

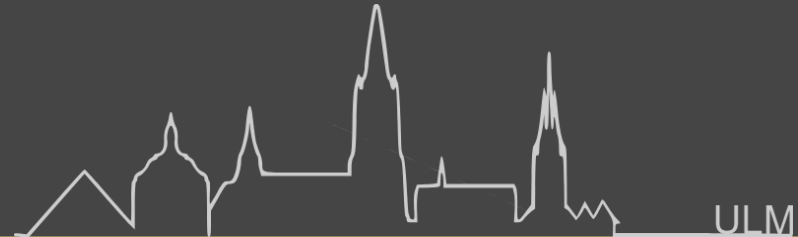


JOpt.TourOptimizer

Intelligent Tour and Resource Optimization
for TMS & WMS Platforms

DNA Evolutions GmbH · www.dna-evolutions.com · October 2026

*Authorized Managing Director:
Dr.-Ing. Dipl.-Phys. Jens Richter*



JOPT.TOUROPTIMIZER

The optimization engine that respects every constraint.



The planning brain inside TMS, WMS and field-service platforms. Your product owns the user experience; JOpt computes feasible, optimized plans.

0

hard-constraint violations, by design

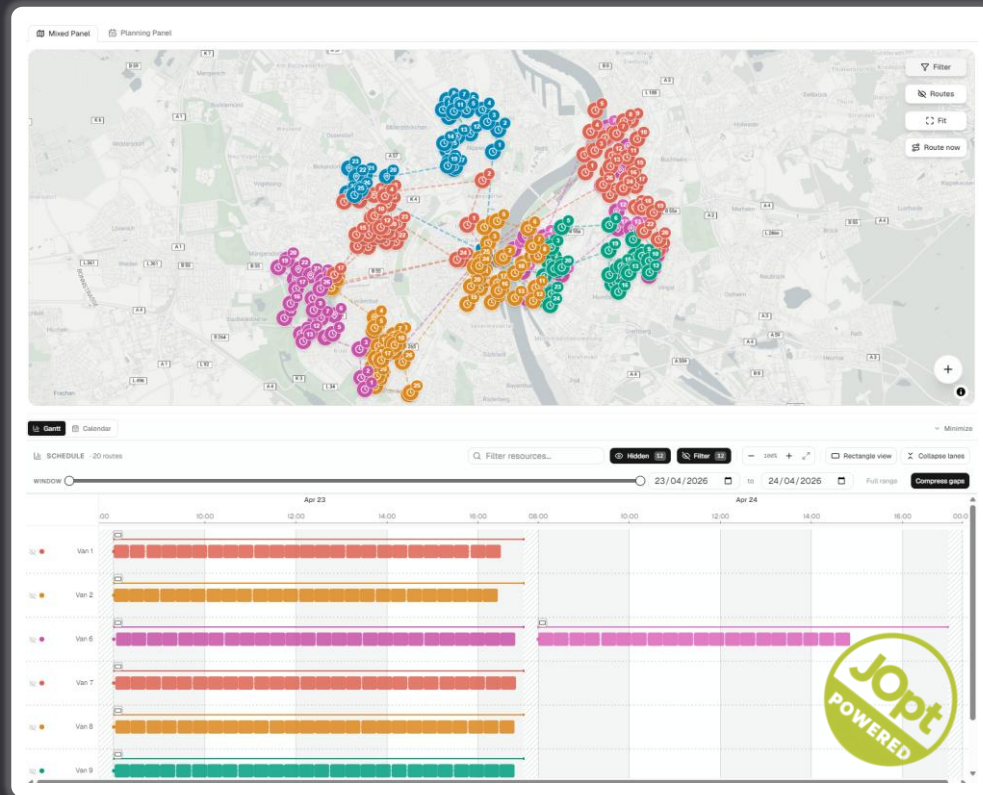
14

capability categories

10 min

to a first optimization

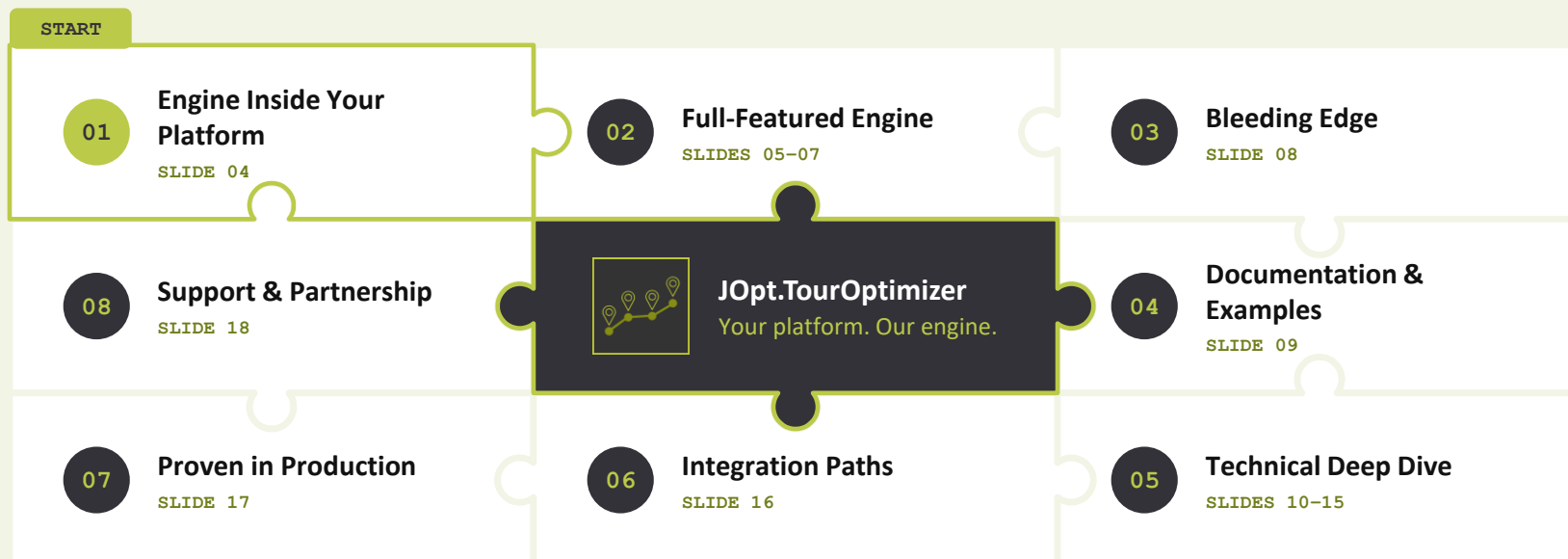
Your platform. Our engine.



Sample problem: 20 optimized routes over two days, map and Gantt view

A Complete Product, Not Just an Algorithm

Eight sections around one engine. Start at 01, then follow clockwise.

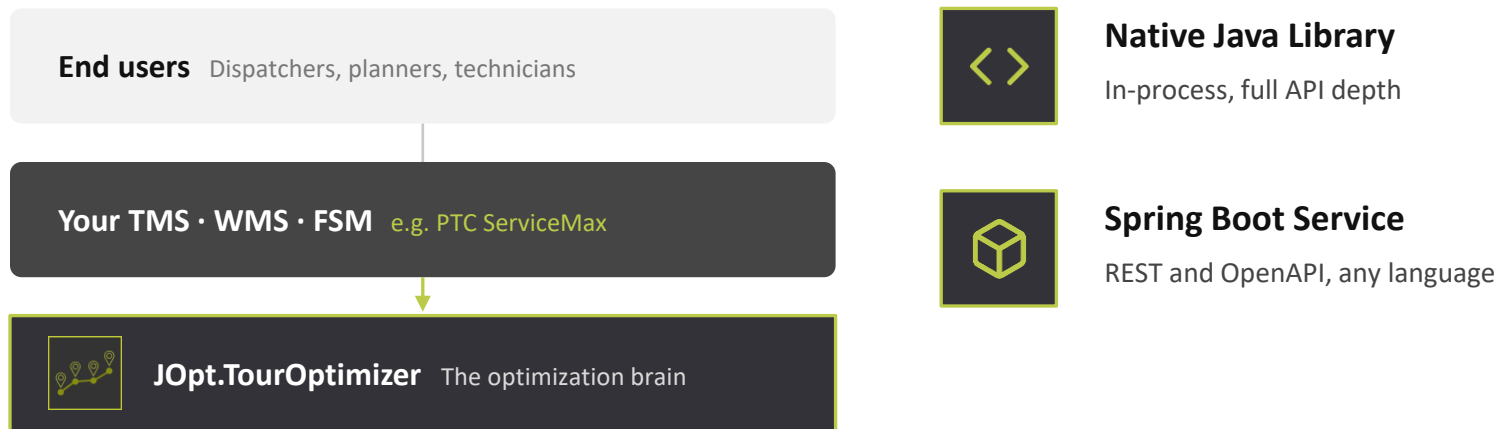


Appendix: slides 19–21

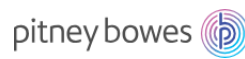
 Footer marker: current section

The Engine Inside Your Platform

Your product owns the experience. JOpt computes the plans.



TRUSTED BY



Shortest Path Is Only the Beginning



The Challenge

Real constraints multiply complexity.



Time



Capacity



Skills



Depots



Fleets



The Solution

An embeddable engine that treats constraints as hard requirements, not preferences.

12+

constraint types per scenario

3×10^{64}

sequences for 50 stops

0

hard-constraint violations



One Engine, Nearly Any Planning Scenario

Combine building blocks freely.



Time windows



Multiple depots



Mixed fleets



Skills



Zones & territories



Pickup & delivery



Split delivery



Dynamic production



Guaranteed SLAs



Multi-day



CO₂ objectives



Custom rules

USED IN



Logistics



Field service



Manufacturing



Healthcare



What Other Optimizers Can't Do



Pillar (Captured) Nodes

SLAs guaranteed by architecture, not penalties.



Pickup & Delivery

Goods tracked in every vehicle, at all times.



Dynamic Manufacturing

The optimizer decides production quantities.



Split Delivery

Orders split across vehicles or days, built in.



AutoFilter

Keeps large, over-constrained runs converging.



Open Assessor

Your business rules inside the solver.



Warm Start

Re-optimize incrementally, resume plans.



Quantum Plugin

D-Wave annealing for TSP construction.



Cutting Edge: AI, Learning and Quantum



AI on Input and Output

AI builds the input model and explains the result.



JOpt AI Assistant

Free GPT that writes JOpt code. MCP coming soon.



Reinforcement Learning

Agents learn the best weights and operators.



Quantum Construction

D-Wave annealing solves TSP subproblems.

JOpt TourOptimizer Assistant ▾



JOpt TourOptimizer Assistant

By Jens Richter 𐄂

✓ Using the creator's recommended model: GPT-5.4 Thinking

Expert coding assistant for JOpt.TourOptimizer by DNA Evolutions — helps with Java SDK, REST API integration, route optimization modeling, and real-world constraint configuration.

How do I run JOpt TourOptimizer as a REST service with...

How do I set up my first optimization with JOpt in Java?

I need to model bittypes skill-based routing - which...

Can you help me with a Pickup & Delivery scenario?

Ask anything

+ ⌚ Thinking ▾



JOpt AI Assistant on the OpenAI GPT Store



Documentation Built for Engineers

From a first optimization in the browser to Kubernetes.



Getting Started

Quickstarts, sandboxes, AI assistant



Learn & Explore

Feature guides, examples by depth



Integrate

REST, OpenAPI, Kubernetes, Terraform

100+

working examples

20+

feature guides

4

browser sandboxes

8

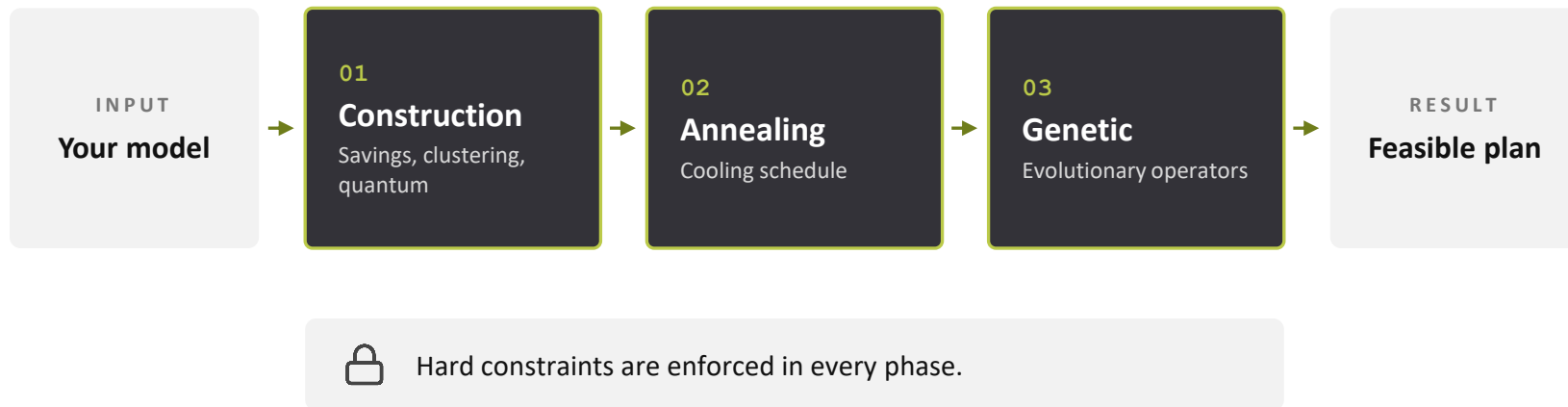
feature spotlights

dna-evolutions.com/docs



Inside the Optimization Engine

Fully modular: reorder or swap phases, or plug in your own algorithms.



Hard Constraints by Architecture

SLAs, skills, zones and more are hard by architecture. Soft constraints optimize within.

PENALTY-BASED

✗ Can be violated

✗ Competes with soft goals

✗ Degrades with scale

✗ No guarantee

JOPT CAPTURED NODE

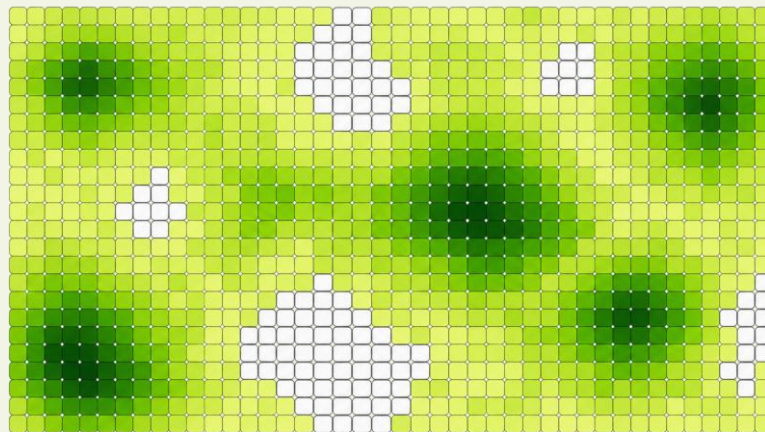
✓ **Cannot be violated**

✓ **Separate layer**

✓ **Scales with complexity**

✓ **Every solution valid**

Invalid solutions are cut from the search space.



Captured Nodes

ONLY IN JOPT

An SLA time window that can never be violated. Other work flows around it.

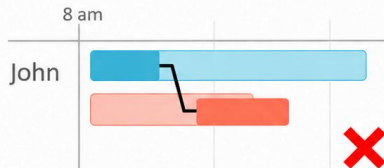
- 01 Reorder normal nodes
- 02 Accept soft lateness elsewhere
- 03 Reassign to another resource
- 04 Cancel rather than violate



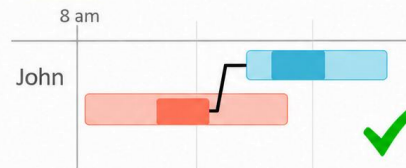
Other optimizers weigh SLA violations against cost. JOpt rules them out.

01 Reorder around the pillar

Before

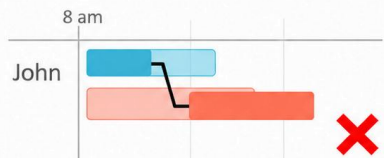


After



02 Reassign when needed

Before



After



Normal node

Pale bar = opening window
Dark bar = actual visit duration



Captured node / Pillar

Pale bar = opening window
Dark bar = actual visit duration



Zones & Territories

Control which resource may serve which node, per shift. Hard or soft.



Per-shift territories

Zones 1 and 2 on Monday, zone 3 on Tuesday. Just a data change.



Multiple assignments

One shift can cover several zones. Border nodes can belong to several zones.



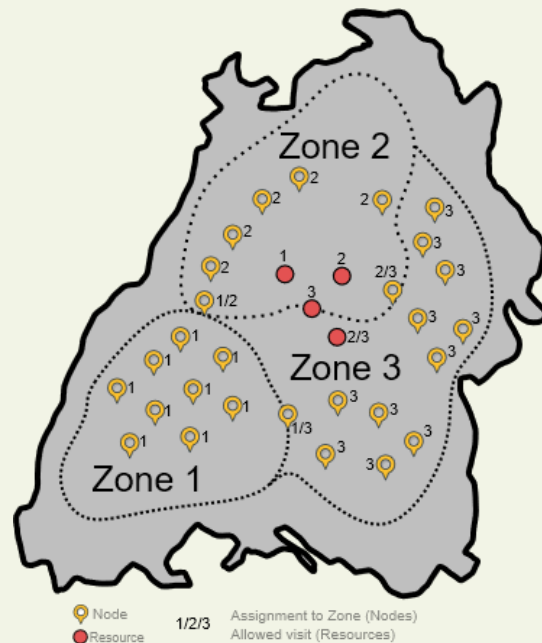
Hard or soft

Hard for contracts and regulation, soft with a tunable penalty.



Zone crossing

Penalize bridge and tunnel crossings between zones.



Zones are abstract: districts, postcodes, customer segments or contract types.

Dynamic Manufacturing & Split Delivery

Production quantities are planning variables the optimizer controls.

Bakery example: the optimizer decides that Aachen produces 10 trays and Cologne 17.

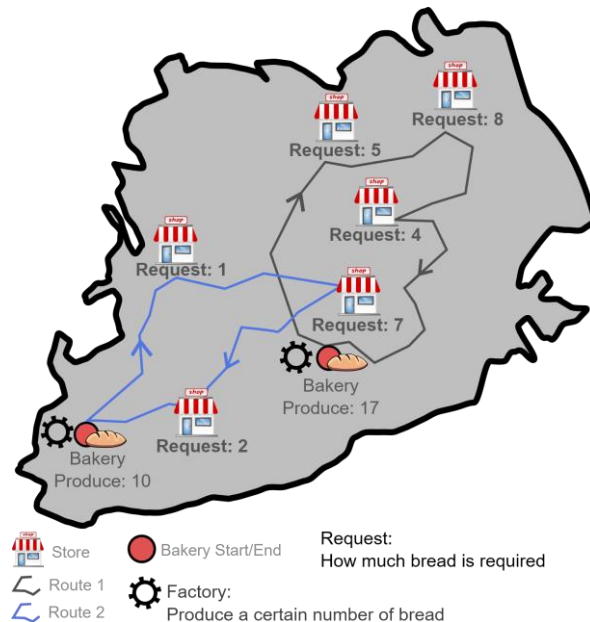
STANDARD VRP ENGINES

- ✗ Supply quantities fixed before solving
- ✗ Production planned separately
- ✗ No dynamic goods creation
- ✗ Split delivery modeled by hand
- ✗ Static demand per node

JOPT WITH FLEXLOADS

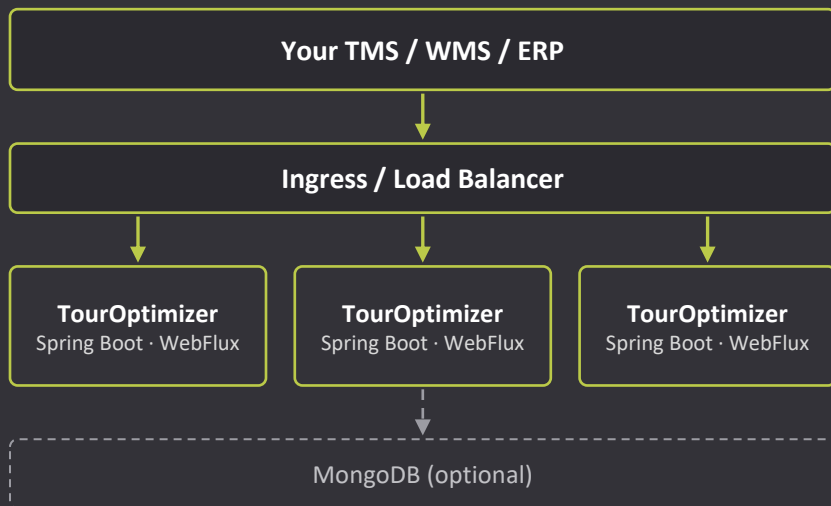
- ✓ Supply quantities optimized during solving
- ✓ Production and routing solved jointly
- ✓ Optimizer creates goods at supply nodes
- ✓ Built-in split delivery optimization
- ✓ Flexible, coupled and timed loads

To our knowledge, unique among route optimization engines.



Service Architecture & API

Stateless Spring Boot containers, cloud or on premises.



POST /api/v1/jobs

Async jobs

POST /api/v1/runs

Sync runs, SSE streams

GET /v3/api-docs

OpenAPI 3 contract

JSON snapshots

Portable state

From Evaluation to Production



01 Evaluate

Sandbox in 10 minutes



02 Model

Add your real constraints



03 Integrate

Native Java or Spring Boot



04 Scale

Kubernetes, Terraform

```
docker run -d --rm -p 8081:8081 \  
-e SPRING_PROFILES_ACTIVE=cors \  
dnaevolutions/jopt_touroptimizer:latest
```

One command to a running REST service.

Swagger UI at localhost:8081. Sandbox quickstart in Appendix B.



Proven Across Sectors

PTC • Schneider Electric • ServiceMax

Field service & Work Force

Kaleris • CAMS

Grocery delivery & Terminals & yards

OfficeTrack

Logistics & Telecommunication & Mining

Amazon Fresh

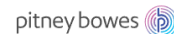
Grocery delivery

Pitney Bowes

Shipping & logistics

Henry Schein

Healthcare distribution



A Build vs Buy Decision, Already Solved

Faster time to market and a stronger product, without years of R&D.



www.dna-evolutions.com

dna-evolutions.com/contact



Documentation & API

dna-evolutions.com/docs



APPENDIX

Reference Material

A

The Full Feature Landscape

20

B

Start Right Away: JOpt Sandboxes

21

The Full Feature Landscape



Time & Schedule

- Multi-day, overnight stays
- Positive & negative FlexTime
- Event nodes for non-geo tasks



Hard Rules & Territories

- Pillar nodes: guaranteed SLAs
- Relations: same route, visitor
- Zone codes, bridge/tunnel rules



Skills & Resources

- Mandatory, preferred, banned
- Expertise levels, cost models
- Qualifications per shift



Pickup & Delivery

- Goods-tracking feasibility
- Simple, timed, flexible loads
- Dynamic production, split delivery



Objectives & Performance

- Distance, time, lateness
- CO₂ and workload balance
- Performance Mode, up to 1.7×



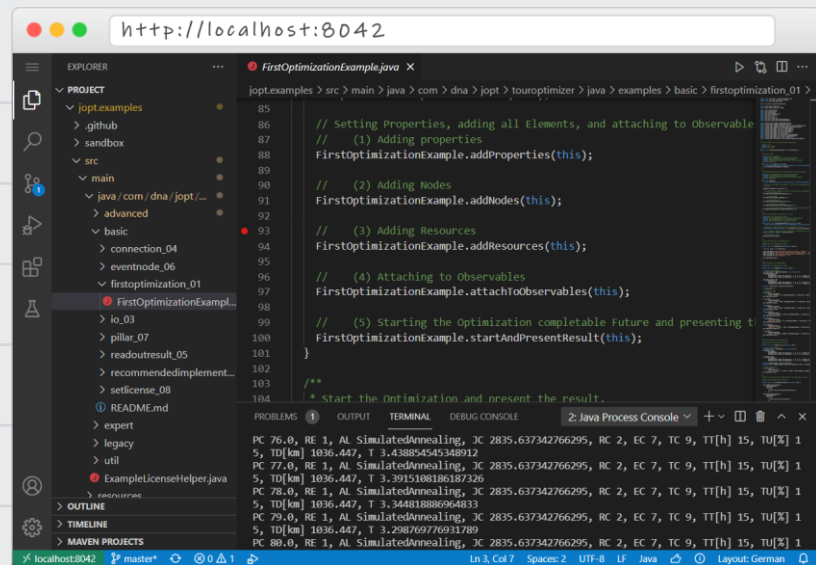
Continuity & Observability

- Snapshots and warm start
- Fire-and-forget with MongoDB
- Live progress streams

Start Right Away: JOpt Sandboxes

Only Docker and a browser are needed.

SANDBOX	CONTAINER	OPEN
Java SDK	jopt_example_server	localhost:8042
Java REST	jopt_rest_example_server	localhost:8043
Python REST	jopt_py_example_server	localhost:8033
C# / .NET REST	jopt_net_example_server	localhost:8023
REST endpoint	jopt_touroptimizer	localhost:8081



Logins: jopt (Java SDK, Python), joptrest (Java REST, C#).