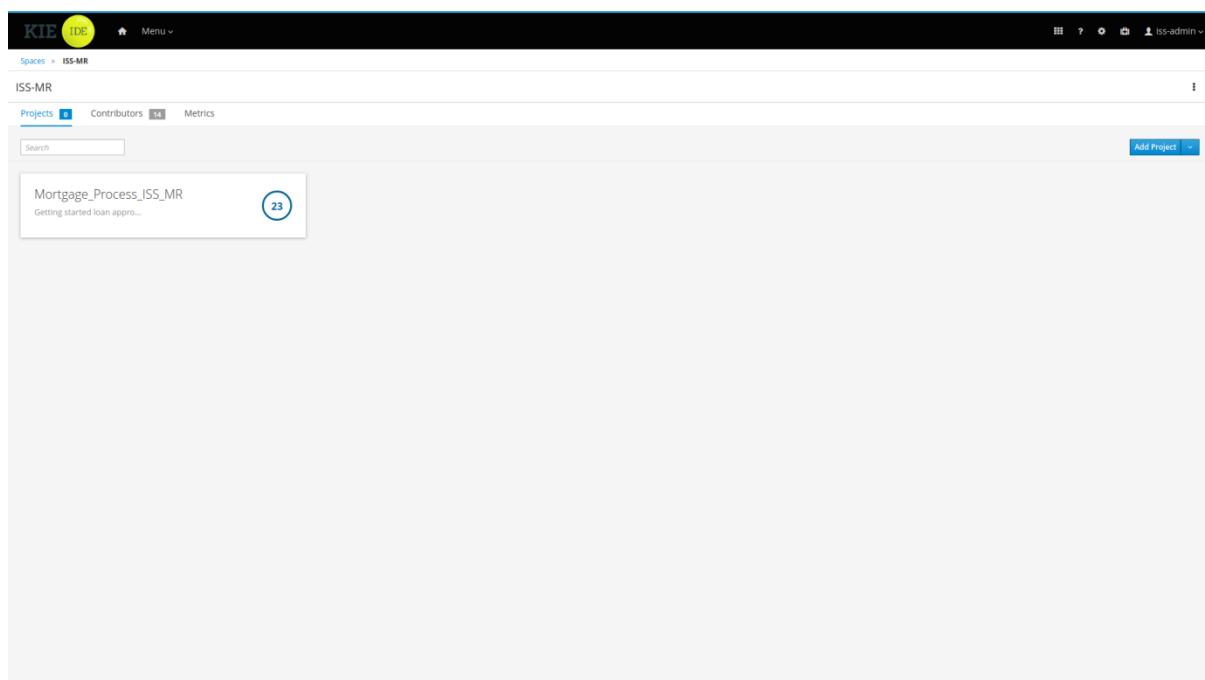
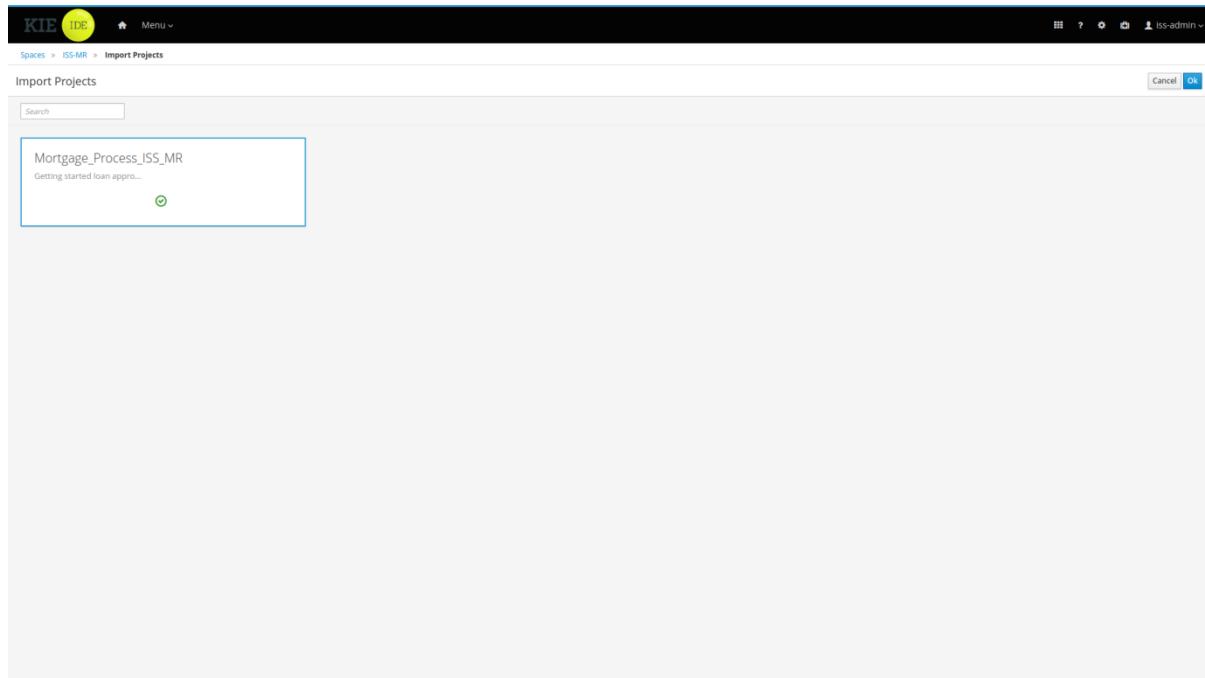
1. Select a folder for exporting, example here uses **/home/iss-user/iss-vm-program/is-intelligent-reasoning-systems/jboss/project-io**
2. Start a Terminal there, key in command **git clone ssh://wbadmin@localhost:8001/MySpace/Mortgage_Process_ISS_MR**
3. Key in password ‘**wbadmin**’ for user wbadmin

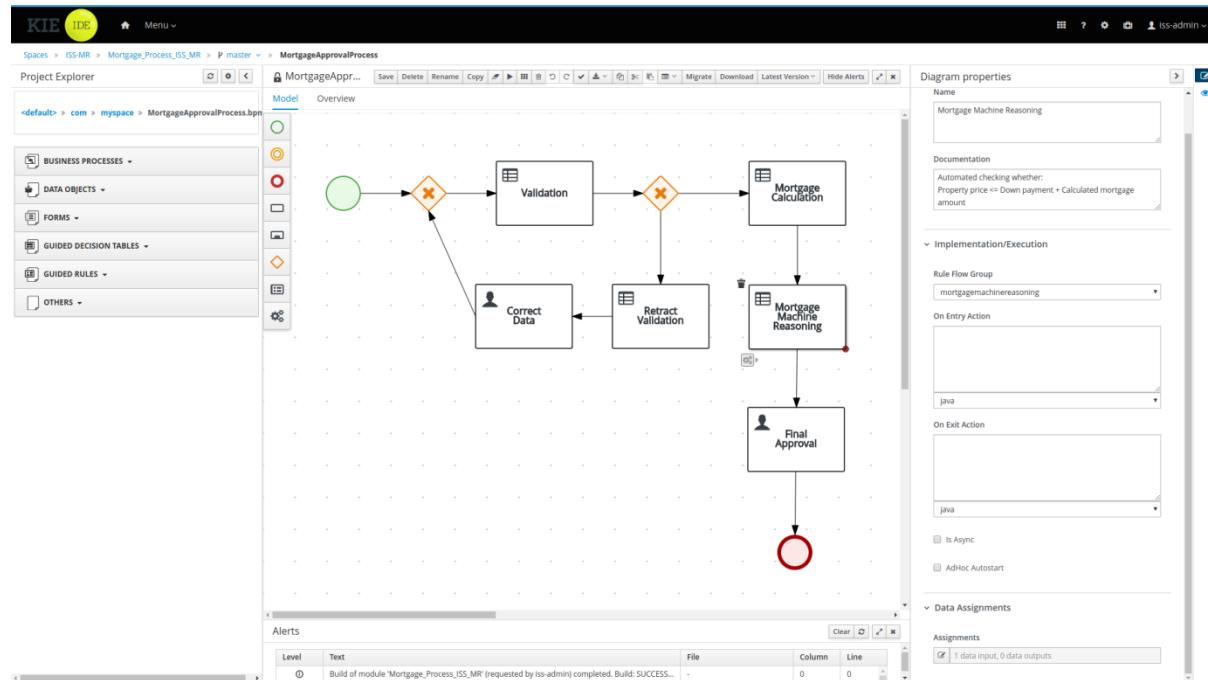


Import project into KIE Workbench

1. In KIE workbench, select/create a project Space, example here uses **ISS-MR**
2. Click menu function '**Import Project**'
3. For Repository URL, key in `file:///home/iss-user/iss-vm-program/is-intelligent-reasoning-systems/jboss/project-io/Mortgage_Process_ISS_MR`







Reference

- <https://developer.jboss.org/thread/269991>
- <https://developer.jboss.org/thread/237411>
- <https://developer.jboss.org/thread/252588>

ANNEX 3 WORKSHOP PROJECT SUBMISSION

**Submission due by 23:59 on last lecture date (+ 14)
One delayed day = 10 marks deduction**

1. [MTech & EEP] Create Github repository for project submission
2. [MTech] Download Github repository as a ZIP file, then upload to NUS LumiNUS / IVLE

Reference <https://github.com/IRS-PM/Workshop-Project-Submission-Template>

The screenshot shows a GitHub repository page for 'Workshop-Project-Submission-Template'. The repository was forked from 'telescopeuser/Workshop-Project-Submission-Template'. It has 11 commits, 1 branch, 0 releases, and 1 contributor. The contributor is Gu Zhan, who updated README. The repository contains files like Miscellaneous, ProjectReport, SystemCode/clips, UserGuide, and README.md. A prominent button at the top right says 'Clone or download' with a 'Download ZIP' option highlighted. The page also includes a section for 'Workshop Project Submission Template: Github Repository & Zip File' and a note about naming convention.

ANNEX 4 KIE OptaPlanner Examples

Which example do you want to see?

 Nurse rostering	 Traveling salesman	 Task assigning	 Cloud balancing
 Conference scheduling	 Vehicle routing	 Hospital bed planning	 Machine reassignment
 Course timetabling	 Rock tour	 Project job scheduling	 N queens
 Exam timetabling	 Coach shuttle gathering	 Cheap time scheduling	 Scrabble compacter
 Meeting scheduling	 Traveling tournament	 Investment allocation	 Dinner party
 Tennis club scheduling	 Flight crew scheduling		

Description

Assign processes to computers.
Each computer must have enough hardware to run all of its processes.
Each used computer incurs a maintenance cost.

[Show web examples](#)
[www.optaplanner.org](#)
[Documentation](#)

Reference

- https://docs.optaplanner.org/latest/optaplanner-docs/html_single/index.html#useCasesAndExamples

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Dinner Party



- **Business Scenario / Problem Description**
- **Miss Manners is throwing another dinner party.**

- This time she invited 144 guests and prepared 12 round tables with 12 seats each.
- Every guest should sit next to someone (left and right) of the opposite gender.
- And that neighbour should have at least one hobby in common with the guest.
- At every table, there should be two politicians, two doctors, two socialites, two coaches, two teachers and two programmers.
- And the two politicians, two doctors, two coaches and two programmers should not be the same kind at a table.

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram



Unsolved dataset shortcuts
wedding01

Solved dataset shortcuts
wedding01-score20

Import... Open... Save as... Export as... Solve

Table 0

Zachary Developer C	Sophia Politician Democrat	Charlie Developer Java	Avery Doctor Surgeon
Leah Teacher English			Mackenzie Coach Football
Matthew Doctor Orthopath			Emily Politician Republican
Alyssa Socialite Democrat	Henry Coach Soccer	Bailey Socialite Democrat	Owen Teacher Math

Table 1

Isabelle Coach Football	Emma Politician Democrat	Eli Teacher Math	Madelyn Doctor Pediatrician
Taylor Socialite Republican			Joseph Developer Cognitiv
Evan Developer Java			Zoe Politician Republican
Riley Coach Basketball	Christian Teacher History	Stella Socialite Republican	Connor Doctor Orthopath

Table 2

Andrew Teacher Math	Ava Politician Democrat		
Kennedy Socialite Republican			
Evelyn Coach Football			
Penelope Socialite Republican	Isaac Coach Baseball		

Table 4

Caleb Doctor Surgeon	Samantha Socialite Republican	Brooklyn Coach Football	Mia Politician Democrat
Lajla Politician Republican			Nathan Teacher Math
Max Teacher English			Ruby Socialite Democrat
Gabriella Developer Java	James Doctor Orthopath	Alaina Developer C	Joshua Coach Baseball

Table 5

Ryan Doctor Surgeon	Ella Doctor Pediatrician	Grayson Teacher Math	Madison Politician Republican
Elena Developer Java			Gabriel Coach Baseball
Colton Developer C			Sadie Socialite Democrat
Elaina Socialite Democrat	Dylan Coach Soccer	Olivia Politician Democrat	Wyatt Teacher Science

Table 6

Nicholas Coach Baseball	Maria Developer Perl		
Annabelle Developer Java			
Jackson Politician Democrat			
Camila Socialite Democrat	Ula Coach Football		

Table 8

Constraint matches Latest best score: 20

Table 9

Table 10

The screenshot displays the Eclipse IDE interface with the following details:

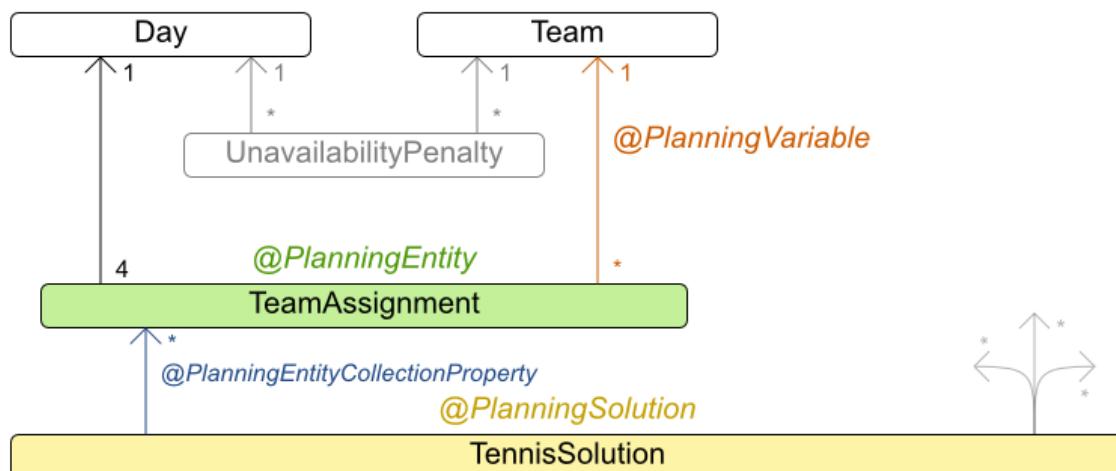
- Package Explorer View:** Shows the project structure with packages like org.optaplanner.examples.conferencescheduling.swingui, org.optaplanner.examples.curriculumcourse.app, org.optaplanner.examples.curriculumcourse.domain, org.optaplanner.examples.curriculumcourse.domain.solver, org.optaplanner.examples.curriculumcourse.persistence, org.optaplanner.examples.curriculumcourse.solver.move, org.optaplanner.examples.curriculumcourse.swingui, org.optaplanner.examples.dinnerparty.app, and org.optaplanner.examples.dinnerparty.domain.
- Outline View:** Shows the class structure of `SeatDesignation`. It includes:
 - Fields: `guest`, `seat`.
 - Methods: `getGuest()`, `getSeat()`, `setSeat(Seat)`, `void`, `getGuestName():String`, `getGuestGender():Gender`, `getGuestJob():Job`, `getGuestJobType():JobType`, `differentKindNeeded(Job otherGuestJob):boolean`.
 - Nested Classes: `Guest`, `Job`, `JobType`.
- Editor View:** Displays the code for `SeatDesignation.java`. The code defines the `SeatDesignation` class as an abstract persistable implementation of `Labeled`. It has fields for `guest` and `seat`, and methods for setting them and getting their names, genders, jobs, and job types. It also includes logic for determining if two guests have the same job type and if a guest is a neighbor of another guest.

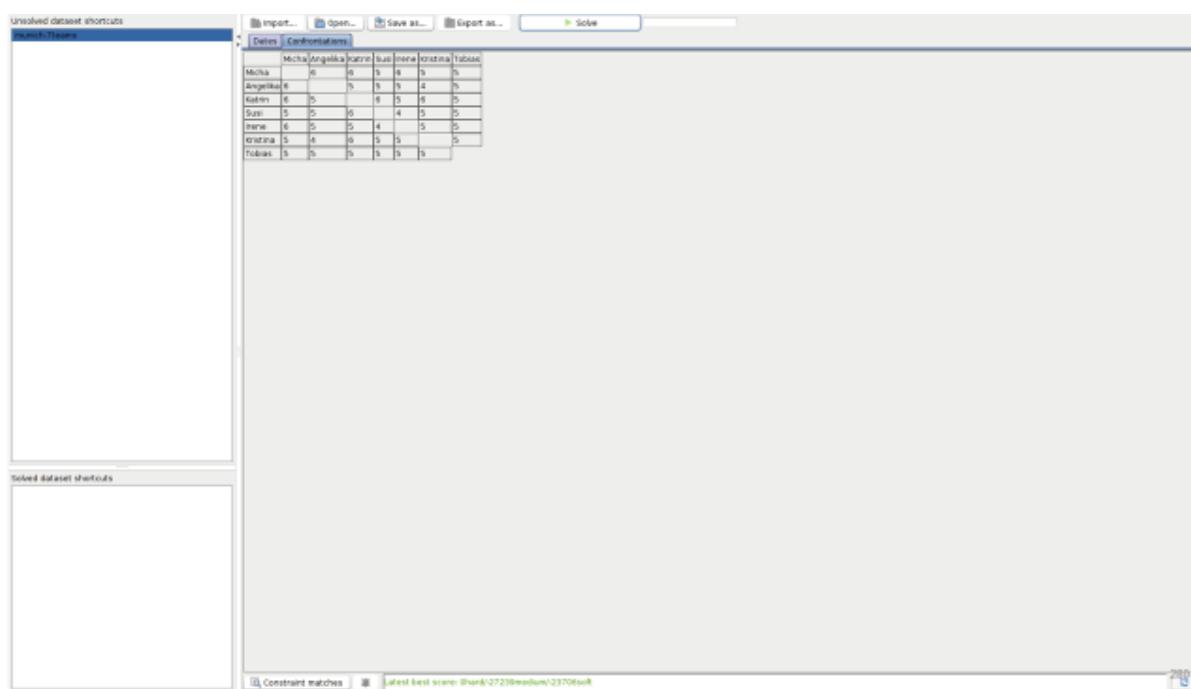
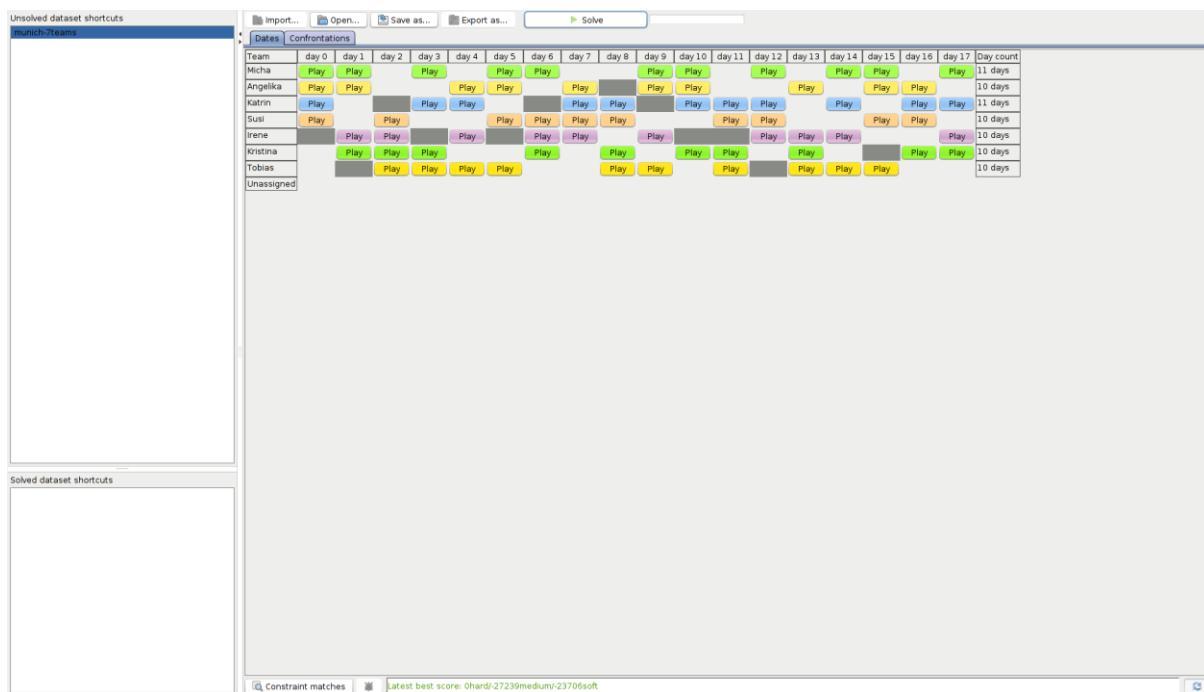
ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Tennis Club Scheduling

- Business Scenario / Problem Description
- Every week the tennis club has four teams playing round robin against each other. Assign those four spots to the teams fairly.
- Hard constraints:
 - Conflict: A team can only play once per day.
 - Unavailability: Some teams are unavailable on some dates.
- Medium constraints:
 - Fair assignment: All teams should play an (almost) equal number of times.
- Soft constraints:
 - Evenly confrontation: Each team should play against every other team an equal number of times.

Tennis class diagram





ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Meeting Scheduling

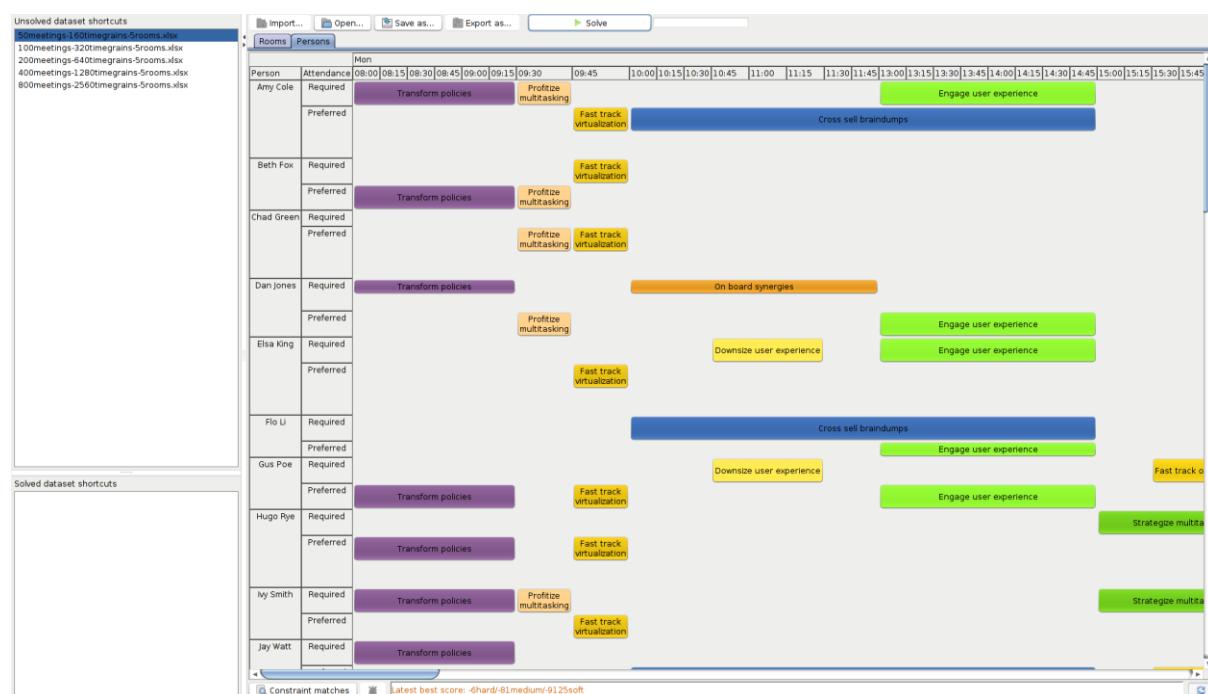
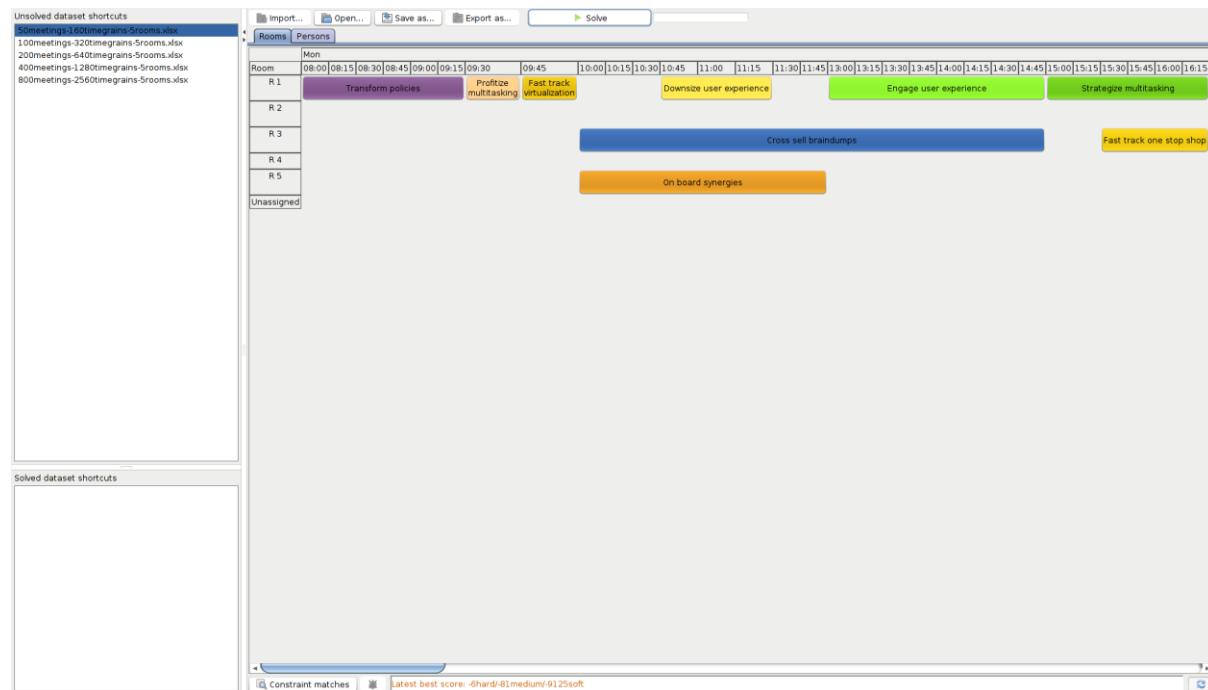
- Business Scenario / Problem Description
- Assign each meeting to a starting time and a room. Meetings have different durations.
- Hard constraints:
 - Room conflict: two meetings must not use the same room at the same time.
 - Required attendance: A person cannot have two required meetings at the same time.
 - Required room capacity: A meeting must not be in a room that doesn't fit all of the meeting's attendees.
 - Start and end on same day: A meeting shouldn't be scheduled over multiple days.
- Medium constraints:
 - Preferred attendance: A person cannot have two preferred meetings at the same time, nor a preferred and a required meeting at the same time.
- Soft constraints:
 - Sooner rather than later: Schedule all meetings as soon as possible.
 - A break between meetings: Any two meetings should have at least one time grain break between them.
 - Overlapping meetings: To minimize the number of meetings in parallel so people don't have to choose one meeting over the other.
 - Assign larger rooms first: If a larger room is available any meeting should be assigned to that room in order to accommodate as many people as possible even if they haven't signed up to that meeting.
 - Room stability: If a person has two consecutive meetings with two or less time grains break between them they better be in the same room.

© 2018 National University of Singapore. All Rights Reserved

286

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram



ANNEX 4 : KIE OPTAPLANNER EXAMPLES

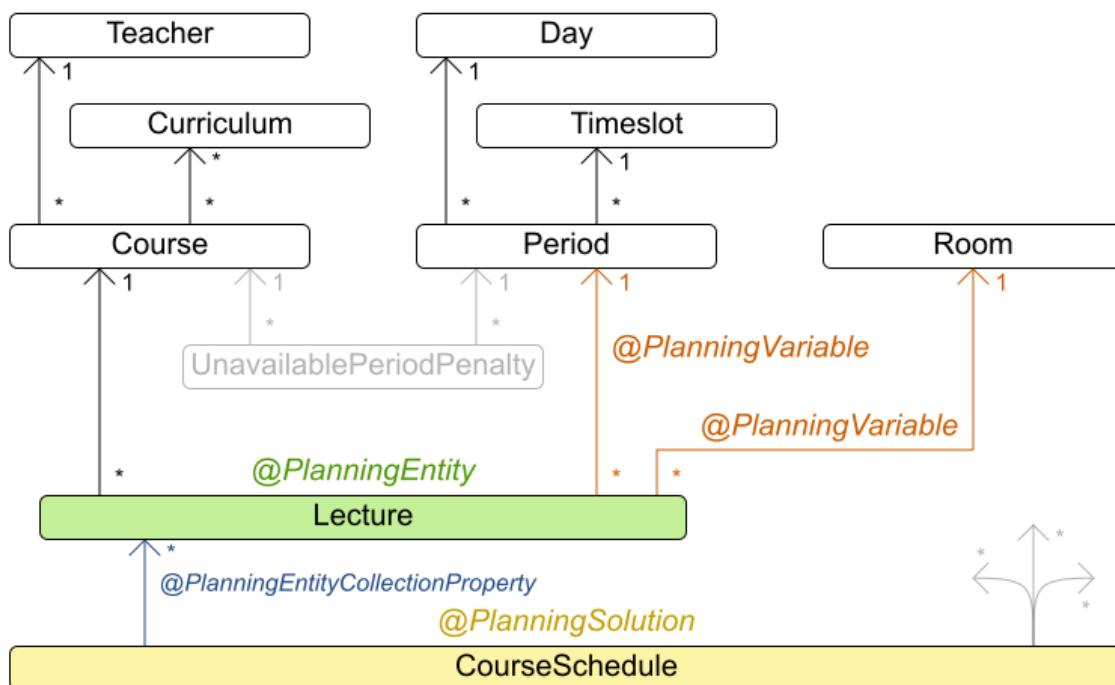
KIE OptaPlanner Deep Dive – Curriculum Course Scheduling

- Business Scenario / Problem Description
 - Schedule each lecture into a timeslot and into a room.
 - Hard constraints:
 - Teacher conflict: A teacher must not have two lectures in the same period.
 - Curriculum conflict: A curriculum must not have two lectures in the same period.
 - Room occupancy: two lectures must not be in the same room in the same period.
 - Unavailable period (specified per dataset): A specific lecture must not be assigned to a specific period.
 - Soft constraints:
 - Room capacity: A room's capacity should not be less than the number of students in its lecture.
 - Minimum working days: Lectures of the same course should be spread out into a minimum number of days.
 - Curriculum compactness: Lectures belonging to the same curriculum should be adjacent to each other (so in consecutive periods).
 - Room stability: Lectures of the same course should be assigned to the same room.
 - The problem is defined by [the International Timetabling Competition 2007 track 3](#).
- http://www.cs.qub.ac.uk/itc2007/curriculumcourse/course_curriculum_index.htm

© 2018 National University of Singapore. All Rights Reserved

234

Curriculum course class diagram



Unsolved dataset shortcuts

- 200lectures-12periods-12rooms
- 400lectures-32periods-25rooms
- 800lectures-32periods-50rooms
- compl1
- compl1_initialized
- compl2
- compl3
- compl4
- compl5
- compl6
- compl7
- compl8
- compl9
- compl10
- compl11
- compl12
- compl13
- compl14
- toy01

Solved dataset shortcuts

Rooms Teachers Curricula

Day	Time	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	Unassigned
Mo	09:00	GermanB-2		MathB-0		ICTB-0	BiologyA-0		Geograph...		HistoryA-2			
	10:00		Geograph...			ArtB-0	MusicC-3		FrenchD-0					
	11:00					MusicC-2			Psycholog...					
	13:00					MusicC-1								
	14:00		MusicA-4			MusicC-0			MusicB-3					
	15:00	Geograph...	ArtB-1	HistoryC-1										
Tu	08:00	GermanB-3	Geograph...						ICTA-1	ArtA-3				
	09:00	GermanB-4				PhysicsC-3	Chemistry...	FrenchC-0			HistoryA-1	Economic...		
	10:00		Geograph...	germanB1				GermanA-0		FrenchE-3		FrenchE-0	PhysicsC5	
	11:00					PhysicsC-0	SpannishA-2		Chemistry...		HistoryA-3	Economic...		
	13:00	Geograph...				PhysicsC-0	SpannishA-2		FrenchD-1		PhysicsB-3	Psycholog...		
	14:00	Geograph...				PhysicsC-0	SpannishA-0		FrenchD-0		MusicB-0	HistoryA-0		
	15:00					MathB-2			Economic...		FrenchD-4	Chemistry...		
We	08:00					PhysicsA-4	ICTB-1				MusicB-4	Chemistry...		
	09:00	GermanB-5	Geograph...			PhysicsA-3		HistoryB-5		MusicB-2	Chemistry...	MathB-2		
	10:00	MusicA-3				PhysicsA-2	HistoryB-1	FrenchC-4		MusicB-3	Chemistry...	MathB-3		
	11:00					PhysicsA-1	HistoryB-3	FrenchC-3		MusicB-0	HistoryD-0		HistoryB-5	
	13:00					PhysicsA-0								Geograph...
	14:00													MathA-2
	15:00													FrenchB-0
Th	08:00								Spannish-4		Geograph...			
	09:00						Spannish-3				Psycholog...	Psycholog...		
	10:00							ICTB-3		Geograph...				
	11:00							HistoryD-1		MusicA-2		Psycholog...		
	13:00							Psycholog...		Chemistry...		Psycholog...		
	14:00							PhysicsA-3		MusicB-1		Chemistry...		
	15:00							PhysicsA-0		MusicB-0		MathA-3		
Fr	08:00								MathB-1	GermanA-1	Economic...			
	09:00	Spannish-1	GermanB-4					MusicC-5		FrenchD-2		FrenchE-0		
	10:00	MathA-0					HistoryC-4		Economic...		FrenchD-3	MathC-2		
	11:00	MathA-1					HistoryC-0		Economic...		MusicA-3	MathC-3		
	13:00										PhysicsB-2	ICTA-0		
	14:00										PhysicsB-1	ICTA-2		
	15:00										Chemistry...	Economic...		
											Chemistry...	MathC-5		
											Chemistry...	FrenchC-2		
											Chemistry...	EnglishB-3		
														Unassigned

Constraint matches | X latest best score: Ohard-26soft

Unsolved dataset shortcuts

- 200lectures-32periods-12rooms
- 400lectures-32periods-25rooms
- 800lectures-32periods-50rooms
- compl1
- compl1_initialized
- compl2
- compl3
- compl4
- compl5
- compl6
- compl7
- compl8
- compl9
- compl10
- compl11
- compl12
- compl13
- compl14
- toy01

Solved dataset shortcuts

Rooms Teachers Curricula

Day	Time	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	Unassigned
Mo	09:00	GermanB-2		MathB-0		ICTB-0	BiologyA-0		Geograph...		HistoryA-2			
	10:00		Geograph...			ArtB-0	MusicC-3		FrenchD-0					
	11:00					MusicC-2			Psycholog...					
	13:00					MusicC-1								
	14:00		MusicA-4			MusicC-0			MusicB-3					
	15:00	Geograph...	ArtB-1	HistoryC-1										
Tu	08:00	GermanB-3	Geograph...						ICTA-1	ArtA-1				
	09:00	GermanB-6				PhysicsC-3	Chemistry...	FrenchC-0		HistoryA-1	Economic...			
	10:00		Geograph...	GermanB1				GermanA-0		FrenchE-3		FrenchE-0	PhysicsC5	
	11:00	Geograph...	GermanB1			PhysicsC-0	SpannishA-2		Chemistry...		HistoryA-3	Economic...		
	13:00	Geograph...				PhysicsC-0	SpannishA-2		FrenchD-1		PhysicsB-3	Psycholog...		
	14:00	Geograph...				PhysicsC-0	SpannishA-0		MusicB-0		HistoryA-0	FrenchA-1		
	15:00					MathB-2			Economic...		FrenchD-4	Chemistry...		
We	08:00					PhysicsA-4	ICTB-1				MusicB-4	Chemistry...		
	09:00	GermanB-5	Geograph...			PhysicsA-3		HistoryB-5		MusicB-2	Chemistry...	MathB-2		
	10:00	GermanB-3				PhysicsA-2	HistoryB-1	FrenchC-4		MusicB-3	Chemistry...	MathB-3		
	11:00	MusicA-3				PhysicsA-1	HistoryB-3	FrenchC-3		MusicB-0	HistoryD-0		FrenchB-0	
	13:00					GermanB-0								
	14:00					Spannish-4	Spannish-3	HistoryD-1		Geograph...		Psycholog...		
	15:00					ICTB-2		HistoryD-0		MusicA-2		Psycholog...		
Th	08:00					Spannish-4	Spannish-3							
	09:00					ICTB-3								
	10:00					HistoryD-1								
	11:00					Psycholog...								
	13:00					Spannish-3								
	14:00					PhysicsA-0								
	15:00					MathB-2								
Fr	08:00					MathB-1	GermanA-1	Economic...		MusicC-5	FrenchD-2	FrenchE-0		
	09:00	Spannish-1	GermanB-4			MusicC-5		Economic...		FrenchD-3	MathC-2			
	10:00	MathA-0				HistoryC-4		Economic...		FrenchD-2	MathC-3			
	11:00	MathA-1				HistoryC-0		Economic...		FrenchD-3	MathC-4			
	13:00													
	14:00													
	15:00													
														Unassigned

Constraint matches | X latest best score: Ohard-26soft

Unsolved dataset shortcuts

- 200lectures-120periods-12rooms
- 400lectures-32periods-25rooms
- 800lectures-32periods-50rooms

Rooms Teachers Curricula

Day	Time	Group A	Group B	Group C	Group D	Group E	Group F	Group G	Group H
Mo	08:00	MathB-0		HistoryA-2	HistoryB-2	Geograph... Germane-2	MathB-0	Geograph... Germane-2	
	09:00	EnglishB-1	BiologyA-0	BiologyA-0	MusicC-4	MusicC-4	MathC-4	MathC-4	MusicC-4
	10:00	EnglishB-4	ArtB-0	FrenchD-0	MusicC-3	MusicC-3	MathC-3	MathC-3	MusicC-3
	11:00	EnglishE-5	Psycholog...	Psycholog...	MusicC-2	MusicC-2	MathC-2	MathC-2	MusicC-2
	13:00	FrenchE-1	MathD-3	MathD-3	MusicC-1	MusicC-1	MathC-3	MathC-3	MusicC-1
	14:00	BiologyC-3	MathD-0	MathD-0	MusicC-0	MusicC-0	BiologyC-3	MusicB-3	MusicC-0
	15:00	EnglishE-9	ArtB-1	HistoryC-3		SpanishB-0	MathC-0	MathC-0	
Tu	08:00	FrenchE-9	Economic...	HistoryA-1	HistoryB-1	Economic...	Germane-3	PhysicsC-5	PhysicsC-5
	09:00	EnglishB-2	MathD-4	HistoryD-4	PhysicsA-5	FinanceB-1	Germane-4	PhysicsC-4	
	10:00	FrenchE-3	Economic...	HistoryA-3	HistoryC-3	Economic...	Chemistry...	PhysicsC-3	PhysicsC-3
	11:00	Chemistry...	Psycholog...	Psycholog...	Germane-0	SpanishB-5	Germane-3	PhysicsC-2	PhysicsC-2
	13:00	FrenchA-0	SpanishA-2	FrenchD-1	PhysicsC-3	SpanishA-4	FrenchA-0	PhysicsC-1	PhysicsC-1
	14:00	FrenchA-1	SpanishA-0	HistoryA-0	HistoryB-0	Geograph...	FrenchA-1	PhysicsC-0	PhysicsC-0
	15:00	MathB-2	Economic...	FrenchD-4	HistoryD-4	Economic...	Chemistry...	MathB-2	HistoryB-4
We	08:00	PhysicsA-4	HistoryD-0	HistoryD-0	HistoryD-0	PhysicsA-4	Chemistry...	MusicB-4	HistoryB-5
	09:00	PhysicsC-3	MathD-5	HistoryD-5	HistoryB-2	PhysicsA-3	Germane-5	MusicB-0	HistoryB-2
	10:00	PhysicsA-2	MathD-2	MathD-2	HistoryD-1	PhysicsC-2	Chemistry...	MusicB-2	HistoryB-1
	11:00	PhysicsA-1	HistoryD-3	HistoryD-3	HistoryD-3	PhysicsA-1	Germane-0	FrenchD-0	HistoryB-3
Th	08:00	Psycholog...	Psycholog...	Psycholog...	SpanishB-4	Geograph...	Psycholog...	SpanishC-4	Geograph...
	09:00	BiologyC-0	HistoryD-1	HistoryD-1	SpanishC-3	Geograph...	BiologyC-0	SpanishC-3	Geograph...
	10:00	BiologyC-1	Psycholog...	Psycholog...	SpanishC-2	Geograph...	BiologyC-1	SpanishC-2	Geograph...
	11:00	Psycholog...	Psycholog...	Psycholog...	SpanishC-1	Geograph...	Psycholog...	SpanishC-1	Chemistry...
	13:00	Chemistry...	MathD-0	MathD-0	SpanishC-0	SpanishA-3	Chemistry...	SpanishC-0	
	14:00	PhysicsA-3	Psycholog...	Psycholog...	PhysicsC-0	PhysicsA-4	EnglishA-2	MusicB-3	
	15:00	EnglishB-8	HistoryD-2	HistoryD-2	PhysicsC-1	Geograph...	FrenchD-2	Chemistry...	
Fr	08:00	MathB-1	Economic...	FrenchD-2	Germane-1	Economic...	EnglishA-0	MathB-1	EnglishA-0
	09:00	Economic...	SpanishA-1	FrenchD-3	MusicC-5	MusicC-5	Germane-4	Economic...	MusicC-5
	10:00	Economic...	Economic...	HistoryC-4	PhysicsC-2	Economic...	EnglishA-1	Economic...	EnglishA-1
	11:00	BiologyC-2	Economic...	HistoryC-0	HistoryD-0	Economic...	BiologyC-2	Economic...	HistoryB-0
	13:00	FrenchC-2	BiologyA-1	BiologyA-1	PhysicsC-4	Geograph...	MathC-5	MathC-5	Geograph...
	14:00	EnglishB-3	Economic...	HistoryC-2	Germane-3	Economic...	Chemistry...	Economic...	Chemistry...
	15:00	Chemistry...	ArtB-2	HistoryC-3	Germane-2	SpanishB-2	Chemistry...	FrenchC-1	Chemistry...
	Unassigned								

Constraint matches | X | Latest best score: Ohard-26soft

Package Explorer Navigator Quick Access

OptaPlannerExamplesApp.java Lecture.java

```

2 * Copyright 2010 Red Hat, Inc. and/or its affiliates.
3 *
4 package org.optaplanner.examples.curriculumcourse.domain;
5
6 import java.util.List;
7
8 import org.knowm.xstream.annotations.XStreamAlias("lecture");
9
10 public class Lecture extends AbstractPersistable {
11
12     private Course course;
13     private int lectureIndexInCourse;
14     private boolean pinned;
15
16     // Planning variables: changes during planning, between score calculations.
17     private Period period;
18     private Room room;
19
20     public Course getCourse() {
21         return course;
22     }
23
24     public void setCourse(Course course) {
25         this.course = course;
26     }
27
28     public int getLectureIndexInCourse() {
29         return lectureIndexInCourse;
30     }
31
32     public void setLectureIndexInCourse(int lectureIndexInCourse) {
33         this.lectureIndexInCourse = lectureIndexInCourse;
34     }
35
36     @PlanningPin
37     public boolean isPinned() {
38         return pinned;
39     }
40
41     public void setPinned(boolean pinned) {
42         this.pinned = pinned;
43     }
44
45     @PlanningVariable(valueRangeProviderRefs = {"periodRange"}, strengthWeightFactoryClass = PeriodStrengthWeightFactory.class)
46     public Period getPeriod() {
47         return period;
48     }
49
50     public void setPeriod(Period period) {
51         this.period = period;
52     }
53
54     @PlanningVariable(valueRangeProviderRefs = {"roomRange"}, strengthWeightFactoryClass = RoomStrengthWeightFactory.class)
55     public Room getRoom() {
56         return room;
57     }
58
59     public void setRoom(Room room) {
60         this.room = room;
61     }
62
63     // *****
64 }
```

org.optaplanner.examples.curriculumcourse.domain.Lecture.java - optaplanner-examples/src/main/java

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Machine Reassignment

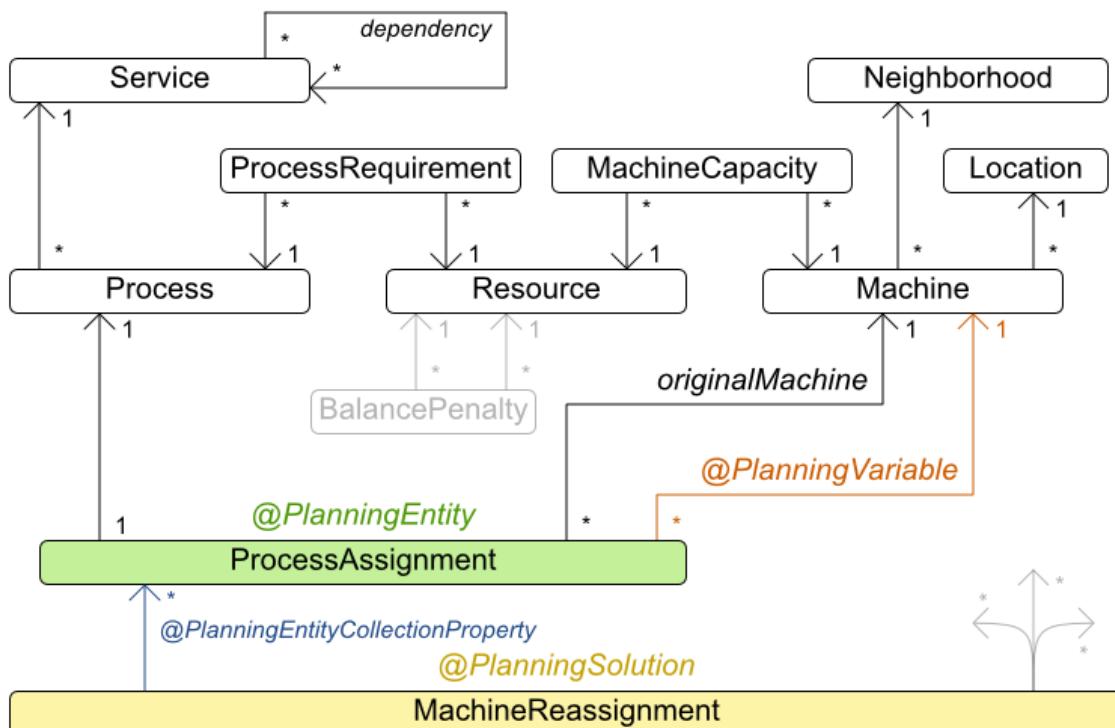
- Business Scenario / Problem Description
- Assign each process to a machine. All processes already have an original (unoptimized) assignment. Each process requires an amount of each resource (such as CPU, RAM, ...). This is a more complex version of the Cloud Balancing example.
- Hard constraints:
 - Maximum capacity: The maximum capacity for each resource for each machine must not be exceeded.
 - Conflict: Processes of the same service must run on distinct machines.
 - Spread: Processes of the same service must be spread out across locations.
 - Dependency: The processes of a service depending on another service must run in the neighborhood of a process of the other service.
 - Transient usage: Some resources are transient and count towards the maximum capacity of both the original machine as the newly assigned machine.
- Soft constraints:
 - Load: The safety capacity for each resource for each machine should not be exceeded.
 - Balance: Leave room for future assignments by balancing the available resources on each machine.
 - Process move cost: A process has a move cost.
 - Service move cost: A service has a move cost.
 - Machine move cost: Moving a process from machine A to machine B has another A-B specific move cost.
- The problem is defined by [the Google ROADEF/EURO Challenge 2012](#).

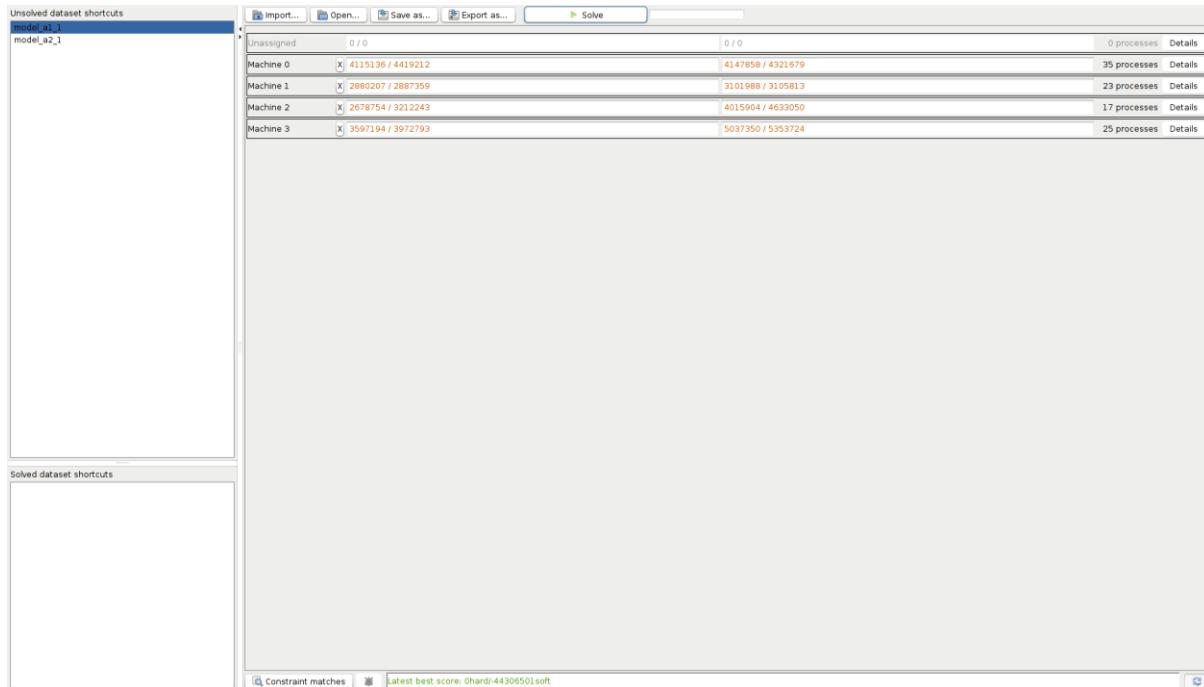
<http://challenge.roadef.org/2012/en/>

© 2018 National University of Singapore. All Rights Reserved

303

Machine reassignment class diagram





The screenshot shows the Eclipse IDE interface with the following details:

- Package Explorer:** Shows the project structure with packages like org.optaplanner.examples.machinereassignment.app, org.optaplanner.examples.machinereassignment.domain, and org.optaplanner.examples.machinereassignment.solver.
- Navigator:** Shows the contents of the selected package, including classes like MrMachine, MrProcess, MrMachineCapacity, MrNeighborhood, and MrProcessAssignment.
- MrProcessAssignment.java:** The active code editor displays the implementation of the MrProcessAssignment class. It includes imports for java.util.Offers, java.util.List, and org.optaplanner.core.domain.common.AbstractPersistable. The class extends AbstractPersistable and implements MrProcessAssignment. It has fields for process, originalMachine, machine, and moved. Methods include setProcess, setOriginalMachine, setMachine, and getMachine. It also includes logic for move costs and neighborhood handling.

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

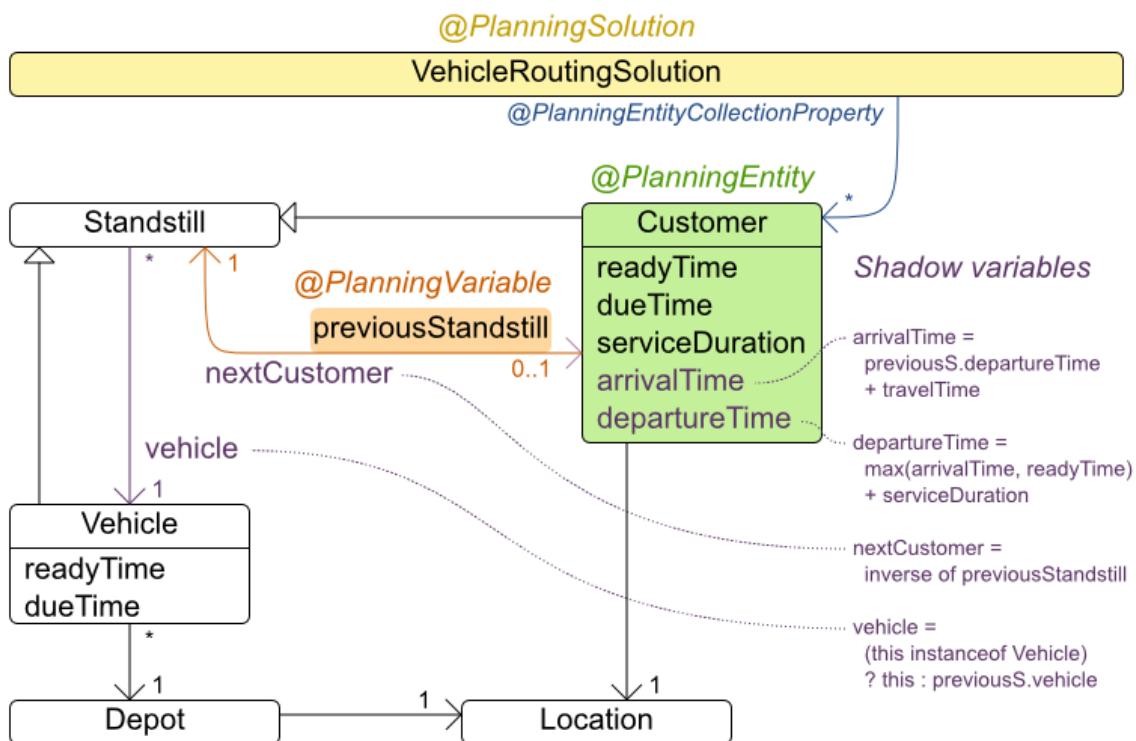
KIE OptaPlanner Deep Dive – Vehicle Routing

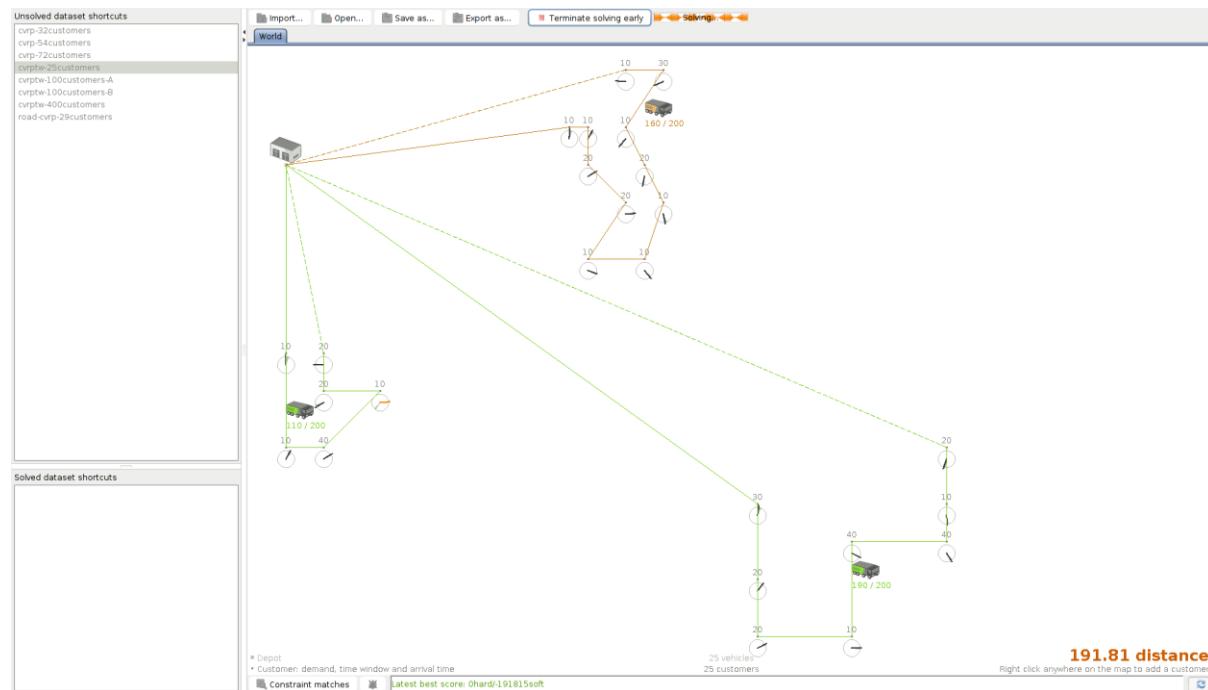
- Business Scenario / Problem Description
- Using a fleet of vehicles, pick up the objects of each customer and bring them to the depot. Each vehicle can service multiple customers, but it has a limited capacity.
- Besides the basic case (CVRP), there is also a variant with time windows (CVRPTW).
- Hard constraints:
 - Vehicle capacity: a vehicle cannot carry more items than its capacity.
 - Time windows (only in CVRPTW):
 - Travel time: Traveling from one location to another takes time.
 - Customer service duration: a vehicle must stay at the customer for the length of the service duration.
 - Customer ready time: a vehicle may arrive before the customer's ready time, but it must wait until the ready time before servicing.
 - Customer due time: a vehicle must arrive on time, before the customer's due time.
- Soft constraints:
 - Total distance: minimize the total distance driven (fuel consumption) of all vehicles.
- The capacitated vehicle routing problem (CVRP) and its time-windowed variant (CVRPTW) are defined by [the VRP web](http://neo.lcc.uma.es/vrp/). <http://neo.lcc.uma.es/vrp/>

© 2018 National University of Singapore. All Rights Reserved

311

Vehicle routing class diagram





Customer.java

```

2 * Copyright 2012 Red Hat, Inc. and/or its affiliates.
3
4 package org.optaplanner.examples.vehiclerouting.domain;
5
6 import com.thoughtworks.xstream.annotations.XStreamAlias;
7
8 @PlanningEntity(difficultyWeightFactoryClass = DepotAngleCustomerDifficultyWeightFactory.class)
9 @XStreamAlias("VrpCustomer")
10 @XStreamAsList("customers")
11
12 public class Customer extends AbstractPersistable implements Standstill {
13     protected Location location;
14     protected int demand;
15     // Planning variables: changes during planning, between score calculations.
16     protected Standstill previousStandstill;
17     // Shadow variables
18     protected Customer nextCustomer;
19     protected Vehicle vehicle;
20
21     @Override
22     public Location getLocation() {
23         return location;
24     }
25
26     public void setLocation(Location location) {
27         this.location = location;
28     }
29
30     public int getDemand() {
31         return demand;
32     }
33
34     public void setDemand(int demand) {
35         this.demand = demand;
36     }
37
38     @PlanningVariable(valueRangeProviderRefs = {"vehicleRange", "customerRange"}, graphType = PlanningVariableGraphType.CHAINED)
39     public Standstill getPreviousStandstill() {
40         return previousStandstill;
41     }
42
43     @Override
44     public void setPreviousStandstill(Standstill previousStandstill) {
45         this.previousStandstill = previousStandstill;
46     }
47
48     @Override
49     public Customer getNextCustomer() {
50         return nextCustomer;
51     }
52
53     @Override
54     public void setNextCustomer(Customer nextCustomer) {
55         this.nextCustomer = nextCustomer;
56     }
57
58     @Override
59     @AnchorShadowVariable(sourceVariableName = "previousStandstill")
60     public Vehicle getVehicle() {
61         return vehicle;
62     }
63
64 }

```

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Project Job Scheduling

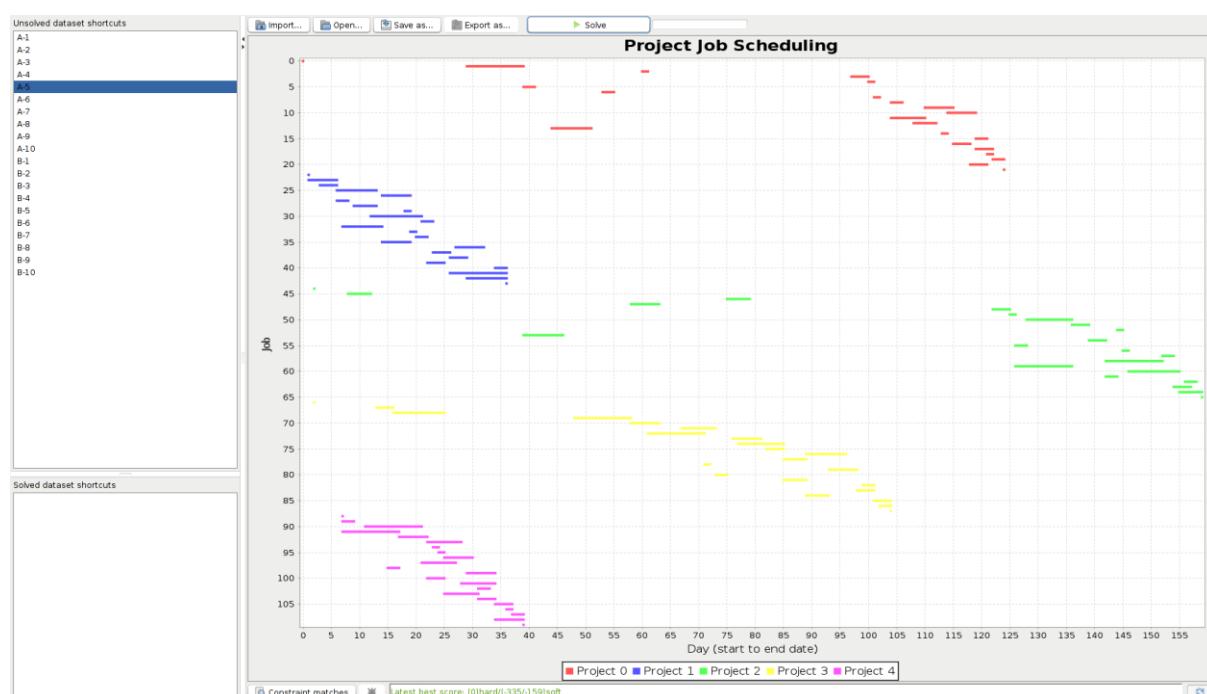
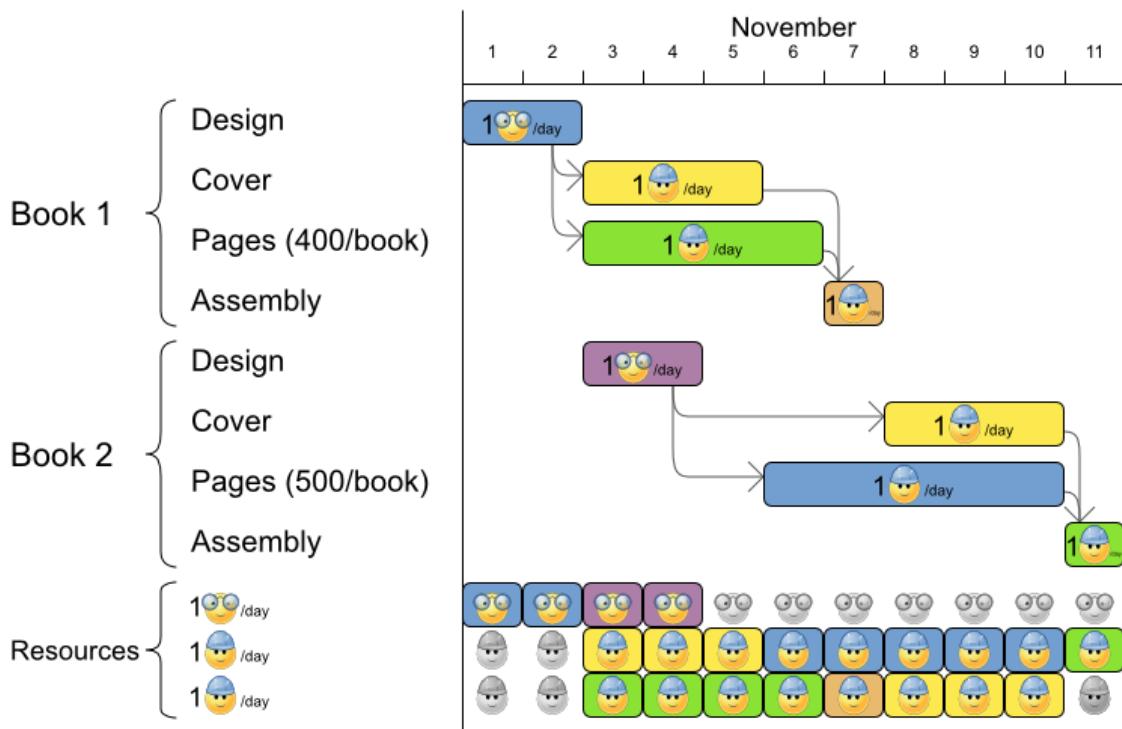
- **Business Scenario / Problem Description**
- Schedule all jobs in time and execution mode to minimize project delays. Each job is part of a project. A job can be executed in different ways: each way is an execution mode that implies a different duration but also different resource usages. This is a form of flexible job shop scheduling.
- **Hard constraints:**
 - Job precedence: a job can only start when all its predecessor jobs are finished.
 - Resource capacity: do not use more resources than available.
 - Resources are local (shared between jobs of the same project) or global (shared between all jobs)
 - Resource are renewable (capacity available per day) or nonrenewable (capacity available for all days)
- **Medium constraints:**
 - Total project delay: minimize the duration (makespan) of each project.
- **Soft constraints:**
 - Total makespan: minimize the duration of the whole multi-project schedule.
 - The problem is defined by the MISTA 2013 challenge.
- **The problem is defined by [the MISTA 2013 challenge](http://gent.cs.kuleuven.be/mista2013challenge/).** <http://gent.cs.kuleuven.be/mista2013challenge/>

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram

Project job scheduling

For each job, choose an execution mode and a start time.



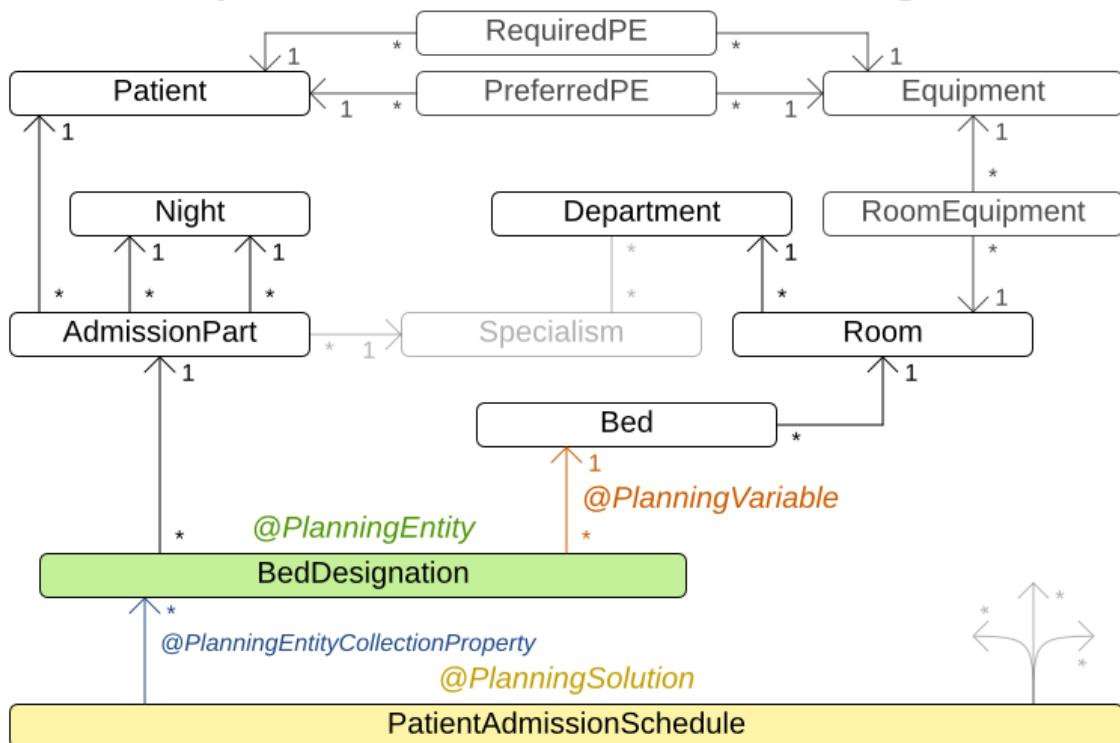
ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Hospital Bed Planning (PAS)

- Business Scenario / Problem Description
- Assign each patient (that will come to the hospital) into a bed for each night that the patient will stay in the hospital. Each bed belongs to a room and each room belongs to a department. The arrival and departure dates of the patients is fixed: only a bed needs to be assigned for each night. This problem features overconstrained datasets.
- Hard constraints:
 - Two patients must not be assigned to the same bed in the same night. Weight: $-1000\text{hard} * \text{conflictNightCount}$.
 - A room can have a gender limitation: only females, only males, the same gender in the same night or no gender limitation at all. Weight: $-50\text{hard} * \text{nightCount}$.
 - A department can have a minimum or maximum age. Weight: $-100\text{hard} * \text{nightCount}$.
 - A patient can require a room with specific equipment(s). Weight: $-50\text{hard} * \text{nightCount}$.
- Medium constraints:
 - Assign every patient to a bed, unless the dataset is over-constrained. Weight: $-1\text{medium} * \text{nightCount}$.
- Soft constraints:
 - A patient can prefer a maximum room size, for example if he/she wants a single room. Weight: $-8\text{soft} * \text{nightCount}$.
 - A patient is best assigned to a department that specializes in his/her problem. Weight: $-10\text{soft} * \text{nightCount}$.
 - A patient is best assigned to a room that specializes in his/her problem. Weight: $-20\text{soft} * \text{nightCount}$.
 - That room specialty should be priority 1. Weight: $-10\text{soft} * (\text{priority} - 1) * \text{nightCount}$.
 - A patient can prefer a room with specific equipment(s). Weight: $-20\text{soft} * \text{nightCount}$.
- The problem is a variant on [Kaho's Patient Scheduling](https://people.cs.kuleuven.be/~wim.vancroonenburg/pas/). <https://people.cs.kuleuven.be/~wim.vancroonenburg/pas/>

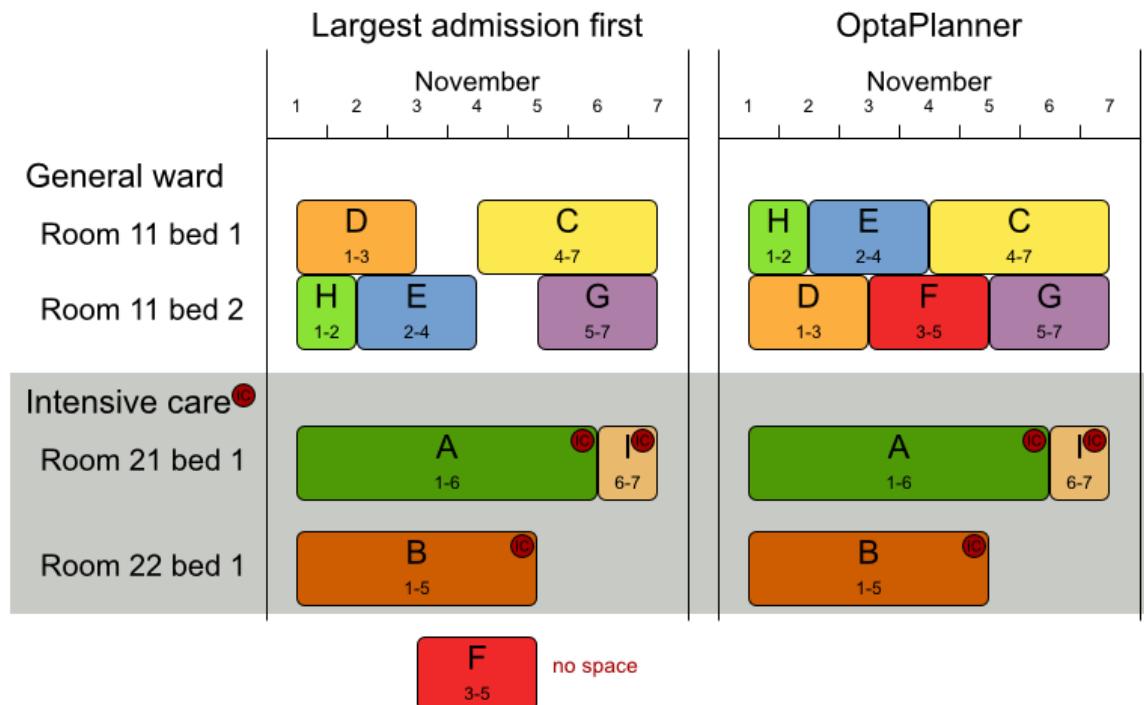
32B

Hospital bed allocation class diagram



Patient admission schedule

Assign each patient a hospital bed.



Unsolved dataset shortcuts

Department	Room	Bed	1-JAN	2-JAN	3-JAN	4-JAN	5-JAN	6-JAN	7-JAN	8-JAN	9-JAN	10-JAN	11-JAN	12-JAN	13-JAN	14-JAN
Unassigned			Patient1	Patient2	Patient3	Patient4	Patient5	Patient6	Patient7	Patient8	Patient9	Patient10	Patient11	Patient12	Patient13	Patient14
			Patient15	Patient16	Patient17	Patient18	Patient19	Patient20	Patient21	Patient22	Patient23	Patient24	Patient25	Patient26	Patient27	Patient28
			Patient29	Patient30	Patient31	Patient32	Patient33	Patient34	Patient35	Patient36	Patient37	Patient38	Patient39	Patient40	Patient41	Patient42
			Patient43	Patient44	Patient45	Patient46	Patient47	Patient48	Patient49	Patient50	Patient51	Patient52	Patient53	Patient54	Patient55	Patient56
			Patient57	Patient58	Patient59	Patient60	Patient61	Patient62	Patient63	Patient64	Patient65	Patient66	Patient67	Patient68	Patient69	Patient70
			Patient71	Patient72	Patient73	Patient74	Patient75	Patient76	Patient77	Patient78	Patient79	Patient80	Patient81	Patient82	Patient83	Patient84
			Patient85	Patient86	Patient87	Patient88	Patient89	Patient90	Patient91	Patient92	Patient93	Patient94	Patient95	Patient96	Patient97	Patient98
			Patient99	Patient100	Patient101	Patient102	Patient103	Patient104	Patient105	Patient106	Patient107	Patient108	Patient109	Patient110	Patient111	Patient112

Solved dataset shortcuts

Department	Room	Bed	1-JAN	2-JAN	3-JAN	4-JAN	5-JAN	6-JAN	7-JAN	8-JAN	9-JAN	10-JAN	11-JAN	12-JAN	13-JAN	14-JAN
			Patient1	Patient2	Patient3	Patient4	Patient5	Patient6	Patient7	Patient8	Patient9	Patient10	Patient11	Patient12	Patient13	Patient14
			Patient15	Patient16	Patient17	Patient18	Patient19	Patient20	Patient21	Patient22	Patient23	Patient24	Patient25	Patient26	Patient27	Patient28
			Patient29	Patient30	Patient31	Patient32	Patient33	Patient34	Patient35	Patient36	Patient37	Patient38	Patient39	Patient40	Patient41	Patient42
			Patient43	Patient44	Patient45	Patient46	Patient47	Patient48	Patient49	Patient50	Patient51	Patient52	Patient53	Patient54	Patient55	Patient56
			Patient57	Patient58	Patient59	Patient60	Patient61	Patient62	Patient63	Patient64	Patient65	Patient66	Patient67	Patient68	Patient69	Patient70
			Patient71	Patient72	Patient73	Patient74	Patient75	Patient76	Patient77	Patient78	Patient79	Patient80	Patient81	Patient82	Patient83	Patient84
			Patient85	Patient86	Patient87	Patient88	Patient89	Patient90	Patient91	Patient92	Patient93	Patient94	Patient95	Patient96	Patient97	Patient98
			Patient99	Patient100	Patient101	Patient102	Patient103	Patient104	Patient105	Patient106	Patient107	Patient108	Patient109	Patient110	Patient111	Patient112

Constraint matches: latest best score: 0hard/1medium/34754soft

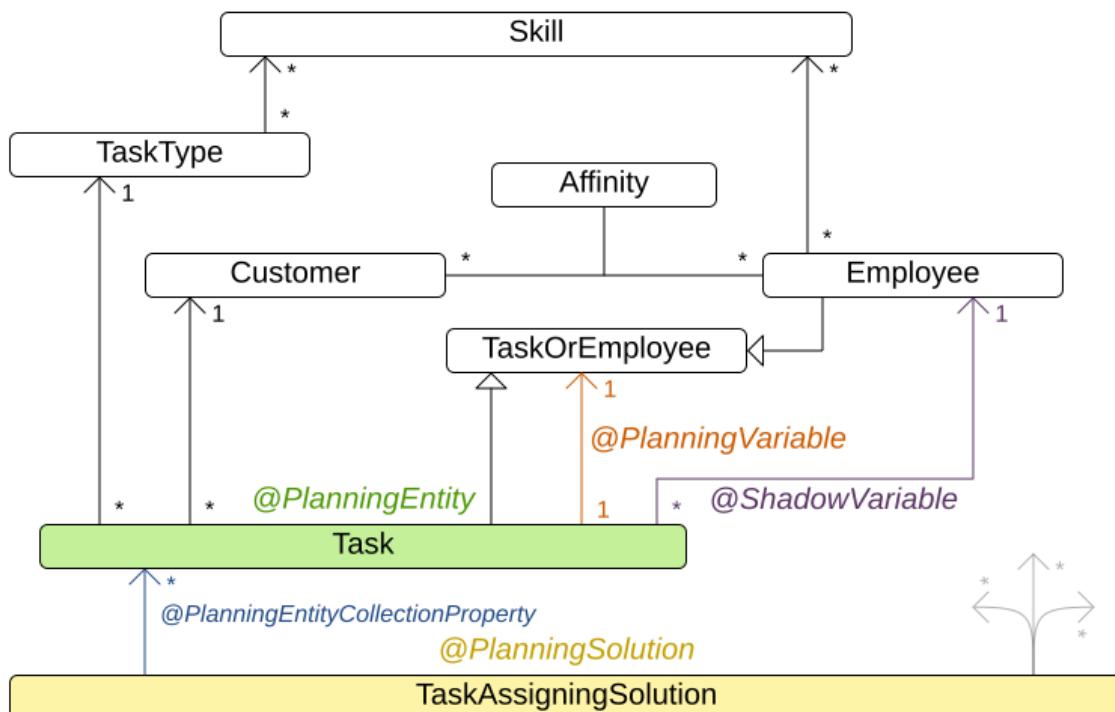
ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Task assigning

- Business Scenario / Problem Description
- Assign each task to a spot in an employee's queue. Each task has a duration which is affected by the employee's affinity level with the task's customer.
- Hard constraints:
 - Skill: Each task requires one or more skills. The employee must possess all these skills.
- Soft level 0 constraints:
 - Critical tasks: Complete critical tasks first, sooner than major and minor tasks.
- Soft level 1 constraints:
 - Minimize makespan: Reduce the time to complete all tasks.
 - Start with the longest working employee first, then the second longest working employee and so forth, to create fairness and load balancing.
- Soft level 2 constraints:
 - Major tasks: Complete major tasks as soon as possible, sooner than minor tasks.
- Soft level 3 constraints:
 - Minor tasks: Complete minor tasks as soon as possible.

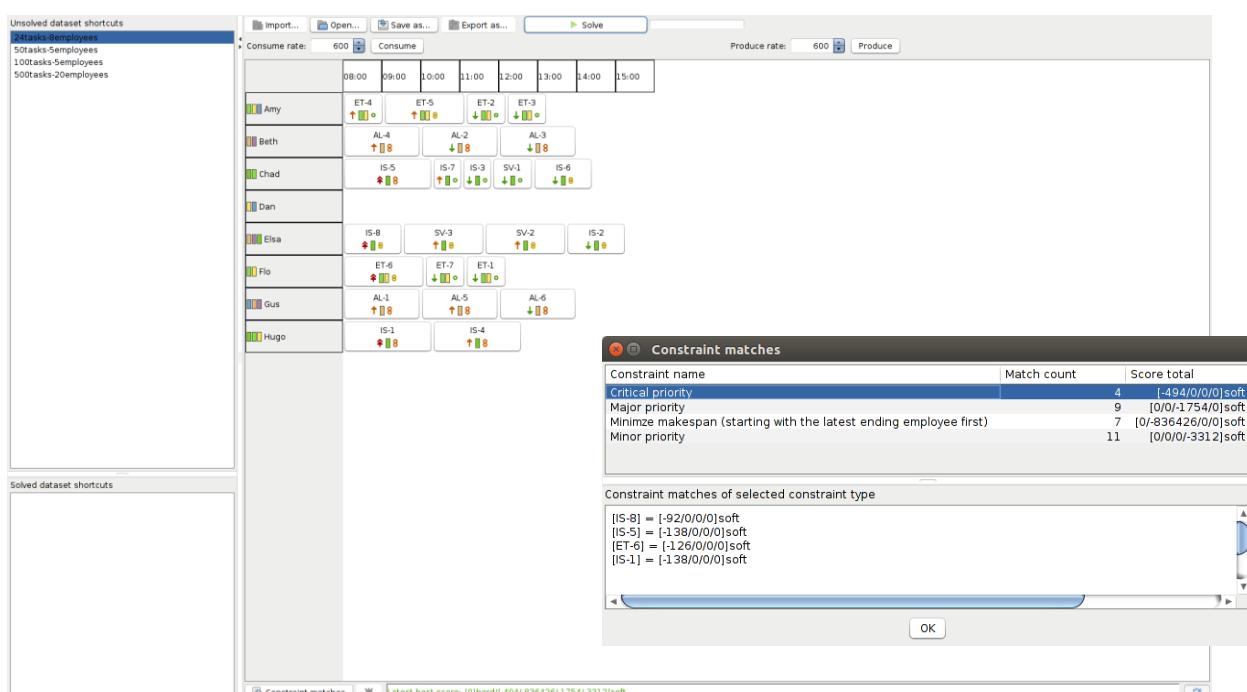
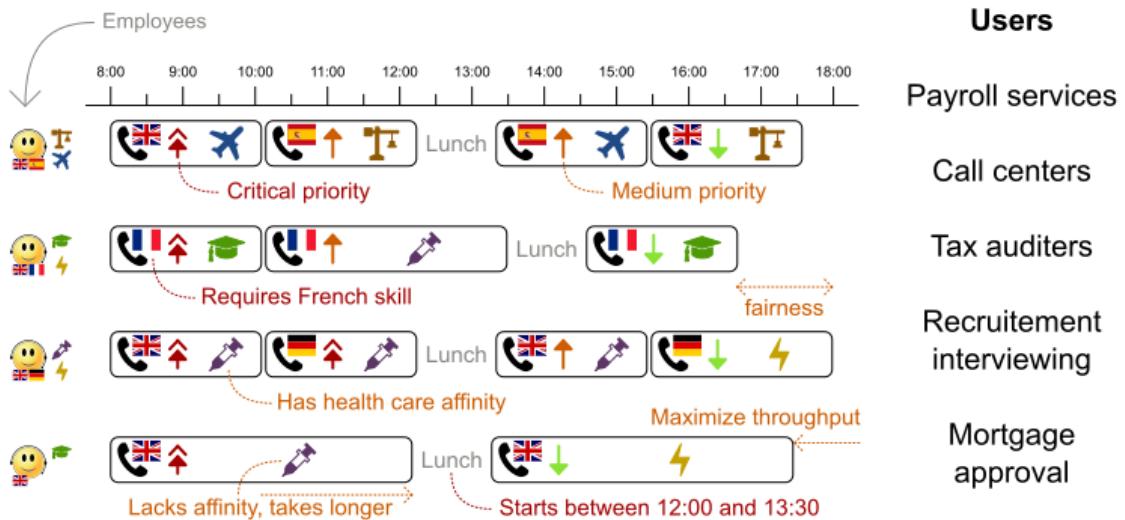
337

Task assigning class diagram



Task assigning

Optimize the task queue of every employee by reassigning and reordering tasks.



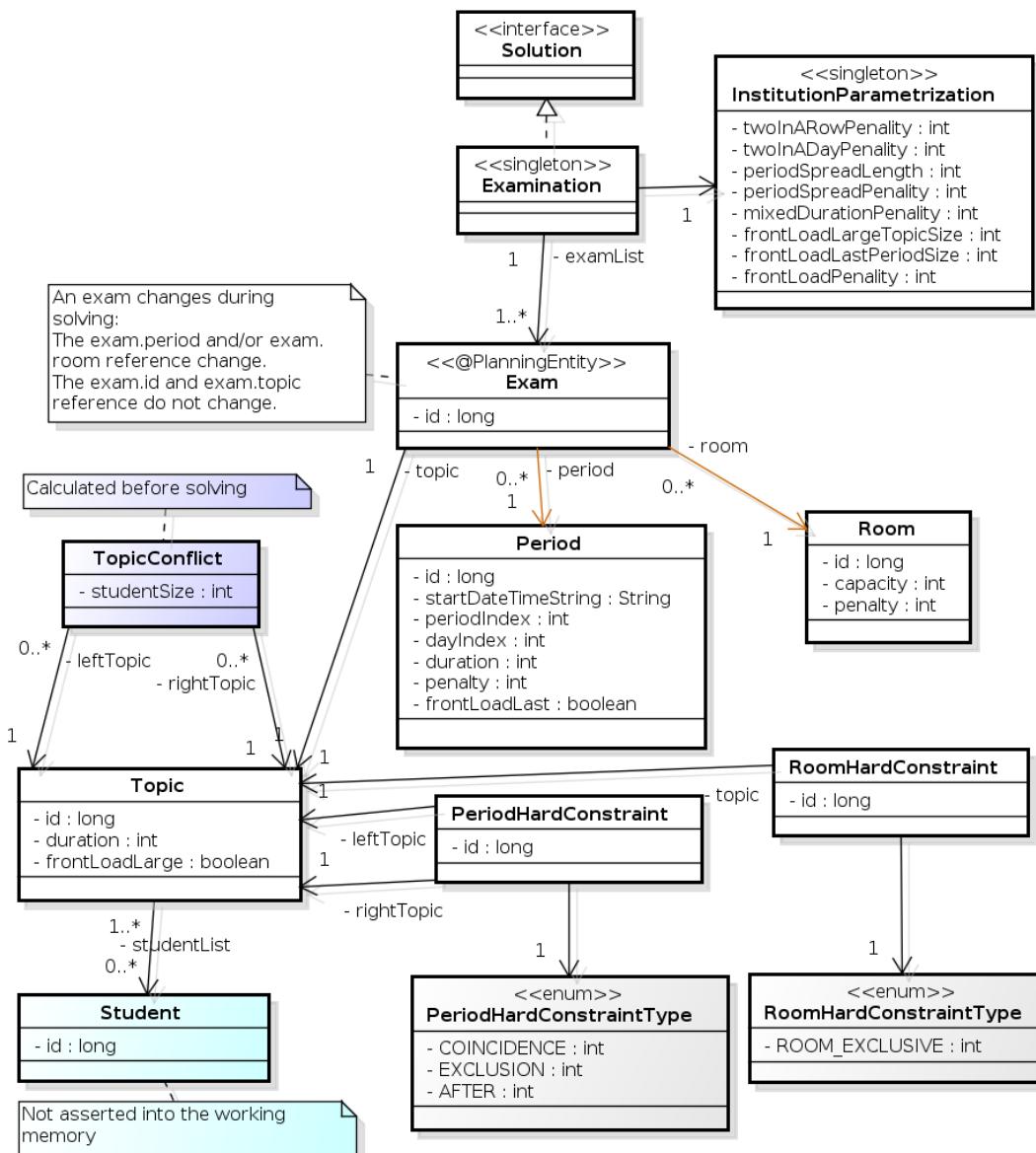
ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Exam Timetabling

- Business Scenario / Problem Description
- Schedule each exam into a period and into a room. Multiple exams can share the same room during the same period.
- Hard constraints:
 - Exam conflict: two exams that share students must not occur in the same period.
 - Room capacity: A room's seating capacity must suffice at all times.
 - Period duration: A period's duration must suffice for all of its exams.
 - Period related hard constraints (specified per dataset):
 - Coincidence: two specified exams must use the same period (but possibly another room).
 - Exclusion: two specified exams must not use the same period.
 - After: A specified exam must occur in a period after another specified exam's period.
 - Room related hard constraints (specified per dataset):
 - Exclusive: one specified exam should not have to share its room with any other exam.
- Soft constraints (each of which has a parametrized penalty):
 - The same student should not have two exams in a row.
 - The same student should not have two exams on the same day.
 - Period spread: two exams that share students should be a number of periods apart.
 - Mixed durations: two exams that share a room should not have different durations.
 - Front load: Large exams should be scheduled earlier in the schedule.
 - Period penalty (specified per dataset): Some periods have a penalty when used.
 - Room penalty (specified per dataset): Some rooms have a penalty when used.
 - It uses large test data sets of real-life universities.
- The problem is defined by [the International Timetabling Competition 2007 track 1](#).

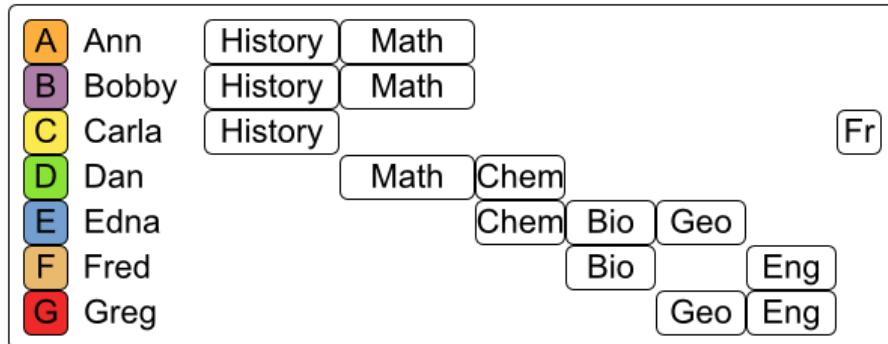
http://www.cs.qub.ac.uk/ittc2007/examtrack/exam_track_index.htm

346



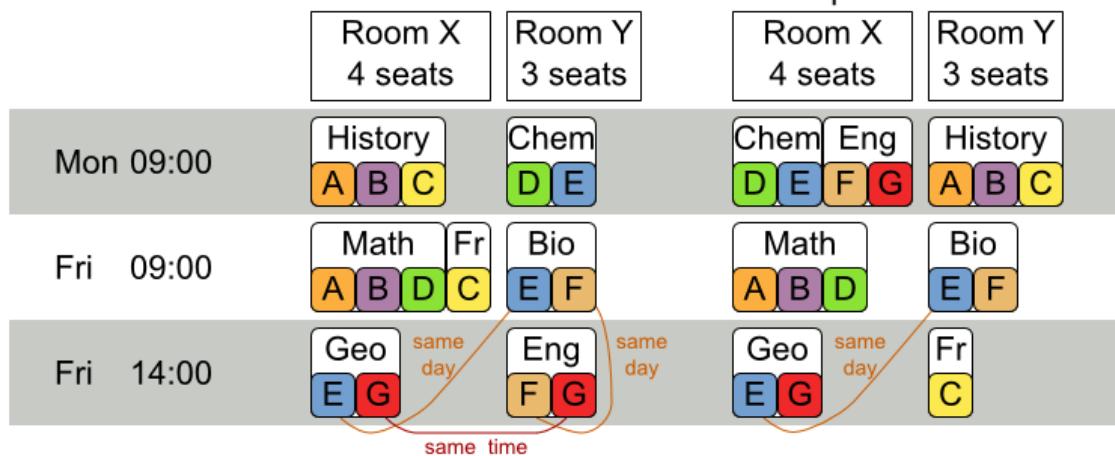
Examination timetabling

Assign each exam a period and a room.



Most students first

OptaPlanner



2 exams in a row penalty	15
2 exams in a day penalty	5
Period spread length	1
Period spread penalty	1
Exams of mixed duration penalty	25
Front load: large exam size	250
Front load: last period size	30
Front load: penalty	5

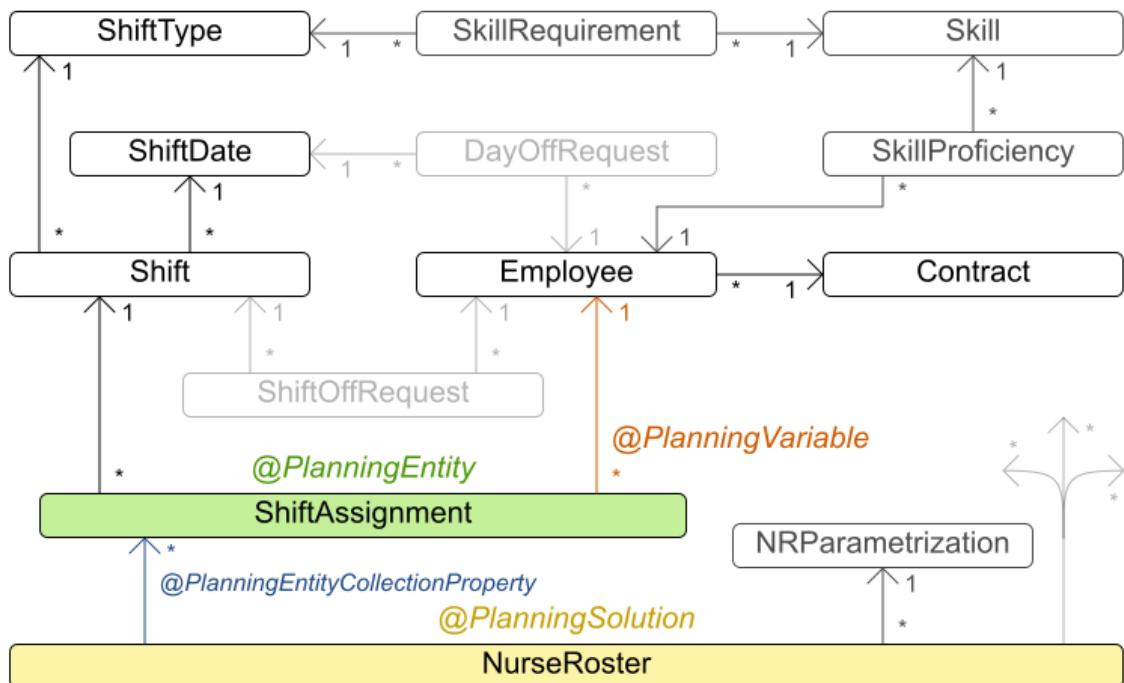
ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Employee Rostering

- **Business Scenario / Problem Description**
- For each shift, assign a nurse to work that shift.
- **Hard constraints:**
 - No unassigned shifts (built-in): Every shift needs to be assigned to an employee.
 - Shift conflict: An employee can have only one shift per day.
- **Soft constraints:**
 - Contract obligations. The business frequently violates these, so they decided to define these as soft constraints instead of hard constraints.
 - Minimum and maximum assignments: Each employee needs to work more than x shifts and less than y shifts (depending on their contract).
 - Minimum and maximum consecutive working days: Each employee needs to work between x and y days in a row (depending on their contract).
 - Minimum and maximum consecutive free days: Each employee needs to be free between x and y days in a row (depending on their contract).
 - Minimum and maximum consecutive working weekends: Each employee needs to work between x and y weekends in a row (depending on their contract).
 - Complete weekends: Each employee needs to work every day in a weekend or not at all.
 - Identical shift types during weekend: Each weekend shift for the same weekend of the same employee must be the same shift type.
 - Unwanted patterns: A combination of unwanted shift types in a row. For example: a late shift followed by an early shift followed by a late shift.
- **Employee wishes:**
 - Day on request: An employee wants to work on a specific day.
 - Day off request: An employee does not want to work on a specific day.
 - Shift on request: An employee wants to be assigned to a specific shift.
 - Shift off request: An employee does not want to be assigned to a specific shift.
- **Alternative skill:** An employee assigned to a skill should have a proficiency in every skill required by that shift.
- The problem is defined by [the International Nurse Rostering Competition 2010](http://www.kuleuven-kortrijk.be/nrpcompetition) <http://www.kuleuven-kortrijk.be/nrpcompetition>

356

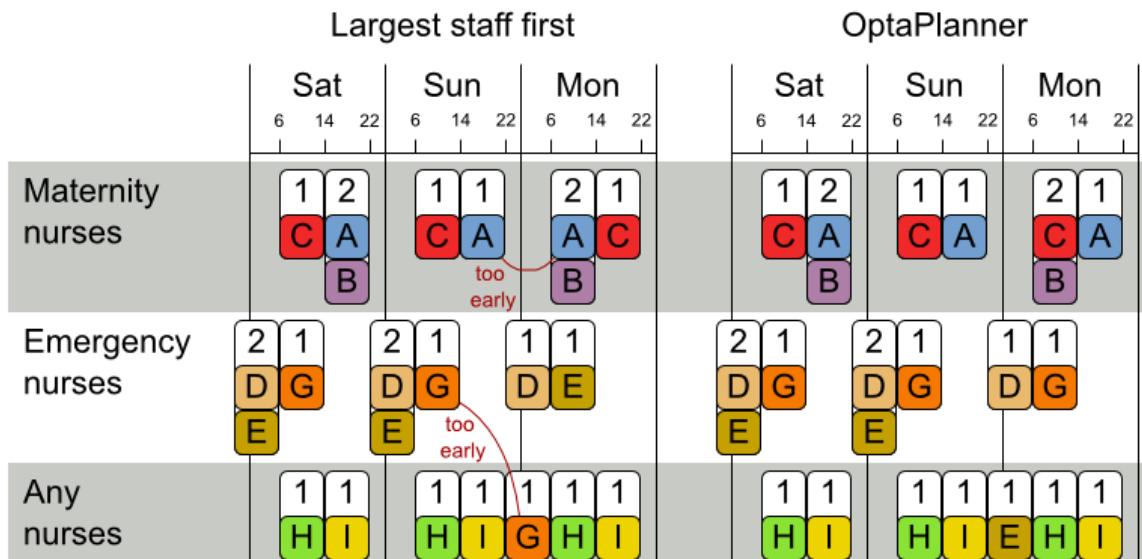
Nurse rostering class diagram



Employee shift rostering

Populate each work shift with a nurse.

Maternity nurses			Emergency nurses			Basic nurses		
A Ann	B Beth	C Cory	D Dan	E Elin	G Greg	H Hue	I Ilse	



Unsolved dataset shortcuts

```
long01
long02
long_hint01
long_hint02
medium01
medium02
medium03
medium04
medium05
medium_hint01
medium_hint02
medium_hint03
medium_hint04
medium_late01_initialized
medium_late02
medium_late03
medium_late04
medium_late05
sprint01
sprint02
sprint_hint01
sprint_hint02
```

Solved dataset shortcuts

Planning window start: Fri 1 Jan Advance 1 day into the future E = Early shift, L = Late shift...

Employee	Fri 1 Jan	Sat 2 Jan	Sun 3 Jan	Mon 4 Jan	Tue 5 Jan	Wed 6 Jan	Thu 7 Jan	Fri 8 Jan	Sat 9 Jan	Sun 10 Jan	Mon 11 Jan	Tue 12 Jan	Wed 13 Jan	Thu 14 Jan	Fri 15 Jan	Sat 16 Jan
Employee 0	X				N	D	D	E	E	E		N	E			
Employee 1	X	E	E	E	D						N	L	E	L	L	
Employee 2	X			N	E	D	L			L	U	L	U	D		
Employee 3	X	N	L	L	E	L	E	L		L	E					
Employee 4	X	U	N	N	L			N	E	N	R	D	D	L	E	N
Employee 5	X	D	D	D		K	L	E	L	E	B		E	E		
Employee 6	X			D	L			L	L	B		D	S	L		
Employee 7	X			L	E	L		E	D	D	E				N	D
Employee 8	X	U			E				N	U						
Employee 9	X	E			E			N				D	L			

Constraint matches

Constraint name	Match count	Score total
Minimum and maximum number of assignments	10	-30soft
dayOffRequest	27	-27soft

Constraint matches of selected constraint type

```
[2010-01-09/D, 2010-01-08 OFF_0] = lsoft
[2010-01-07/D, 2010-01-07 OFF_0] = lsoft
[2010-01-27/E, 2010-01-27 OFF_0] = lsoft
[2010-01-15/L, 2010-01-15 OFF_1] = lsoft
[2010-01-18/E, 2010-01-18 OFF_1] = lsoft
[2010-01-17/L, 2010-01-17 OFF_1] = lsoft
[2010-01-04/E, 2010-01-04 OFF_1] = lsoft
```

OK

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Traveling Tournament Problem (TTP)



- **Business Scenario / Problem Description**
- **Schedule matches between n teams.**
- **Hard constraints:**
 - Each team plays twice against every other team: once home and once away.
 - Each team has exactly one match on each timeslot.
 - No team must have more than three consecutive home or three consecutive away matches.
 - No repeaters: no two consecutive matches of the same two opposing teams.
- **Soft constraints:**
 - Minimize the total distance traveled by all teams.
- **The problem is defined on [Michael Trick's website](#).**
<http://mat.tepper.cmu.edu/TOURN/>

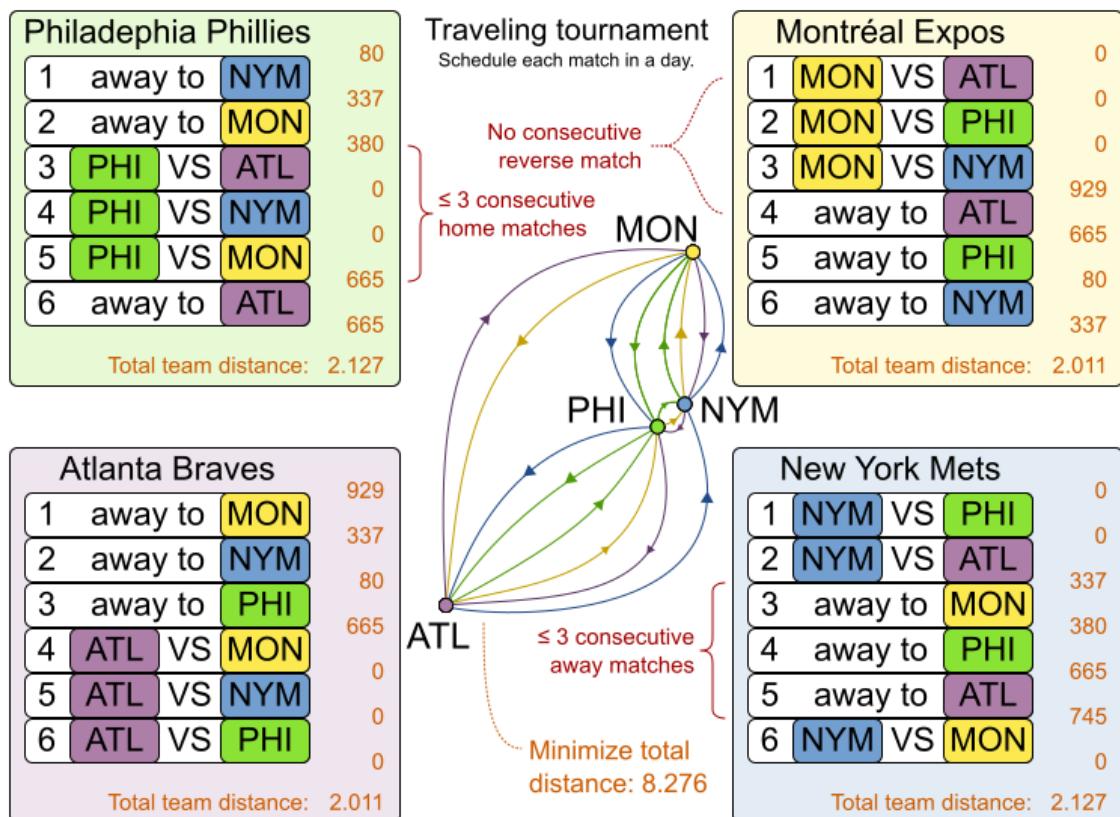
366

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram



369



Unsolved dataset shortcuts

Solved dataset shortcuts

Teams

Day	ATL	NYM	PHL	MON	Unassigned
0	MON	PHL	NYM	ATL	
1	NYM	ATL	MON	PHL	
2	PHL	MON	ATL	NYM	
3	MON	PHL	NYM	ATL	
4	NYM	ATL	MON	PHL	
5	PHL	MON	ATL	NYM	
Unassigned					

Constraint matches

Constraint name	Match count	Score total
awayToAwayHop	7	-2544soft
awayToEndHop	2	-1002soft
awayToHomeHop	3	-1790soft
homeToAwayHop	3	-1931soft
startToAwayHop	2	-1009soft

Constraint matches of selected constraint type

```
[PHL+MON, ATL+MON] = -665soft
[PHL+ATL, NYM+ATL] = -80soft
[MON+PHL, NYM+PHL] = -337soft
[ATL+NYM, PHL+NYM] = -665soft
[NYM+MON, PHL+MON] = -80soft
[NYM+ATL, MON+ATL] = -337soft
[PHL+NYM, MON+NYM] = -380soft
```

OK

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Cheap Time Scheduling

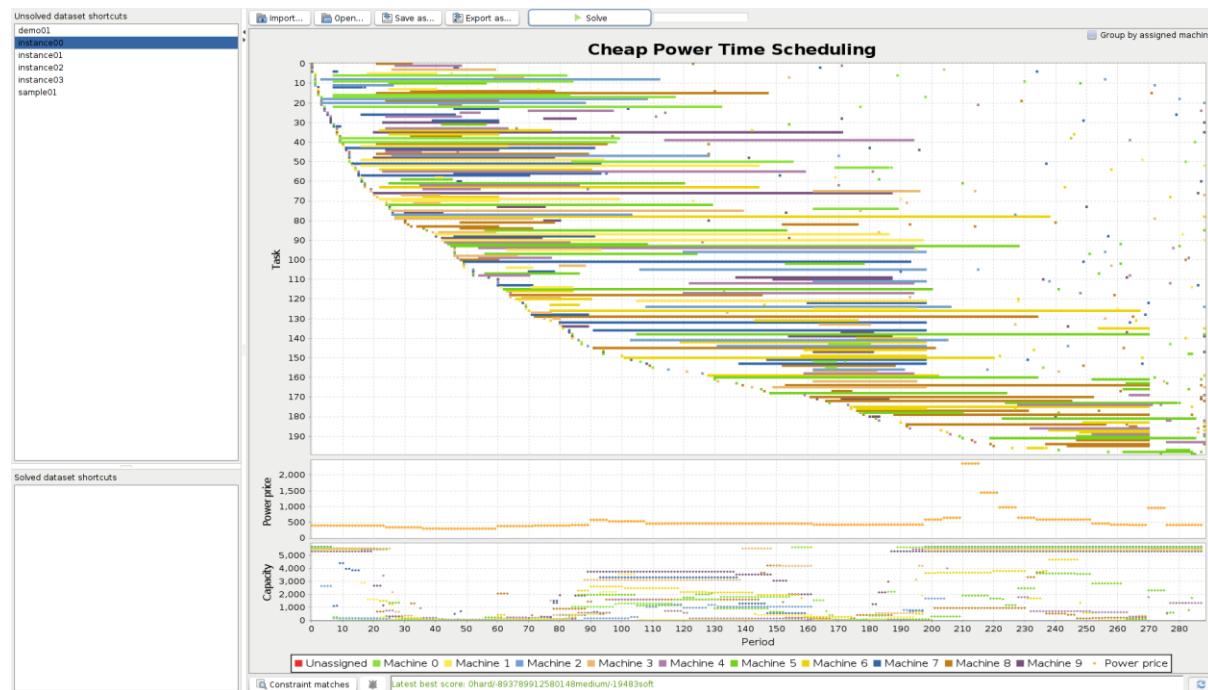
- **Business Scenario / Problem Description**
- **Schedule all tasks in time and on a machine to minimize power cost. Power prices differs in time.**
This is a form of job shop scheduling.
- **Hard constraints:**
 - Start time limits: each task must start between its earliest start and latest start limit.
 - Maximum capacity: the maximum capacity for each resource for each machine must not be exceeded.
 - Startup and shutdown: each machine must be active in the periods during which it has assigned tasks. Between tasks it is allowed to be idle to avoid startup and shutdown costs.
- **Medium constraints:**
 - Power cost: minimize the total power cost of the whole schedule.
 - Machine power cost: Each active or idle machine consumes power, which infers a power cost (depending on the power price during that time).
 - Task power cost: Each task consumes power too, which infers a power cost (depending on the power price during its time).
 - Machine startup and shutdown cost: Every time a machine starts up or shuts down, an extra cost is inflicted.
- **Soft constraints (addendum to the original problem definition):**
 - Start early: prefer starting a task sooner rather than later.

375

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram

377



The figure shows a screenshot of the Eclipse IDE. The left side displays the 'Package Explorer' and 'Navigator' panes, showing the project structure of 'OptaplannerExamplesApp' with various Java files like 'TaskAssignment.java', 'TaskRequirement.java', and 'Machine.java'. The right side shows the code editor for 'TaskAssignment.java'. The code is as follows:

```

/*
 * Copyright 2014 Red Hat, Inc. and/or its affiliates.
 */
package org.optaplanner.examples.cheatime.domain;
import com.thoughtworks.xstream.annotations.XStreamAlias;
@PlanningEntity(difficultyComparatorClass = TaskAssignmentDifficultyComparator.class)
@XStreamAlias("CTaskAssignment")
public class TaskAssignment extends AbstractPersistable {
    private Task task;
    // Planning variables: changes during planning, between score calculations.
    private Machine machine;
    private Integer startPeriod;
    public Task getTask() {
        return task;
    }
    public void setTask(Task task) {
        this.task = task;
    }
    @PlanningVariable(valueRangeProviderRefs = {"machineRange"})
    public Machine getMachine() {
        return machine;
    }
    public void setMachine(Machine machine) {
        this.machine = machine;
    }
    @PlanningVariable(valueRangeProviderRefs = {"startPeriodRange"})
    public Integer getStartPeriod() {
        return startPeriod;
    }
    public void setStartPeriod(Integer startPeriod) {
        this.startPeriod = startPeriod;
    }
    /**
     * The startPeriod is included and the endPeriod is excluded.
     */
    public Integer getEndPeriod() {
        if (startPeriod == null) {
            return null;
        }
        return startPeriod + task.getDuration();
    }
    public String getLabel() {
        return task.getLabel();
    }
}

```

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Investment Asset Class Allocation



- **Business Scenario / Problem Description**
- Decide the relative quantity to invest in each asset class.
- **Hard constraints:**
 - Risk maximum: the total standard deviation must not be higher than the standard deviation maximum.
 - Total standard deviation calculation takes asset class correlations into account by applying Markowitz Portfolio Theory.
 - Region maximum: Each region has a quantity maximum.
 - Sector maximum: Each sector has a quantity maximum.
- **Soft constraints:**
 - Maximize expected return.

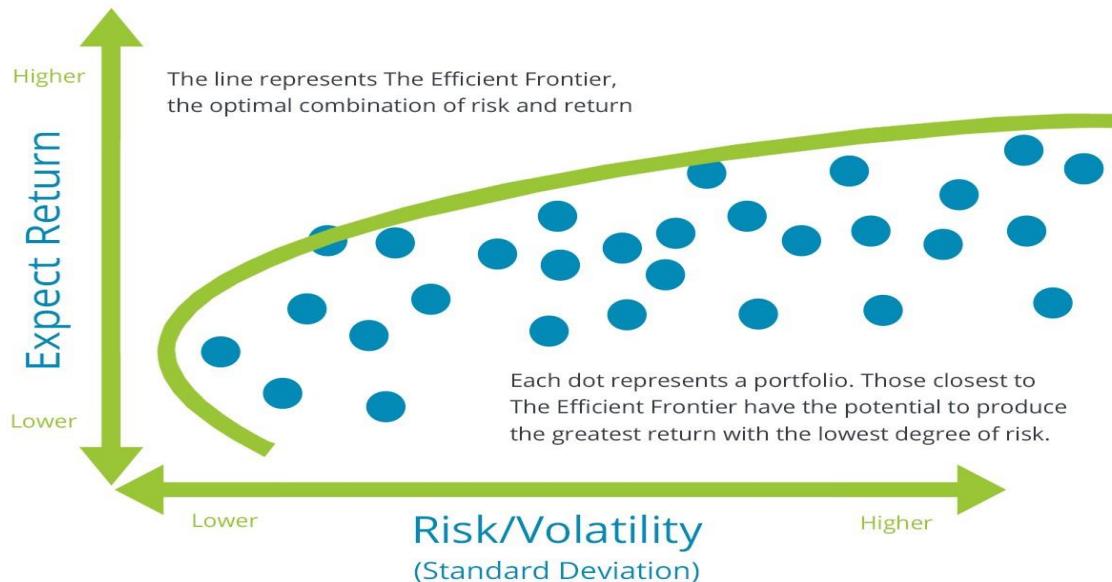
384

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram



388



[Link](https://www.guidedchoice.com/video/dr-harry-markowitz-father-of-modern-portfolio-theory/) <https://www.guidedchoice.com/video/dr-harry-markowitz-father-of-modern-portfolio-theory/>

Unsolved dataset shortcuts

dataset_1	Import...	Open...	Save as...	Export as...	Solve
irrlink_1	Standard deviation maximum 15.0%				

Asset classes Regions Sectors

Asset class	Region	Sector	Expected return	Standard deviation risk	Correlation											
					Red Hat...	Google...	Oracle C...	Apple Inc...	Microsoft...	Tesla Mo...	Ford Mot...	Toyota M...	General...	Starbuck...	McDonald...	
Red Hat, Inc.	Global	Tech	13.6%	29.1%	0.0%	0.050	0.600	0.130	0.140	0.230	0.210	0.080	0.320	0.330	0.000	
Google Inc.	Global	Tech	15.6%	21.5%	8.7%	0.050	0.000	0.260	0.180	0.100	0.098	0.200	0.210	0.210	0.000	
Oracle Corporation	Global	Tech	12.3%	21.7%	1.8%	0.600	0.050	0.000	0.190	0.330	0.140	0.420	0.190	0.500	0.170	-0.010
Apple Inc.	Global	Tech	20.8%	24.1%	29.6%	0.130	0.260	0.190	0.000	0.270	0.010	0.150	0.180	0.250	0.230	0.030
Microsoft Corporation	Global	Tech	17.0%	20.7%	10.3%	0.140	0.180	0.320	0.270	0.000	0.180	0.200	0.250	0.320	0.170	0.160
Tesla Motors, Inc.	Global	Cars	54.7%	53.9%	13.1%	0.230	0.100	0.140	0.010	0.180	0.000	0.320	0.160	0.230	0.240	-0.050
Ford Motor Company	Global	Cars	1.0%	25.9%	0.0%	0.210	0.080	0.420	0.150	0.290	0.320	0.000	0.240	0.830	0.360	0.100
Toyota Motor Corp Ltd Ord	Global	Cars	13.6%	19.2%	16.9%	0.080	0.200	0.190	0.180	0.250	0.160	0.240	0.000	0.360	0.320	0.100
General Motors Company	Global	Cars	2.1%	29.5%	0.0%	0.320	0.210	0.500	0.250	0.320	0.230	0.830	0.360	0.000	0.300	0.090
Starbucks Corporation	Global	Food	33.2%	19.7%	20.0%	0.330	0.210	0.170	0.230	0.170	0.240	0.360	0.320	0.300	0.000	0.310
McDonald's Corporation	Global	Food	8.0%	11.3%	0.0%	0.000	0.210	-0.010	0.030	0.160	-0.050	0.100	0.100	0.090	0.310	0.000
Total			28.0%	15.0%	100.0%											

Constraint matches

Constraint name	Match count	Score total
Maximize expected return	11	279888soft

Constraint matches of selected constraint type

- [11-McDonald's Corporation] = 0
- [10-Starbucks Corporation] = 66400soft
- [9-General Motors Company] = 0
- [8-Toyota Motor Corp Ltd Ord] = 22984soft
- [7-Ford Motor Company] = 0
- [6-Tesla Motors, Inc.] = 71657soft
- [5-Microsoft Corporation] = 18437soft
- [4-Apple Inc.] = 85248soft
- [3-Oracle Corporation] = 2214soft
- [2-Google Inc.] = 12948soft
- [1-Red Hat, Inc.] = 0

OK

Constraint matches Latest best score: 0hard/279888soft

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Conference Scheduling

- **Business Scenario / Problem Description**
- **Assign each conference talk to a timeslot and a room, after the talks have been accepted.**
- **Hard constraints (unless configured otherwise):**
 - Talk type of timeslot: The type of a talk must match the timeslot's talk type.
 - Room unavailable timeslots: A talk's room must be available during the talk's timeslot.
 - Room conflict: Two talks can't use the same room during overlapping timeslots.
 - Speaker unavailable timeslots: Every talk's speaker must be available during the talk's timeslot.
 - Speaker conflict: Two talks can't share a speaker during overlapping timeslots.
 - Generic purpose timeslot and room tags
 - Speaker required timeslot tag: If a speaker has a required timeslot tag, then all his/her talks must be assigned to a timeslot with that tag.
 - Speaker prohibited timeslot tag: If a speaker has a prohibited timeslot tag, then all his/her talks cannot be assigned to a timeslot with that tag.
 - Talk required timeslot tag: If a talk has a required timeslot tag, then it must be assigned to a timeslot with that tag.
 - Talk prohibited timeslot tag: If a talk has a prohibited timeslot tag, then it cannot be assigned to a timeslot with that tag.
 - Speaker required room tag: If a speaker has a required room tag, then all his/her talks must be assigned to a room with that tag.
 - Speaker prohibited room tag: If a speaker has a prohibited room tag, then all his/her talks cannot be assigned to a room with that tag.
 - Talk required room tag: If a talk has a required room tag, then it must be assigned to a room with that tag.
 - Talk prohibited room tag: If a talk has a prohibited room tag, then it cannot be assigned to a room with that tag.
 - Talk prerequisite talks: A talk must be scheduled after all its prerequisite talks.
 - Consecutive talks pause: A speaker who has more than one talk must have a break between them.
 - Talk mutually-exclusive-talks tags: Talks that share such tags must not be scheduled in overlapping timeslots.
- **Medium constraints (unless configured otherwise):**
 - Published timeslot: A published talk must not be scheduled at a different timeslot than currently published. If a hard constraint's input data changes after publishing (such as speaker unavailability), then this medium constraint will be minimally broken to attain a new feasible solution.

394

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

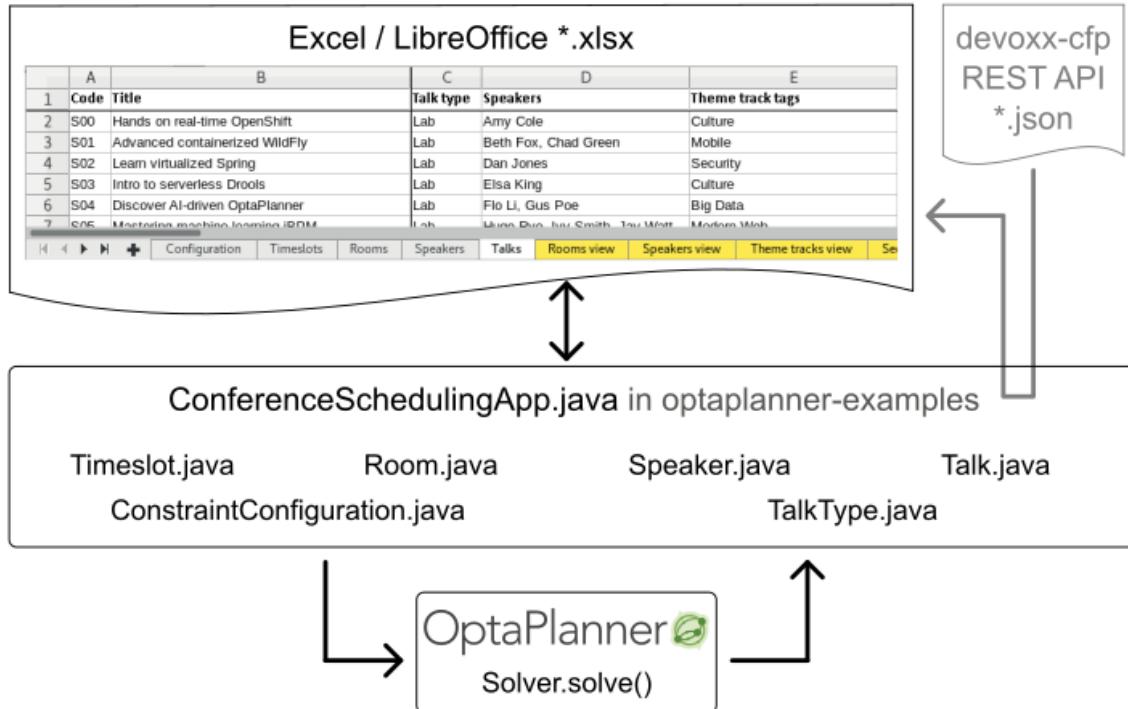
KIE OptaPlanner Deep Dive – Conference Scheduling

- **Business Scenario / Problem Description**
- **Assign each conference talk to a timeslot and a room, after the talks have been accepted.**
- **Hard constraints (unless configured otherwise):**
 - Talk type of timeslot: The type of a talk must match the timeslot's talk type.
 - Room unavailable timeslots: A talk's room must be available during the talk's timeslot.
 - Room conflict: Two talks can't use the same room during overlapping timeslots.
 - Speaker unavailable timeslots: Every talk's speaker must be available during the talk's timeslot.
 - Speaker conflict: Two talks can't share a speaker during overlapping timeslots.
 - Generic purpose timeslot and room tags
 - Speaker required timeslot tag: If a speaker has a required timeslot tag, then all his/her talks must be assigned to a timeslot with that tag.
 - Speaker prohibited timeslot tag: If a speaker has a prohibited timeslot tag, then all his/her talks cannot be assigned to a timeslot with that tag.
 - Talk required timeslot tag: If a talk has a required timeslot tag, then it must be assigned to a timeslot with that tag.
 - Talk prohibited timeslot tag: If a talk has a prohibited timeslot tag, then it cannot be assigned to a timeslot with that tag.
 - Speaker required room tag: If a speaker has a required room tag, then all his/her talks must be assigned to a room with that tag.
 - Speaker prohibited room tag: If a speaker has a prohibited room tag, then all his/her talks cannot be assigned to a room with that tag.
 - Talk required room tag: If a talk has a required room tag, then it must be assigned to a room with that tag.
 - Talk prohibited room tag: If a talk has a prohibited room tag, then it cannot be assigned to a room with that tag.
 - Talk prerequisite talks: A talk must be scheduled after all its prerequisite talks.
 - Consecutive talks pause: A speaker who has more than one talk must have a break between them.
 - Talk mutually-exclusive-talks tags: Talks that share such tags must not be scheduled in overlapping timeslots.
- **Medium constraints (unless configured otherwise):**
 - Published timeslot: A published talk must not be scheduled at a different timeslot than currently published. If a hard constraint's input data changes after publishing (such as speaker unavailability), then this medium constraint will be minimally broken to attain a new feasible solution.

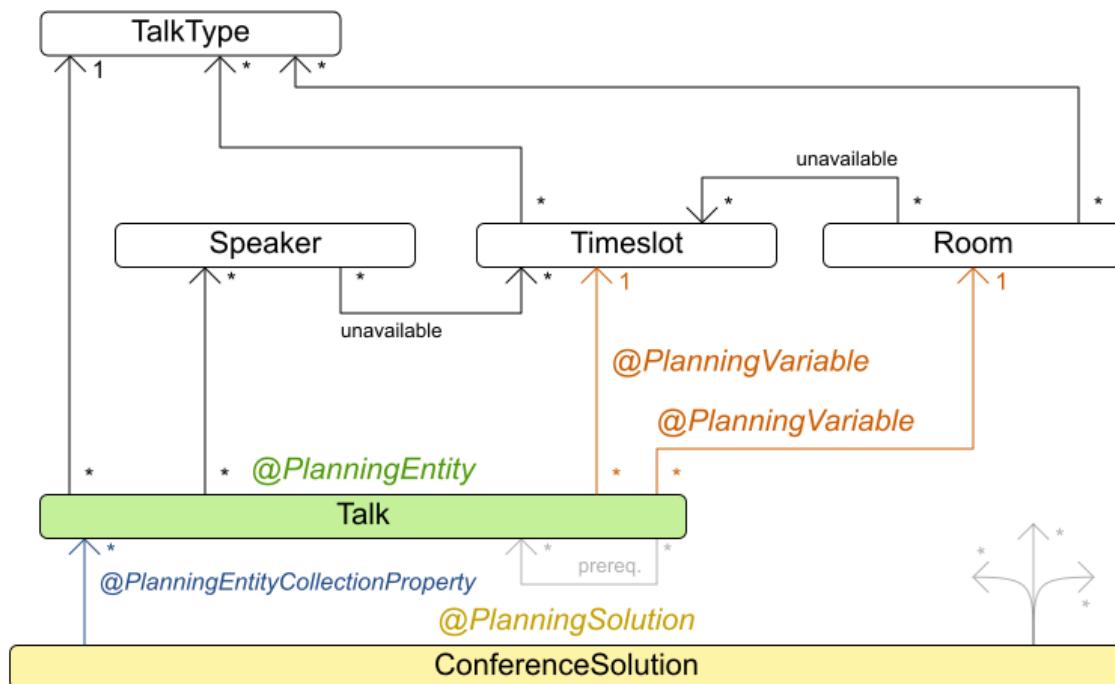
394

Conference scheduling architecture

Planner works with plain Java objects that are read/written to *.xlsx

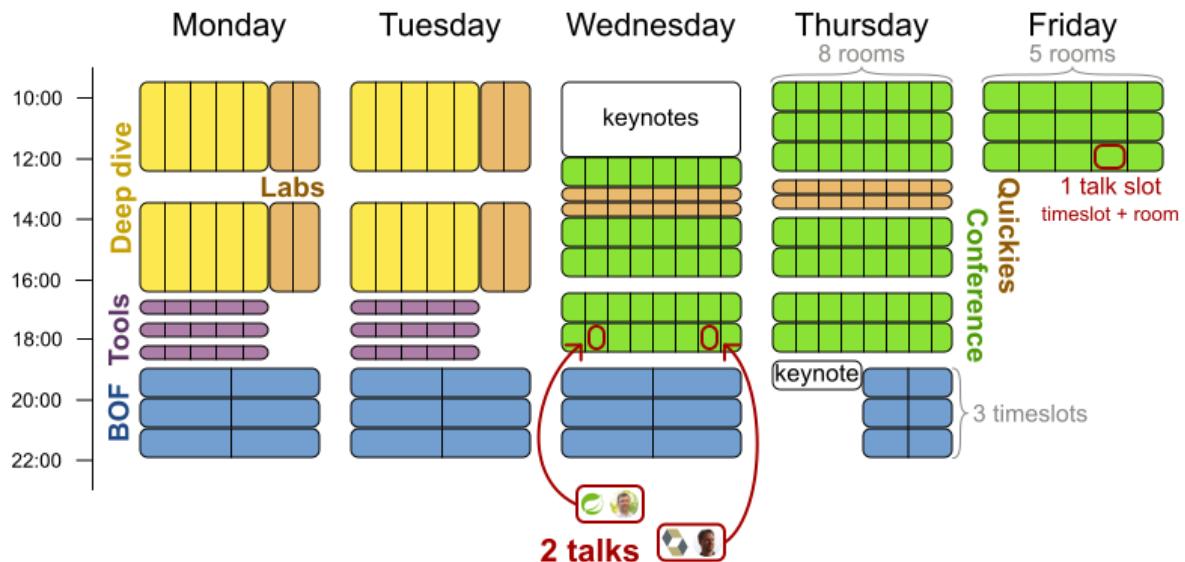


Conference scheduling class diagram



Conference scheduling problem

Assign each talk to a timeslot and a room.



Devoxx Belgium assigns 214 talks to 40 timeslots and 10 rooms for 3500 attendees.

The screenshot shows a software interface for solving a constraint satisfaction problem (CSP). The main window displays a table titled "Constraint matches" with the following data:

Constraint name	Match count	Score total
Audience level diversity	15	15soft
Audience type diversity	13	13soft
Crowd control	13	-130soft
Popular talks	22	-220soft
Room stability	17	-170soft
Speaker unavailable timeslot	1	-1hard
Theme track conflict	3	-30soft

Below the table, a message states: "[S11, ConferenceParametrization-0] = -1hard". The bottom status bar indicates "latest best score: -1hard/0medium/522soft".

A	B	C
1 Conference name	Javox 2021	
2		
3 Constraint	Weight	Description
4 Theme track conflict	10	Soft penalty per common theme track of 2 talks that have an overlapping timeslot
5 Sector conflict	10	Soft penalty per common sector of 2 talks that have an overlapping timeslot
6 Audience type diversity	1	Soft reward per 2 talks that have the same timeslot and a different audience type
7 Audience type theme track conflict	0	Soft penalty per 2 talks that have a common audience type, have a common theme track and have an overlapping timeslot
8 Audience level diversity	1	Soft reward per 2 talks that have the same timeslot and a different audience level
9 Audience level flow per content violation	10	Soft penalty per common content of 2 talks with a different audience level for which the easier talk isn't scheduled earlier than the other talk
10 Content conflict	10	Soft penalty per common content of 2 talks that have an overlapping timeslot
11 Language diversity	10	Soft reward per 2 talks that have the same timeslot and a different language
12 Speaker preferred timeslot tags	20	Soft penalty per missing preferred tag in a talk's timeslot
13 Speaker undesired timeslot tags	20	Soft penalty per undesired tag in a talk's timeslot
14 Talk preferred timeslot tags	20	Soft penalty per missing preferred tag in a talk's timeslot
15 Talk undesired timeslot tags	20	Soft penalty per undesired tag in a talk's timeslot
16 Speaker preferred room tags	20	Soft penalty per missing preferred tag in a talk's room
17 Speaker undesired room tags	20	Soft penalty per undesired tag in a talk's room
18 Talk preferred room tags	20	Soft penalty per missing preferred tag in a talk's room
19 Talk undesired room tags	20	Soft penalty per undesired tag in a talk's room
20 Same day talks	10	Soft penalty per common content/theme of 2 talks that are scheduled on different days
21 Popular talks	10	Soft penalty per 2 talks where the less popular one (has lower favorite count) is assigned a larger room than the more popular talk
22 Crowd control	10	Soft penalty per talk with crowd control risk greater than zero that are not in pairs
23 Published room	10	Soft penalty per talk scheduled at a different room than its published one
24 Room stability	10	Soft penalty per two talks with the same track scheduled in the same day but at different rooms
25		
26 Talk mutually-exclusive-talks tags	1	Medium penalty per two talks that share the same Mutually exclusive talks tag that are scheduled in overlapping timeslots
27 Published timeslot	10	Medium penalty per talk scheduled at a different timeslot than its published one
28		
29 Talk type of timeslot	10000	Hard penalty per talk in a timeslot with another talk type
30 Talk type of room	10000	Hard penalty per talk in a room with another talk type
31 Room unavailable timeslot	10000	Hard penalty per talk with an unavailable room at its timeslot
32 Room conflict	10	Hard penalty per pair of talks in the same room in overlapping timeslots
33 Speaker unavailable timeslot	1	Hard penalty per talk with an unavailable speaker at its timeslot
34 Speaker conflict	1	Hard penalty per pair of talks with the same speaker in overlapping timeslots
35 Speaker required timeslot tags	1	Hard penalty per missing required tag in a talk's timeslot
36 Speaker prohibited timeslot tags	1	Hard penalty per prohibited tag in a talk's timeslot
37 Talk required timeslot tags	1	Hard penalty per missing required tag in a talk's timeslot
38 Talk prohibited timeslot tags	1	Hard penalty per prohibited tag in a talk's timeslot
39 Speaker required room tags	1	Hard penalty per missing required tag in a talk's room
40 Speaker prohibited room tags	1	Hard penalty per prohibited tag in a talk's room
41 Talk required room tags	1	Hard penalty per missing required tag in a talk's room
42 Talk prohibited room tags	1	Hard penalty per prohibited tag in a talk's room
43 Talk prerequisite talks	1	Hard penalty per talk that is scheduled before any of its prerequisite talks

A	B	C	D	E	F	G
1	Mon 2018-10-01					
2 Room	10:15-12:15	10:15-11:00	11:30-12:15	13:00-15:00	15:30-16:15	16:30-17:15
3 R 1		S10: Prepare for streaming GWT Dan Jones	S13: Grok distributed Weld Hugo Rye		S09: Debug enterprise Hibernate Chad Green	S08: Securing scalable Docker Beth Fox
4 R 2		S16: Deliver stable Tensorflow Amy Fox	S17: Implement platform-independent VertX Beth Green		S04: Discover AI-driven OptaPlanner Gus Poe, Hugo Rye	S14: Troubleshooting reliable RestEasy Ivy Smith
5 R 3		S05: Mastering machine learning BPM Ivy Smith	S11: Understand mobile Errai Elsa King, Flo Li		S03: Intro to serverless Drools Flo Li	S07: Building deep learning XStream Amy Fox, Beth Green, Amy Cole
6 R 4		S12: Applying modern Angular Gus Poe	S15: Using secure Android Jay Watt		S02: Learn virtualized Spring Elsa King	S06: Tuning IOT-driven Camel Jay Watt
7 R 5	S00: Hands on real-time OpenShift Amy Cole, Beth Fox		S01: Advanced containerized WildFly Chad Green, Dan Jones			
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Rock Tour



- **Business Scenario / Problem Description**

- Drive the rock bus from rock-show to rock-show, but schedule rock-shows only on available days.

- **Hard constraints:**

- Schedule every required show.
- Schedule as many shows as possible.

- **Medium constraints:**

- Maximize revenue opportunity.
- Minimize driving time.
- Visit sooner than later.

- **Soft constraints:**

- Avoid long driving times.

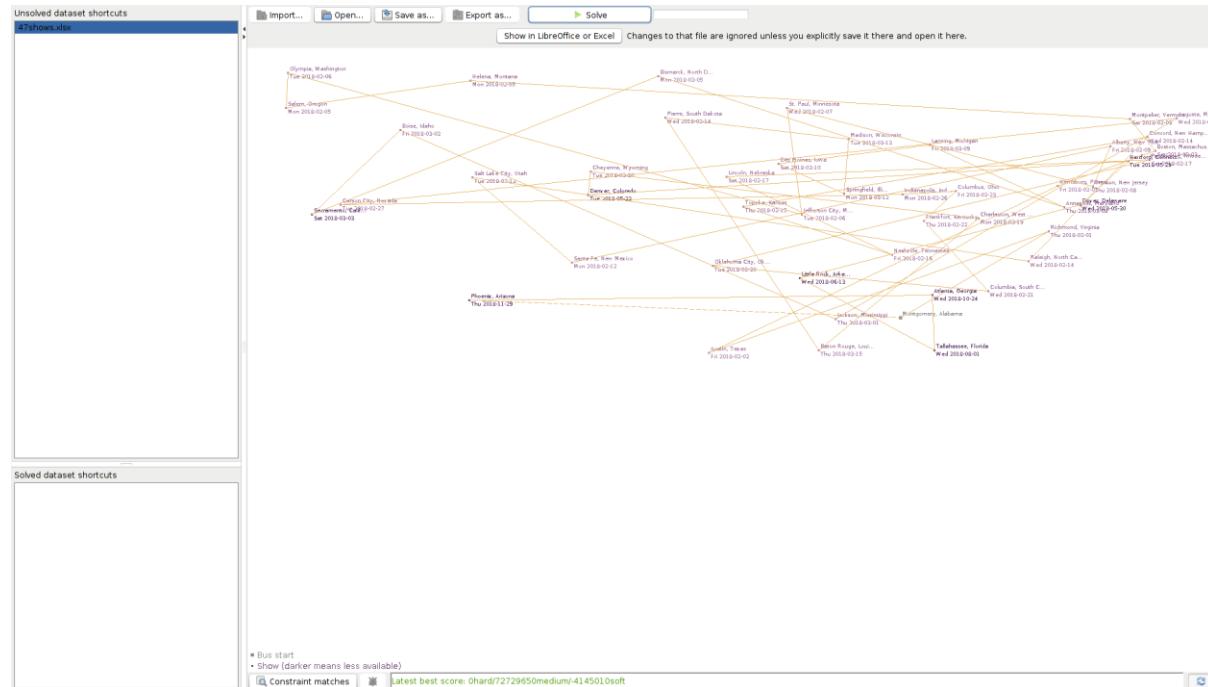
409

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram



410



	Date	Venue name	City name	Driving time	Driving time per week	Latitude	Longitude	Duration (in days)	Revenue opportunity	Required	Available dates size
1	Thu 2018-02-01	Richmond, Virginia	Richmond, Virginia	0 hours 6 minutes		37.538857	-77.43364	0.5	2400000	0	224
2	Fri 2018-02-02	Austin, Texas	Austin, Texas	0 hours 2 minutes		30.27467	-97.740349	0.5	200000	0	256
3		Harrisburg, Pennsylvania	Harrisburg, Pennsylvania	0 hours 0 minutes		40.264378	-76.883598	0.5	100000	0	252
4	Sat 2018-02-03	Boston, Massachusetts	Boston, Massachusetts	0 hours 1 minutes		42.358162	-71.063698	0.5	600000	0	225
5		Montpelier, Vermont	Montpelier, Vermont	0 hours 0 minutes		44.262436	-72.580536	0.5	100000	0	209
6	Sun 2018-02-04				0 hours 12 minutes						
7	Mon 2018-02-05	Helena, Montana	Helena, Montana	0 hours 1 minutes		46.585709	-112.018417	0.5	1300000	0	215
8		Salem, Oregon	Salem, Oregon	0 hours 0 minutes		44.938461	-123.030403	0.5	1200000	0	231
9	Tue 2018-02-06	Olympia, Washington	Olympia, Washington	0 hours 0 minutes		47.035805	-122.905014	0.5	1800000	0	222
10		Jefferson City, Missouri	Jefferson City, Missouri	0 hours 3 minutes		38.579201	-92.172935	0.5	1800000	0	235
11	Wed 2018-02-07	St. Paul, Minnesota	St. Paul, Minnesota	0 hours 1 minutes		44.955097	-93.102211	1	600000	0	248
12	Thu 2018-02-08	Trenton, New Jersey	Trenton, New Jersey	0 hours 3 minutes		40.220596	-74.769913	0.5	1100000	0	227
13	Fri 2018-	Albany, New York	Albany, New York	0 hours 0 minutes							

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

KIE OptaPlanner Deep Dive – Flight Crew Scheduling

- **Business Scenario / Problem Description**

- Assign flights to pilots and flight attendants.

- **Hard constraints:**

- Required skill: each flight assignment has a required skill. For example, flight AB0001 requires 2 pilots and 3 flight attendants.
- Flight conflict: each employee can only attend one flight at the same time
- Transfer between two flights: between two flights, an employee must be able to transfer from the arrival airport to the departure airport. For example, Ann arrives in Brussels at 10:00 and departs in Amsterdam at 15:00.
- Employee unavailability: the employee must be available on the day of the flight. For example, Ann is on PTO on 1-Feb.

- **Soft constraints:**

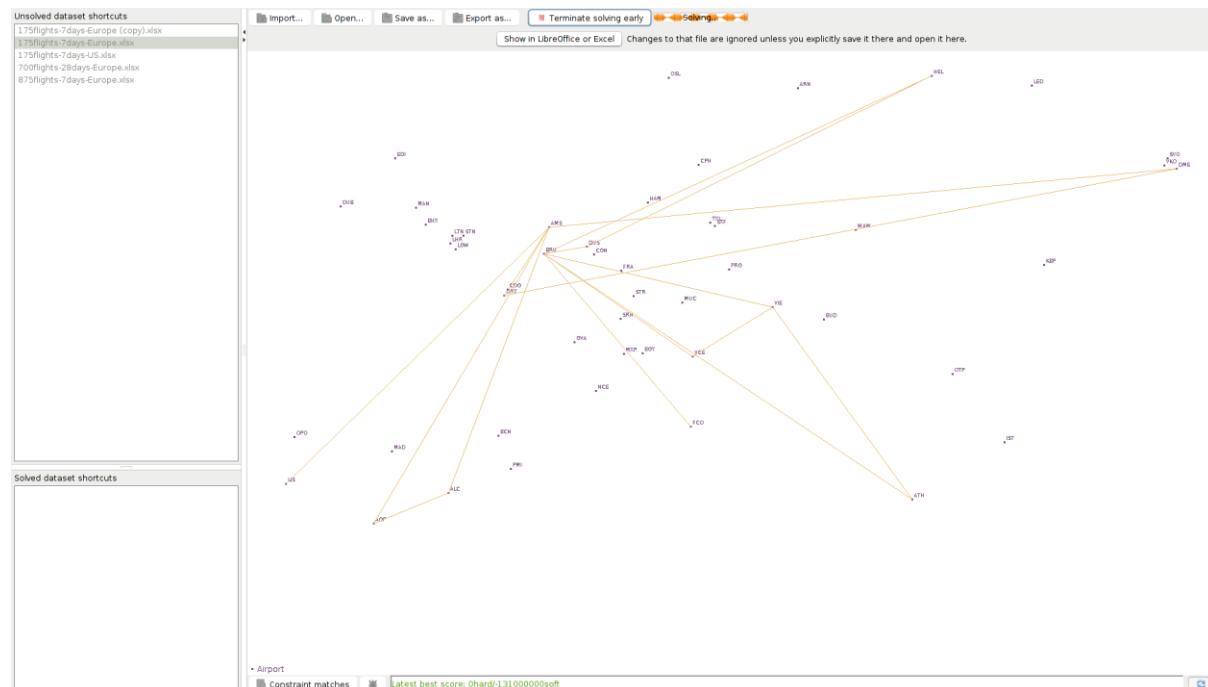
- First assignment departing from home
- Last assignment arriving at home
- Load balance flight duration total per employee

420

ANNEX 4 : KIE OPTAPLANNER EXAMPLES

Exercise: Draw Class Diagram

421



A	B	C	D	E	F	G	H
Flight number	Departure airport code	Departure UTC date time	Arrival airport code	Arrival UTC date time	Employee skill requirements	Employee assignments	
2	AB003	BRU	2018-01-01 07:08	ATH	2018-01-01 10:44	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Cole, Flo T. Li, Chad Q. Green, Dan R. Jones, Elsa S. Li
3	AB008	BRU	2018-01-01 09:01	VIE	2018-01-01 11:26	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Rye, Amy O. Green, Hugo V. Watt, Ivy W. Cole, Jay X. Fox
4	AB021	BRU	2018-01-01 09:36	FCO	2018-01-01 12:16	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Rye, Amy O. Green, Hugo V. Watt, Ivy W. Cole, Jay X. Fox
5	AB018	BRU	2018-01-01 11:40	DUS	2018-01-01 13:19	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Smith, Flo T. Smith, Chad Q. Li, Elsa S. Rye, Ivy W. Fox
6	AB007	BRU	2018-01-01 16:27	VCE	2018-01-01 18:47	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Fox, Beth P. Green, Elsa S. King, Hugo V. Rye, Jay X. Watt
7	AB001	AMS	2018-01-01 06:39	AGP	2018-01-01 10:02	Pilot, Flight attendant, Flight attendant, Flight attendant	Flo T. Poe, Beth P. Jones, Chad Q. Jones, Dan R. King, Hugo V. Smith
8	AB023	AMS	2018-01-01 06:55	AGP	2018-01-01 10:18	Pilot, Flight attendant, Flight attendant, Flight attendant	Flo T. Rye, Amy O. Jones, Elsa S. Poe, Dan R. Poe, Hugo V. Cole
9	AB011	AMS	2018-01-01 07:02	LIS	2018-01-01 10:22	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. King, Gus U. Watt, Jay X. Green, Chad Q. Rye, Elsa S. Watt
10	AB005	AMS	2018-01-01 11:22	DME	2018-01-01 15:02	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Cole, Flo T. Cole, Hugo V. Green, Jay X. King, Dan R. Smith
11	AB013	AMS	2018-01-01 13:54	DME	2018-01-01 17:34	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Poe, Beth P. Jones, Chad Q. Jones, Dan R. King, Hugo V. Smith
12	AB022	FCO	2018-01-01 17:45	BRU	2018-01-01 19:59	Pilot, Flight attendant, Flight attendant, Flight attendant	Flo T. Rye, Amy O. Jones, Elsa S. Poe, Dan R. Poe, Hugo V. Cole
13	AB015	ORY	2018-01-01 16:59	AMS	2018-01-01 18:45	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. King, Gus U. Watt, Jay X. Green, Chad Q. Rye, Elsa S. Watt
14	AB017	DME	2018-01-01 10:57	AMS	2018-01-01 14:37	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Cole, Flo T. Cole, Hugo V. Green, Jay X. King, Dan R. Smith
15	AB014	DME	2018-01-01 13:56	ORY	2018-01-01 17:56	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Poe, Flo U. Watt, Ivy W. Li, Chad Q. Cole, Ivy W. Poe
16	AB012	LIS	2018-01-01 09:12	AMS	2018-01-01 12:32	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Cole, Amy O. Li, Ivy W. Green, Jay X. Jones, Dan R. Smith
17	AB019	DUS	2018-01-01 14:45	HEL	2018-01-01 17:45	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. King, Flo T. Cole, Hugo V. Green, Jay X. King, Dan R. Smith
18	AB009	VIE	2018-01-01 06:17	VCE	2018-01-01 08:12	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Poe, Gus U. Green, Hugo V. Green, Jay X. King, Dan R. Watt
19	AB005	VIE	2018-01-01 12:53	BRU	2018-01-01 15:18	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Poe, Amy O. Fox, Dan R. Li, Chad Q. King, Jay X. Cole
20	AB004	ATH	2018-01-01 07:26	VIE	2018-01-01 10:12	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Rye, Beth P. Smith, Elsa S. Cole, Ivy W. King, Jay X. Li
21	AB020	HEL	2018-01-01 10:55	BRU	2018-01-01 14:03	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Li, Flo U. Poe, Hugo V. Watt, Ivy W. Cole, Jay X. Fox
22	AB002	AGP	2018-01-01 08:49	AMS	2018-01-01 12:12	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Watt, Gus U. King, Chad R. Rye, Ivy W. Rye, Hugo V. Jones
23	AB024	AGP	2018-01-01 17:44	ALC	2018-01-01 19:37	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Fox, Beth P. Green, Elsa S. King, Hugo V. Rye, Jay X. Watt
24	AB018	ALC	2018-01-01 14:44	AMS	2018-01-01 17:50	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. King, Flo T. Cole, Ivy W. Jones, Dan S. King, Chad Q. Watt
25	AB007	VCE	2018-01-01 09:14	BRU	2018-01-01 10:29	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Rye, Amy O. Green, Hugo V. Green, Jay X. King, Dan R. Smith
26	AB010	ORY	2018-01-01 10:45	BRU	2018-01-01 13:45	Pilot, Flight attendant, Flight attendant, Flight attendant	Jay X. King, Flo T. Cole, Hugo V. Green, Jay X. King, Dan R. Watt
27	AB003	BRU	2018-01-02 07:09	ATH	2018-01-02 10:44	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Green, Gus U. Poe, Ivy W. Watt, Jay X. Cole, Chad Q. King
28	AB008	BRU	2018-01-02 09:01	VIE	2018-01-02 11:26	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Fox, Gus U. Rye, Dan R. Li, Hugo V. Watt, Ivy W. Cole
29	AB021	BRU	2018-01-02 09:36	FCO	2018-01-02 12:16	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Green, Beth P. Poe, Jay X. Fox, Hugo V. Green, Jay X. King
30	AB018	BRU	2018-01-02 11:40	DUS	2018-01-02 13:19	Pilot, Flight attendant, Flight attendant, Flight attendant	Flo T. Green, Gus U. Jones, Chad Q. Smith, Dan R. Watt, Jay X. Poe
31	AB006	BRU	2018-01-02 16:27	VCE	2018-01-02 18:47	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Smith, Flo T. Jones, Dan R. Poe, Elsa S. Green, Hugo V. Li
32	AB001	AMS	2018-01-02 06:39	AGP	2018-01-02 10:02	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Fox, Amy O. Poe, Ivy W. Smith, Chad Q. Poe, Dan R. Cole
33	AB023	AMS	2018-01-02 06:55	AGP	2018-01-02 10:18	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Rye, Flo T. Fox, Elsa S. Fox, Hugo V. King, Ivy W. Li
34	AB011	AMS	2018-01-02 07:02	LIS	2018-01-02 10:22	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Watt, Gus U. King, Chad Q. Cole, Ivy W. Poe, Jay X. Rye
35	AB005	AMS	2018-01-02 11:22	DME	2018-01-02 15:02	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Rye, Beth P. Smith, Elsa S. Cole, Ivy W. King, Dan S. King
36	AB013	AMS	2018-01-02 13:54	DME	2018-01-02 17:34	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Watt, Gus U. King, Chad Q. Cole, Ivy W. Poe, Jay X. Rye
37	AB022	FCO	2018-01-02 17:16	BRU	2018-01-02 19:56	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. Green, Beth P. Poe, Jay X. Fox, Hugo V. Green, Jay X. King
38	AB015	ORY	2018-01-02 16:50	AMS	2018-01-02 20:45	Pilot, Flight attendant, Flight attendant, Flight attendant	Gus U. Rye, Amy O. Green, Hugo V. Green, Jay X. King, Dan R. Smith
39	AB017	DME	2018-01-02 18:57	AMS	2018-01-02 21:29	Pilot, Flight attendant, Flight attendant, Flight attendant	Beth P. Watt, Gus U. King, Chad Q. Cole, Ivy W. Poe, Jay X. Rye
40	AB014	DME	2018-01-02 13:57	ORY	2018-01-02 17:56	Pilot, Flight attendant, Flight attendant, Flight attendant	Amy O. King, Flo T. Cole, Hugo V. Green, Jay X. King, Dan R. Watt
41	AB012	LIS	2018-01-02 09:12	AMS	2018-01-02 12:32	Pilot, Flight attendant, Flight attendant, Flight attendant	Flo T. Rye, Amy O. Jones, Elsa S. Poe, Dan R. Poe, Hugo V. Cole
42	AB019	DUS	2018-01-02 14:45	HEL	2018-01-02 17:45	Pilot, Flight attendant, Flight attendant, Flight attendant	Flo T. Green, Gus U. Jones, Chad Q. Smith, Dan R. Watt, Jay X. Poe

The End of Workshop Project Guide