Gurobi Webinar

Choosing a Math Programming Solver

What to look for and when to make a change

Alison Cozad
Optimization Engineer

March 2022

GUROBI OPTIMIZATION

Alison Cozad

Optimization Engineer at Gurobi

Ph.D. in Chemical Engineering, Optimization
Carnegie Mellon University

Worked in various optimization, real-time optimization, and data science roles in Oil & Gas

GUROBI OPTIMIZATION

What do I mean by optimization?

Also known as… Mathematical programming, Decision Science, Decision Optimization, Decision Analytics, Prescription analytics, Operations Research

Decision making process

Mapped to your "choosing a solver" decision

GUROBI OPTIMIZATION

Based on the Seven Steps to a Decision Making Process from UMass Dartmouth

Decision making process

Mapped to your "choosing a solver" decision

GUROBI OPTIMIZATION

most of our discussion

1. IDENTIFY THE DECISION
2. GATHER INFORMATION
3. IDENTIFY ALTERNATIVES
4. WEIGHT THE EVIDENCE
5. CHOOSE AMONG ALTERNATIVES
6. TAKE ACTION
7. REVIEW YOUR DECISION

What to look for in a solver?

Which fit my business problem?

Based on the Seven Steps to a Decision Making Process from UMass Dartmouth

What is an optimization solver?

Some general solver types

Commercial Solver

Both licensing and maintenance fees, offer improved performance, more polished interface, and enhanced support

Open-Source Solvers

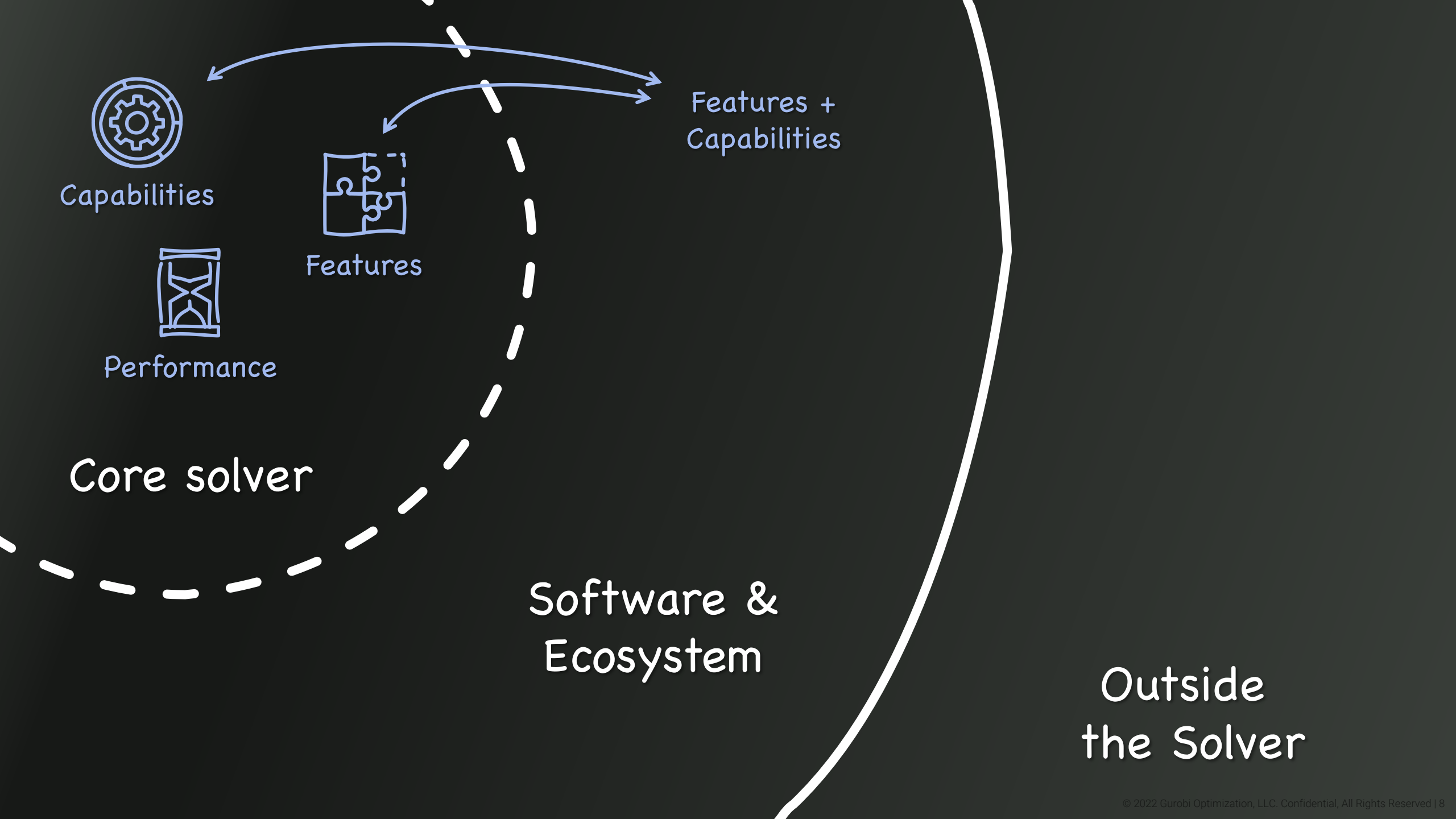
Quick and easy to get started with a license, but there's a tremendous performance difference

Modeling Systems & Frameworks

Framework to hot swap solvers with your model to easily switch solvers

Heuristics

Tailored 'smart strategies' that are sometimes considered an alternative to a solver



Pick a solver that fits your model type

GUROBI OPTIMIZATION

Optimization models can be classified by their properties:

- Continuous vs. Discrete
- Unconstrainted vs. Constrained
- Linear vs. Quadratic vs. Nonlinear
- Convex vs. Nonconvex
- Number of objective functions
- Deterministic vs. Stochastic

In this presentation, we will focus on **Mixed-Integer Optimization.**

For a more granular optimization tree, see the Neos' Optimization Taxonomy

Mixed integer optimization problems

Telecommunications
Facility Location Planning, Customer Management

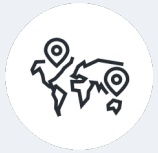
Healthcare
Staff scheduling. Hospital capacity planning

Electric Power
Optimal grid design, network modeling, outage planning

Financial services
Investment portfolio optimization, Fraud cost reduction

Agribusiness
Inventory management, Product Portfolio Optimization

Oil, gas, and chemicals
Planning, scheduling, blending, real-time optimization



Supply Chain
Network design, production planning, transportation selection

Manufacturing
Spare parts planning, Job shop scheduling, shift assignments

Packaged Goods
Inventory optimization, Demand planning, Production planning

Core Solver Capabilities & Features

There are huge differences between solvers when it comes to product features.

01

SUPPORTED MODEL TYPES

Can it handle the type of model you are trying to solve?

INTERFACES

PRODUCTIVITY

MODEL ANALYSIS

SOLVER TUNING

Gurobi solves the broadest range of problems – regardless of size or type.

MIQP	**MILP**	**QP**
LP	**QCP**	**Non-Convex MIQCP**
SOCP	**Convex MIQCP**	**Bi-linear**

Core Solver Capabilities & Features

There are huge differences between solvers when it comes to product features.

MODEL TYPES

02

INTERFACES (API)

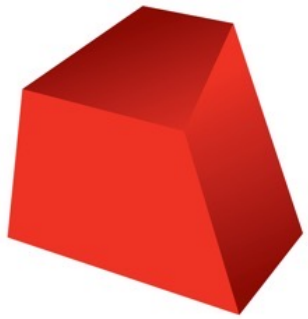
Is the solver available in the application or language that your team is already proficient in?

PRODUCTIVITY

MODEL ANALYSIS

SOLVER TUNING

Gurobi incorporates easily into your application



Gurobi Optimizer

Lightweight APIs

Your Application

Core Solver Capabilities & Features

There are huge differences between solvers when it comes to product features.

MODEL TYPES

INTERFACES

03

PRODUCTIVITY TOOLS

What tools are available to skip tedious or error prone modeling tasks?

MODEL ANALYSIS

SOLVER TUNING

Gurobi Productivity tools

are built into the solver to make modeling more efficient, easy, and effective

Multi-objective models

Many real-life decision problems have multiple conflicting objectives

Automatic modeling of logical conditions

Automatic linearization of nonlinear functions

Automatic approximation of non-linear functions

Core Solver Capabilities & Features

There are huge differences between solvers when it comes to product features.

MODEL TYPES

INTERFACES

PRODUCTIVITY

04

MODEL ANALYSIS TOOLS

What is available to understand solutions, mitigate risk, and cope with infeasibilities?

SOLVER TUNING

Gurobi's Model Analysis Tools

Help you to understand your solution, mitigate risks, and identify infeasibility

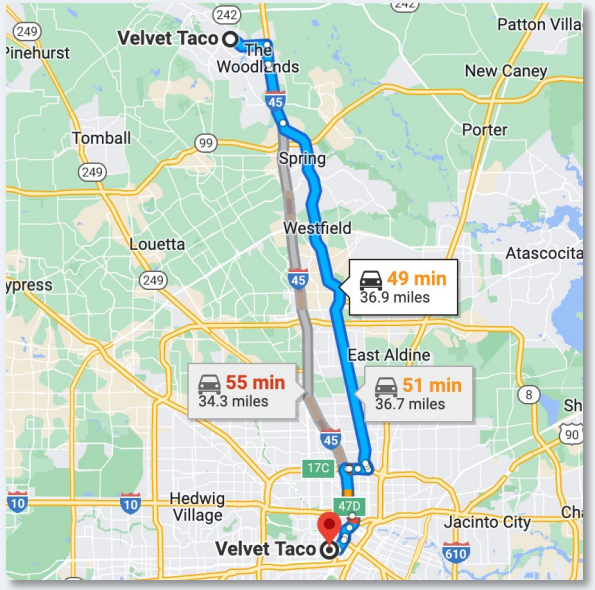
GUROBI OPTIMIZATION

01
Multi-Scenario Models

An efficient and thorough 'What-if' analysis

02
Solution Pool

Collect and retrieve alternative model solutions



03
Infeasibility Analysis

Shed some light on why a model has become infeasible

Core Solver Capabilities & Features

There are huge differences between solvers when it comes to product features.

MODEL TYPES

INTERFACES

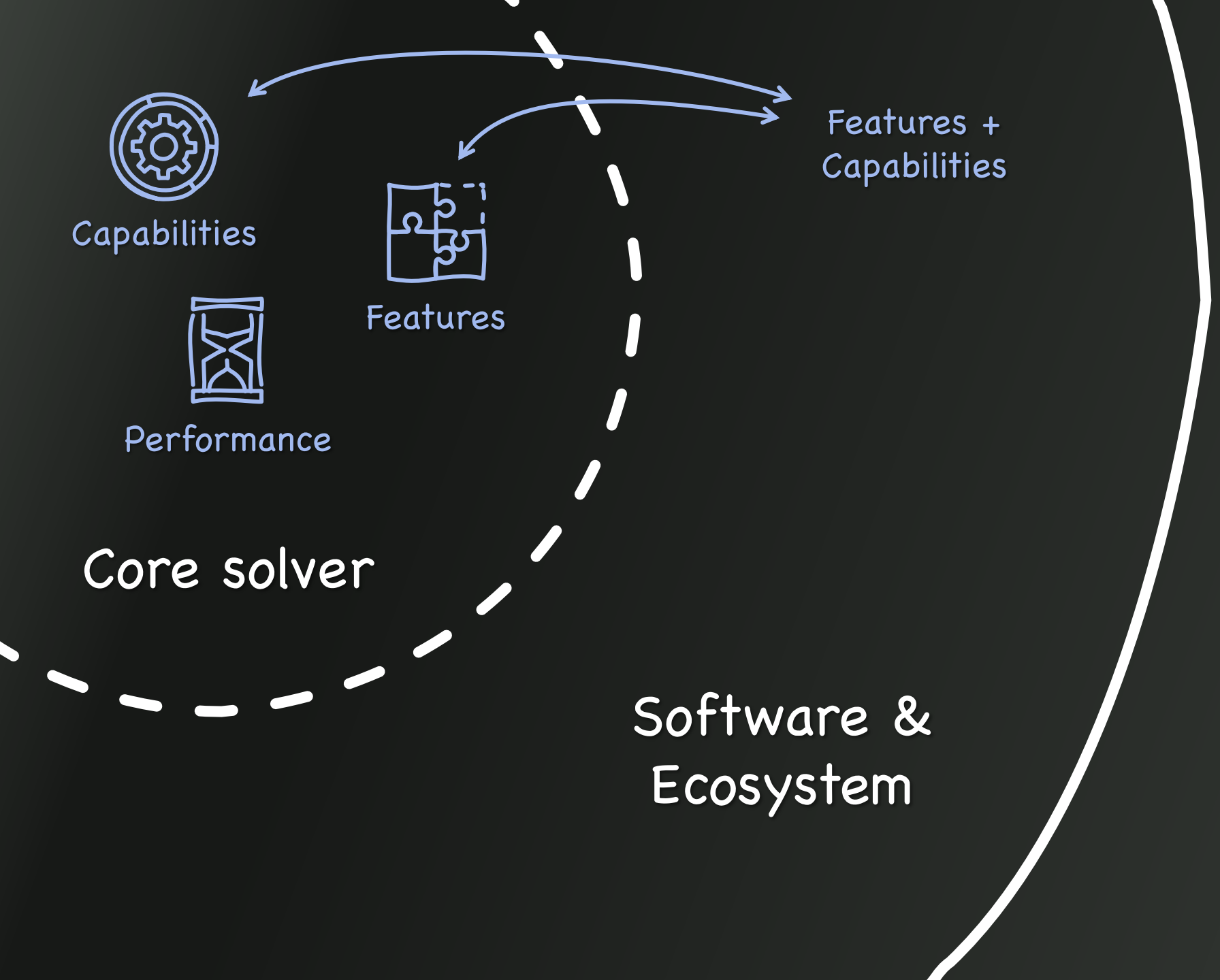
PRODUCTIVITY

MODEL ANALYSIS

05

SOLVER TUNING HELP

How does the solver help you navigate the parameters they have available?

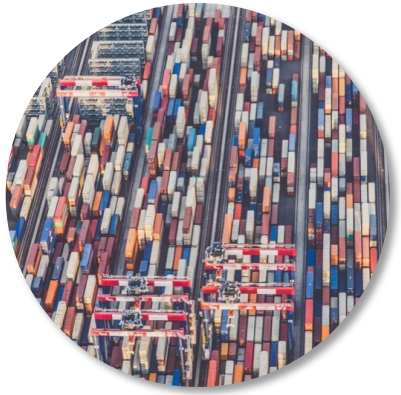


“The number one buying factor in mathematical solvers is speed.”

- McKinsey & Company

What does a faster solver buy you?

Solver performance is a combination of speed, robustness, tractability, and scalability



Expand the scope

More scalable solutions means can allow for considering more decisions at once

Real-time optimization

Faster models may mean you can set up a closed-loop system that finds then implements results

Voice in the room

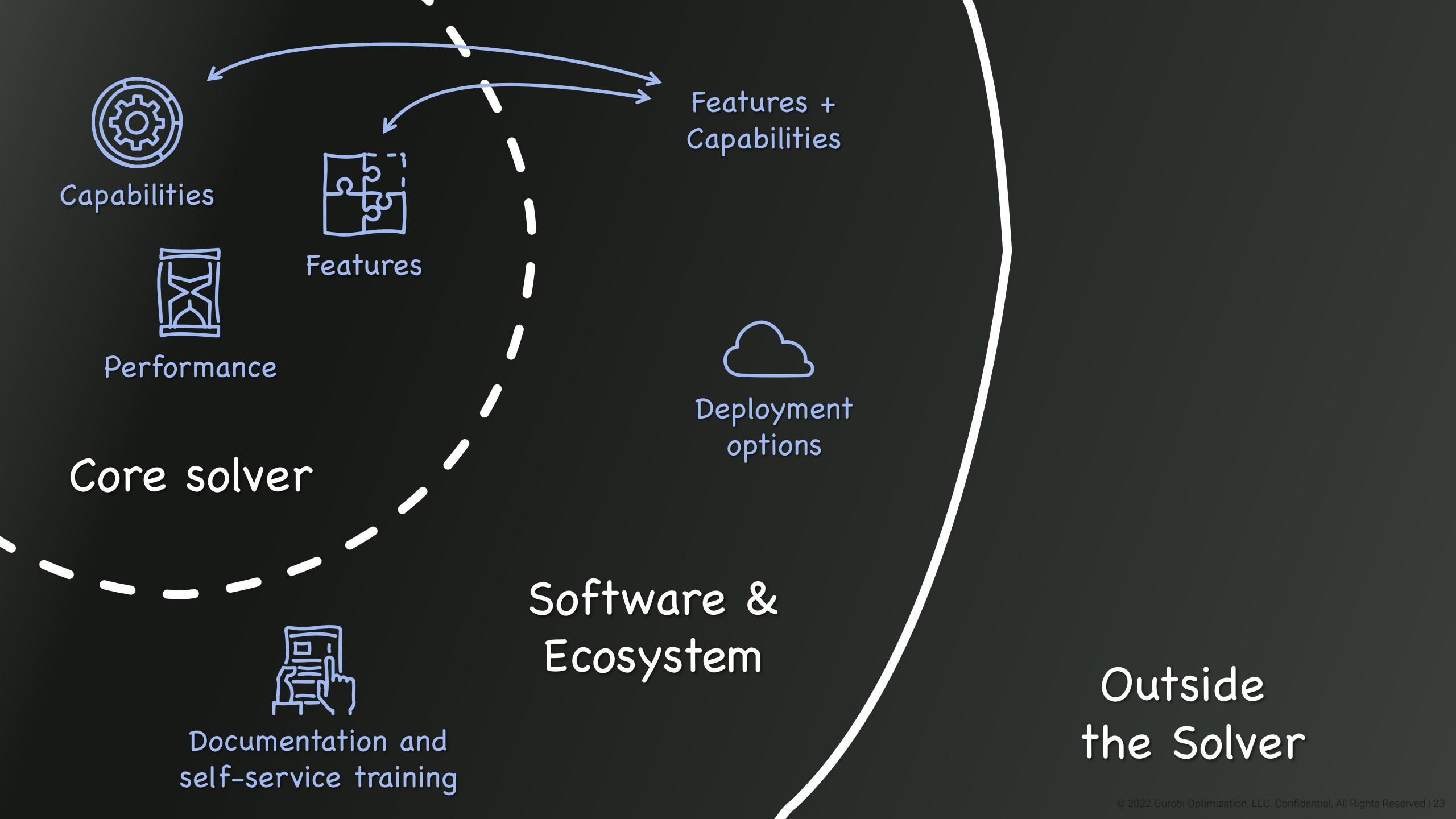
Answer questions as they arise in decision-making meetings

More questions answered

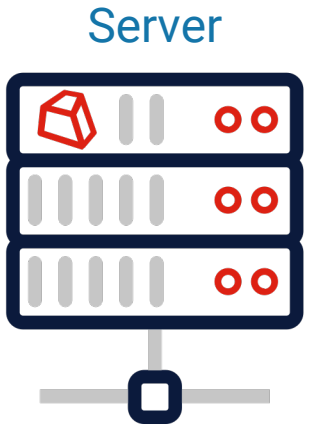
What if I had one more truck?

What if the price increased by ___?

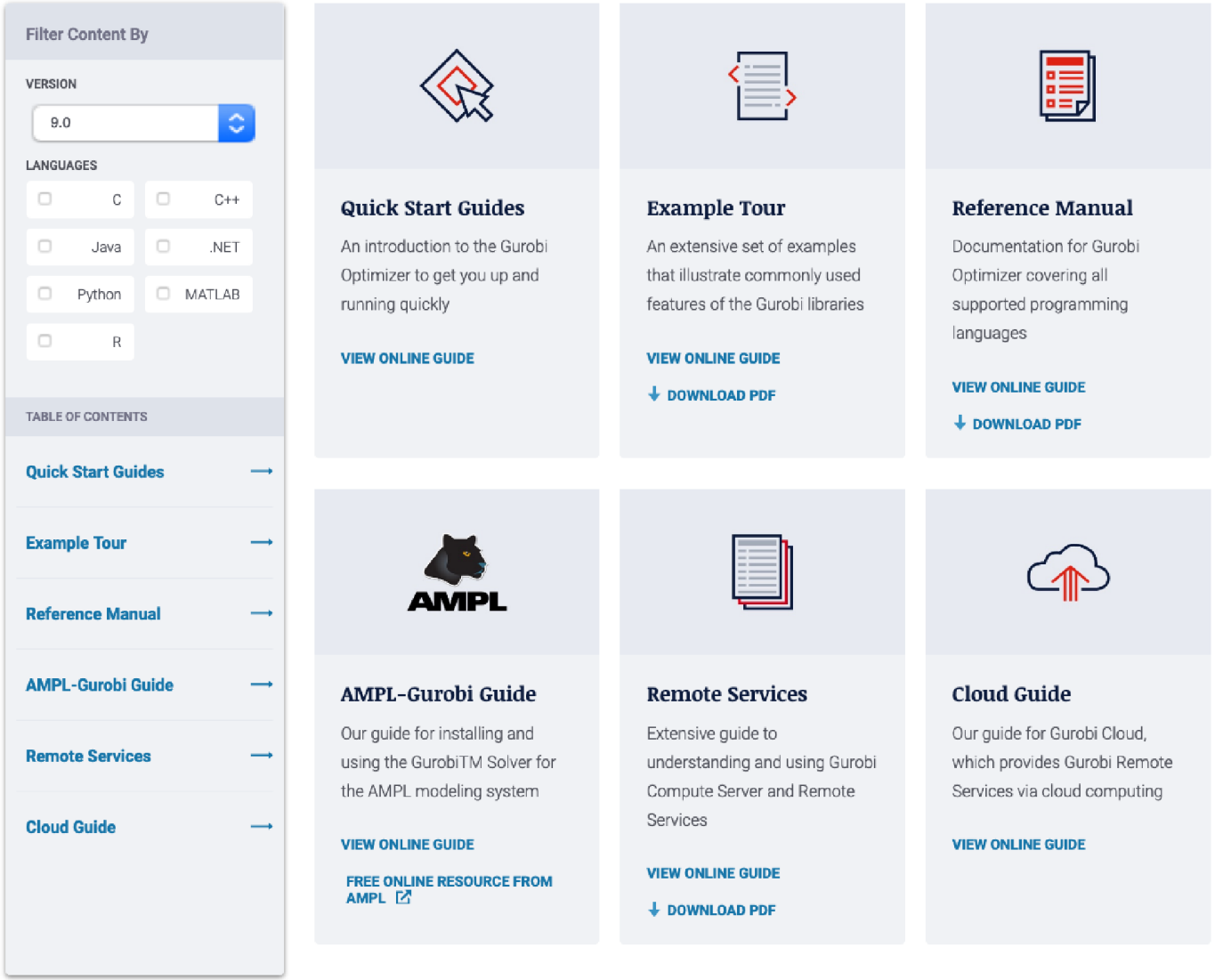
What constraints are holding us back?

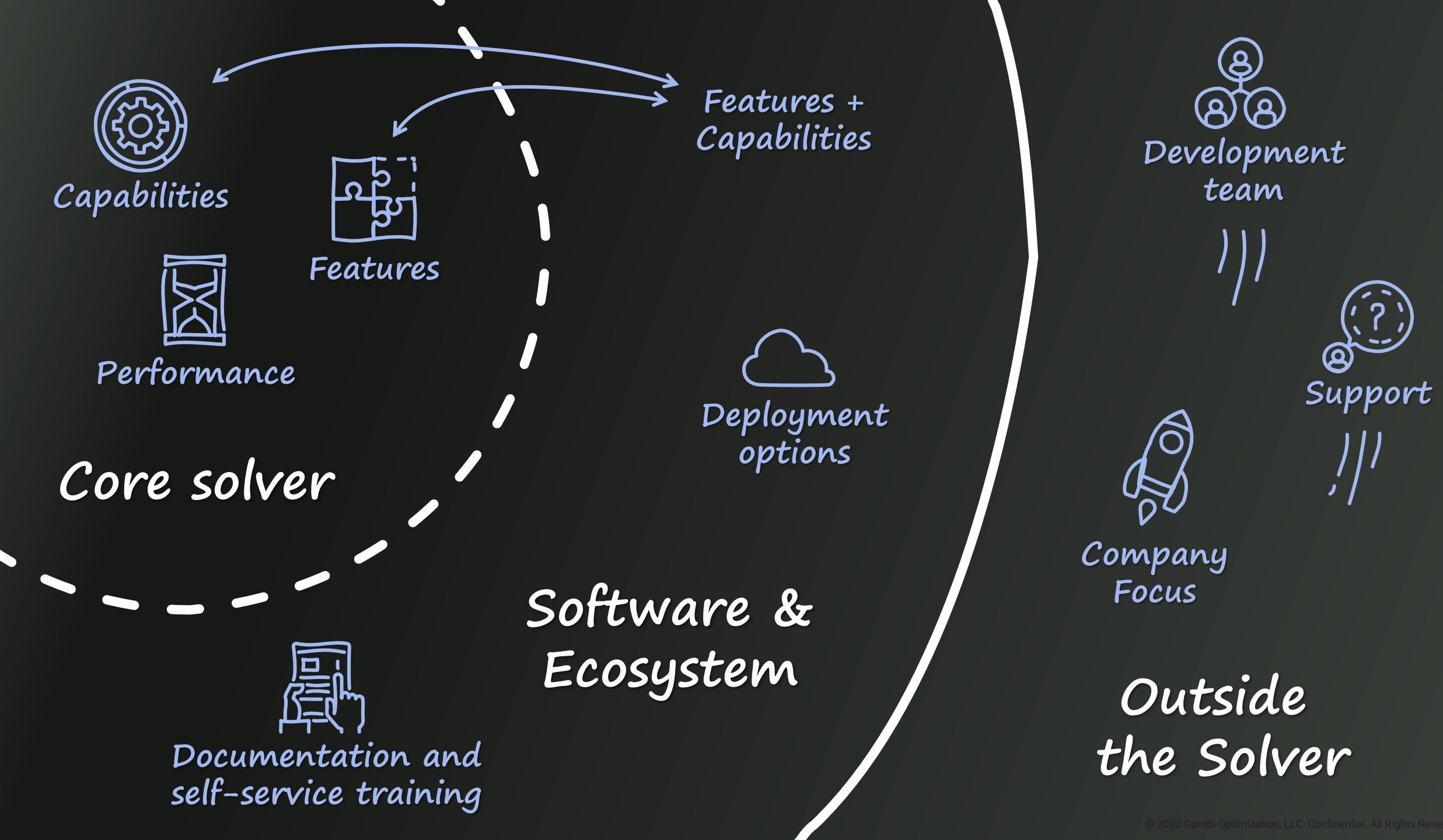


Solve your model in the way that's most convenient, based on your existing IT environment.



Our on-line documentation allows users to get the most from Gurobi Optimizer.





Company Focus

GUROBI OPTIMIZATION

Gurobi has a single product and our focus is in three areas:

Technical Superiority
Our vision is clear: We work hard to stay the most powerful solver on the market.

Ease of Use
If we can make it substantially easier to use our product, we do it.

Customer Centricity
Our customers' models and personal feedback drive our activities.

…because optimization is all we do…

Gurobi's Development Team

Dr. Ed Rothberg
CEO

Dr. Zonghao Gu
CTO

Dr. Tobias Achterberg
VP of R&D

Dr. Ed Klotz
Sr. Mathematical Optimization Specialist

Dr. Michel Jaczynski
Sr. Architect

Dr. Pierre Bonami
Sr. Developer

Dr. Stefan Heinz
Sr. Developer

Dr. Roland Wunderling
Sr. Developer

Dr. Robert Luce
Developer

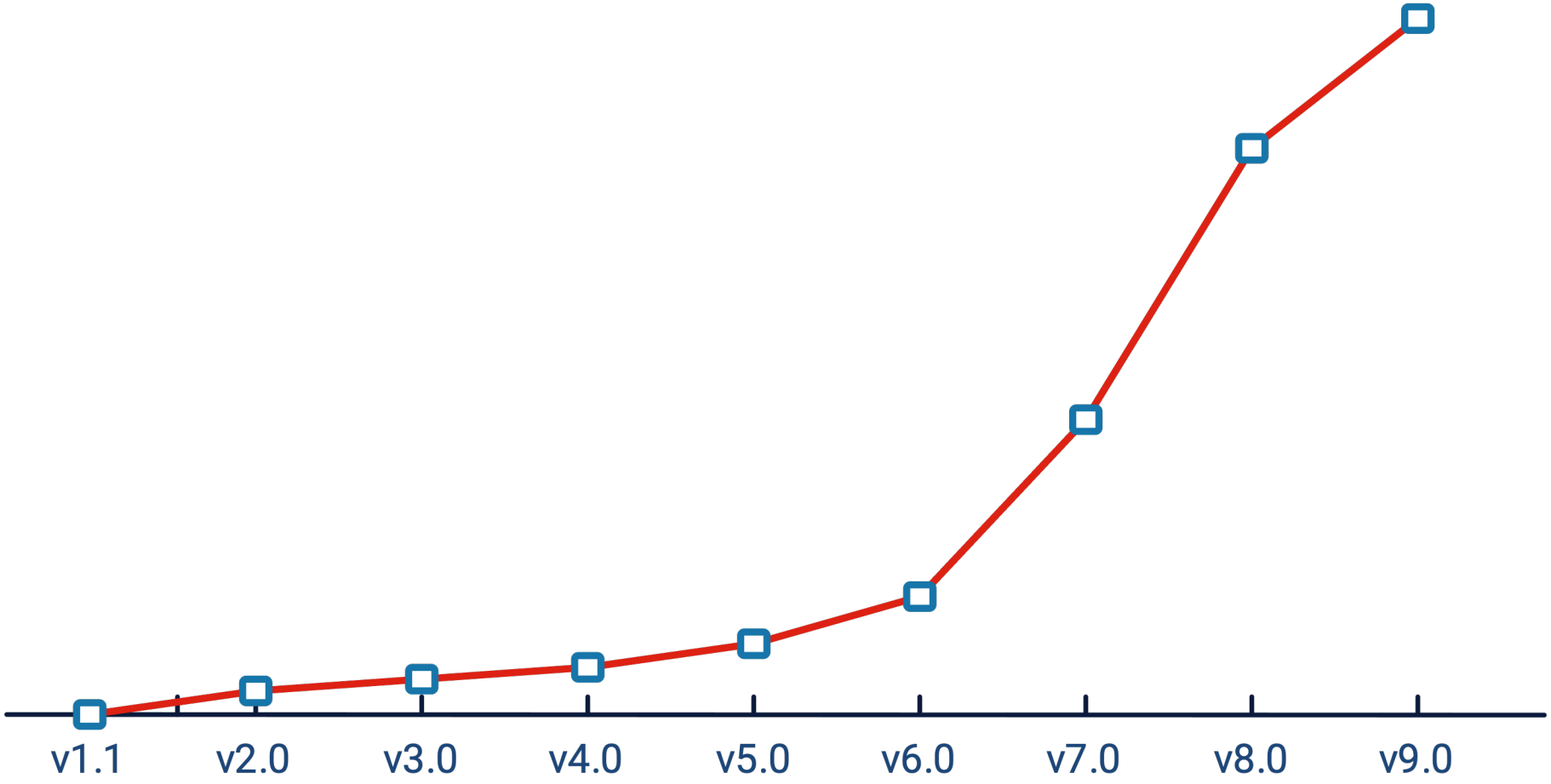
Olivier Noiret
Developer

Fernando Orozco
Sr. Software Engineer

Michael Winkler
Developer

We are obsessed with increasing performance

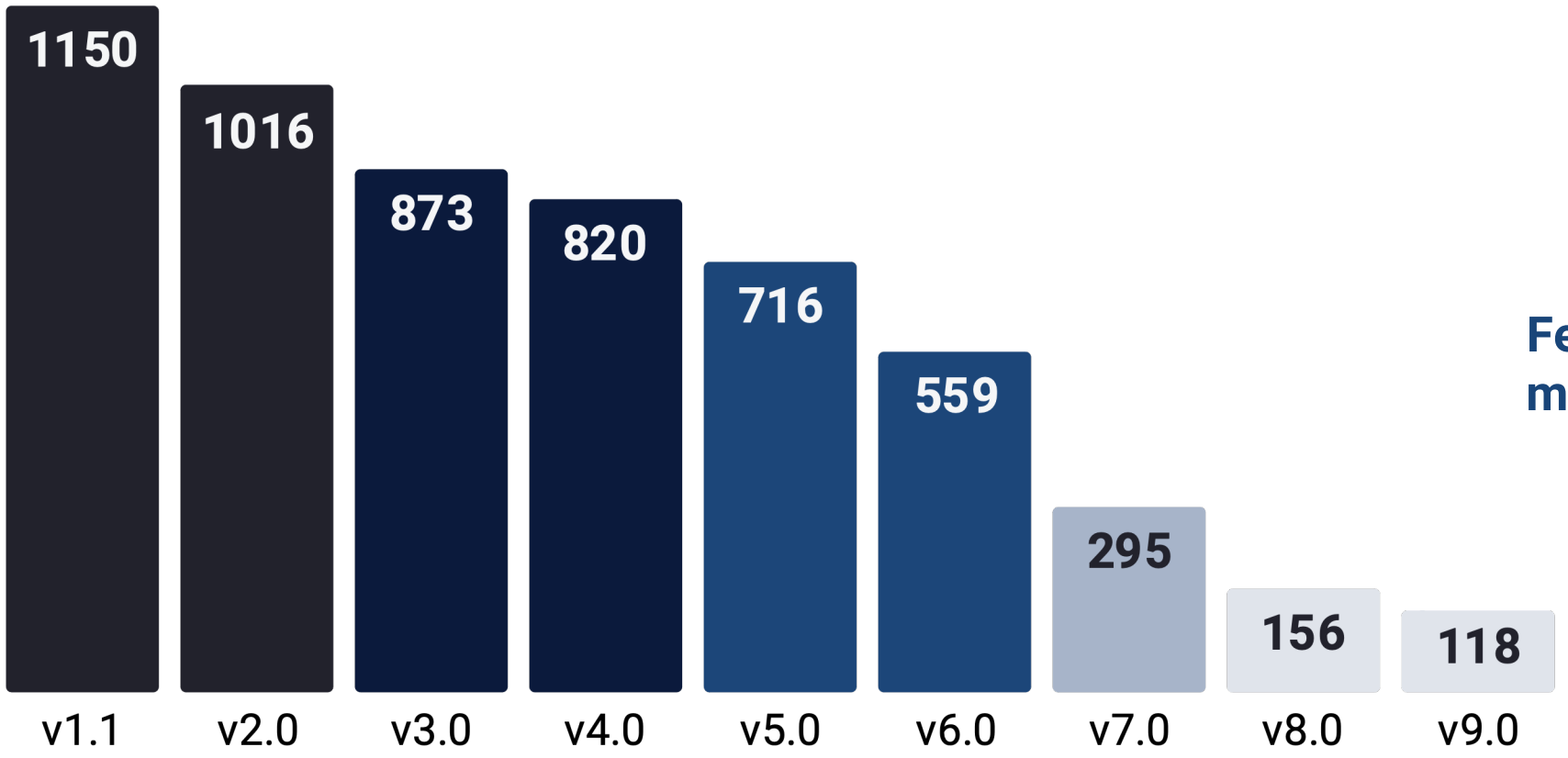
GUROBI OPTIMIZATION



59x

MIP Improvement over 9+ years

We have an extreme focus on solving “unsolvable” models



Fewer unsolvable models with each release!

Gurobi Support

We're right there with you as you build, solve, and deploy optimization models

24
Optimization experts around the globe

Direct access

to PhD-level experts for advice, insights, and collaboration

200+
years of experience with commercial models

We understand

the challenges you are facing because we've seen them before

so many
surprises and insights found in your models

We're curious

to get to know you and your challenges to find helpful nuggets to make you more effective

17
new features in 9.5 based on customer comments

We listen

to the ideas, issues, and challenges you are having to help shape Gurobi

Beyond the Basics: Support Questions

Gurobi Support is here to make you better and more effective

I am new to optimization

and I have written a blending problem that scales poorly with increasing problem size. Can you help me improve the solve time of the model?

I am not sure how many threads I need for my models.

Can you help me understand the impact of threads on performance?

.

My objective gets worse

when I add a piecewise linear constraint. Why is this happening?

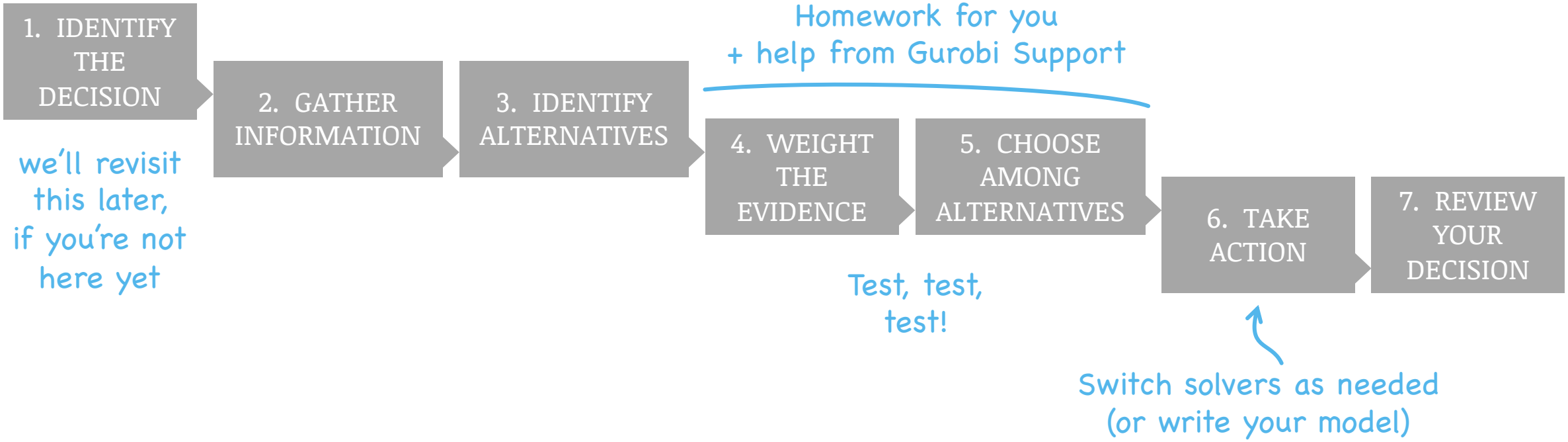
Most of my models solve quickly

except for this one. Why is that and what can I do to make an improvement?

Decision making process

Mapped to your "choosing a solver" decision

GUROBI OPTIMIZATION



Based on the Seven Steps to a Decision Making Process from UMass Dartmouth

Decision making process

Mapped to your "choosing a solver" decision

GUROBI OPTIMIZATION

Based on the Seven Steps to a Decision Making Process from UMass Dartmouth

Do I need to switch solvers?

GUROBI OPTIMIZATION

It‘s all about the performance experience:

- Model runtimes do not scale with more data
- The results of the current solver are not robust/reliable
- The current solver struggles with increased model complexity

There can be significant economic value behind better solver performance:

- Benefits of reliable, optimal results in shorter time
- Most performance improvements are on the algorithmic level

Switching essential parts of a software system always comes with certain risks.

- Lack of knowledge
- Change of system behavior
- Deployment effort

GUROBI OPTIMIZATION

If I can leave you with three main thoughts

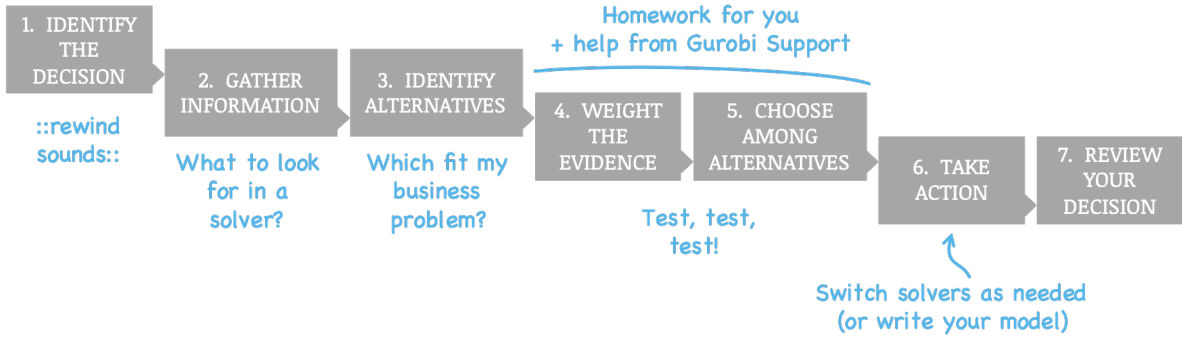
Speed matters

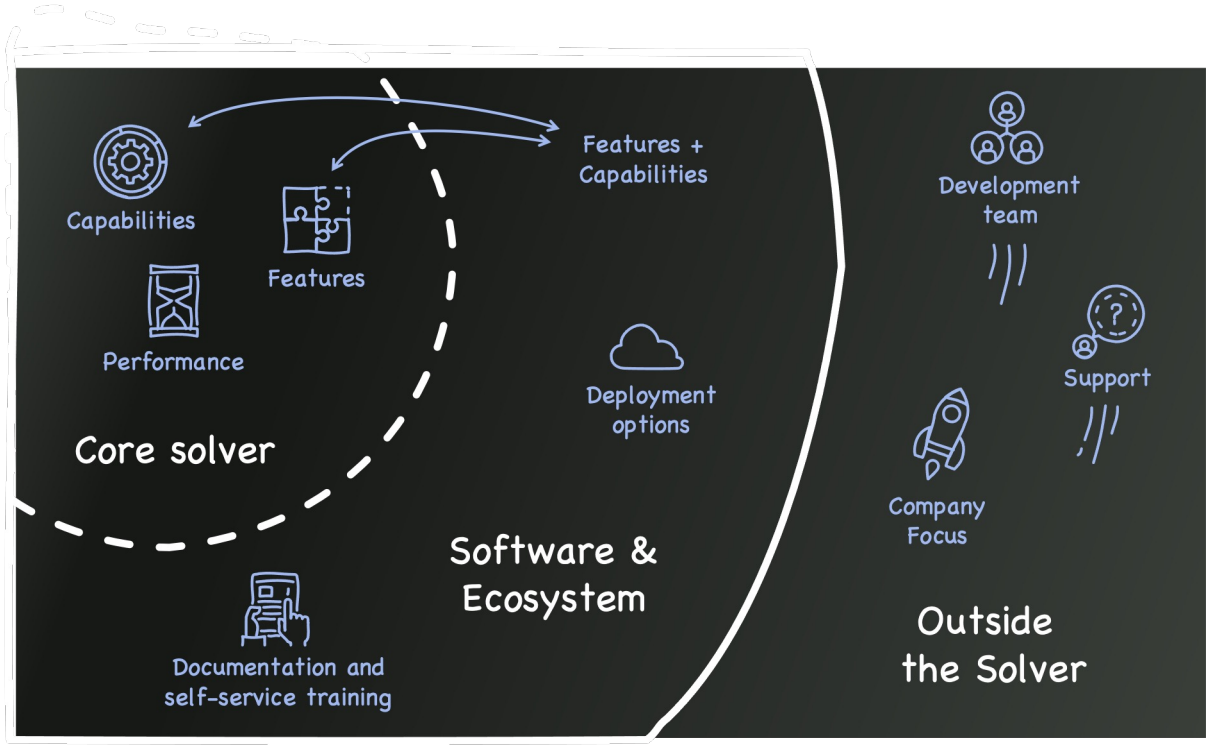
R&D, Company focus matters

Support matters

Decision making process

Mapped to your "choosing a solver" decision





Thank You

For more information: gurobi.com

Alison Cozad
Optimization Engineer

cozad@gurobi.com

Numerical Challenges

GUROBI OPTIMIZATION

Mathematical computations on a computer suffer from an inherent problem: Limited accuracy

- The infinite set of real numbers is mapped to a finite set of states (64 bits).
- In fact, there are infinite numbers without an exact representation.
- Emulating exact arithmetics in software is not an options: It is just too slow

Rounding errors everywhere

- Even the most basic mathematical laws do not hold anymore
- Violations and infeasibilities are expected and need to be handled properly
- Special care is required so that small rounding errors do not sum up to large errors in the results

A numerically stable implementation of optimization algorithms is very challenging

- Requires substantial modifications and extensions of the algorithms
- Important to install safeguards everywhere to distinguish actual improvements from random noise.