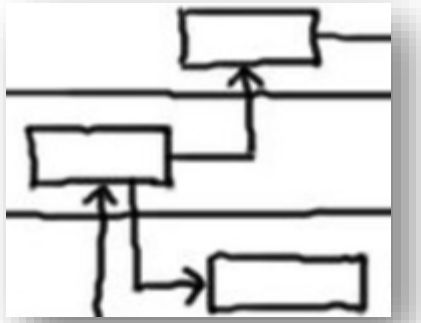
# KALASIM
## DISCRETE EVENT SIMULATOR

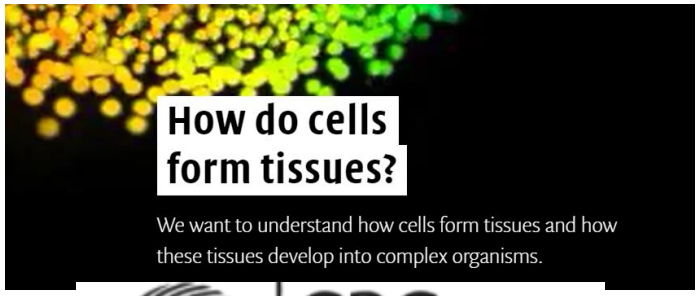


Dr. Holger Brandl – FOSDEM 2022

# ABOUT ME


HONDA

Universität Bielefeld




CBG
Max Planck Institute of Molecular Cell Biology and Genetics

SYSTEMA
Art of Automation

INDUSTRY 4.0


2010 — 2015 — 2020

GitHub Hobbies

`kscript` `krangl` `kravis` `kalasim`

systema.com/optimization

Blog Resources Support Contact

SYSTEMA
Art of Automation

INDUSTRY 4.0 COMPANY INDUSTRIES CAREER

Digital Transformation

Smart Manufacturing

SYSTEMA Portfolio

SAP Portfolio

Resources

Integration

Automation

Optimization

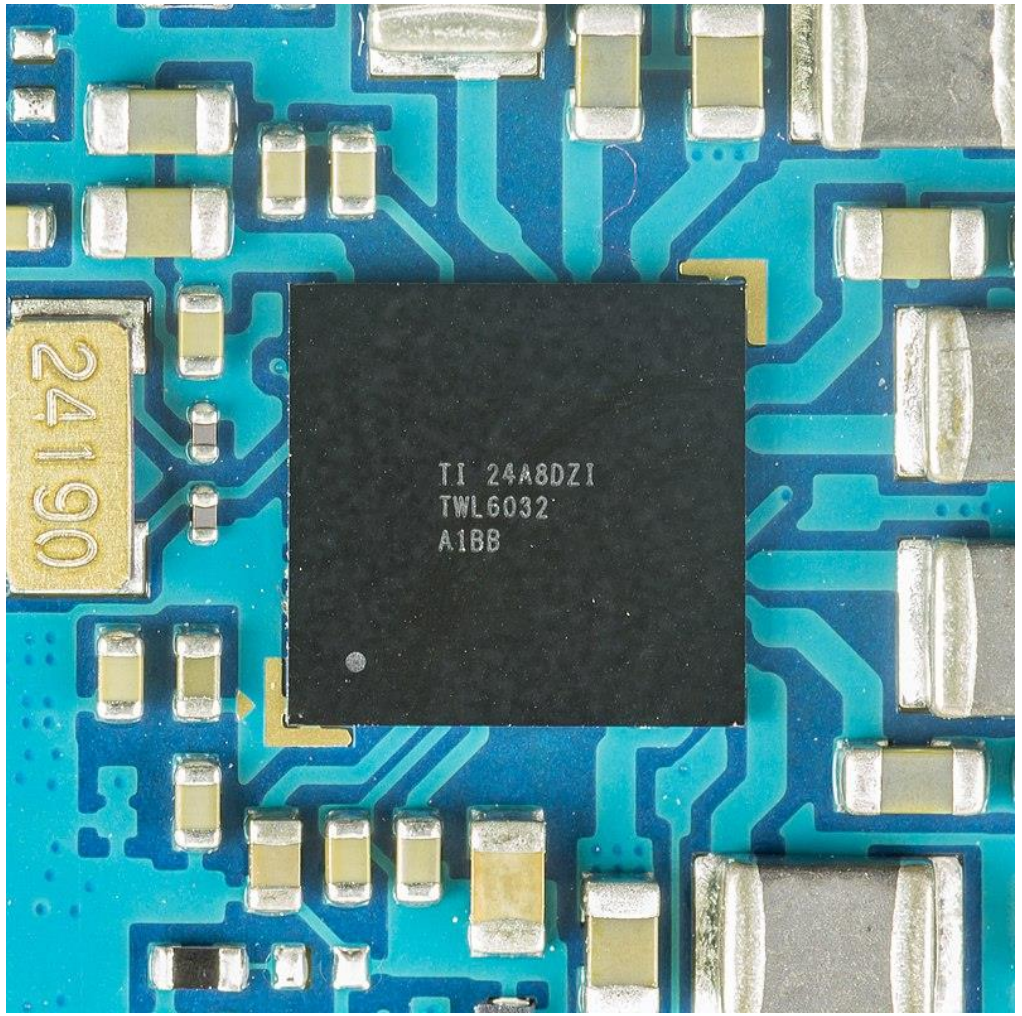
Visualization

Migration

Optimization is the art of maximizing manufacturing efficiency, throughput, OEE, yield, and quality by monitoring, analyzing, and iteratively tuning manufacturing processes.

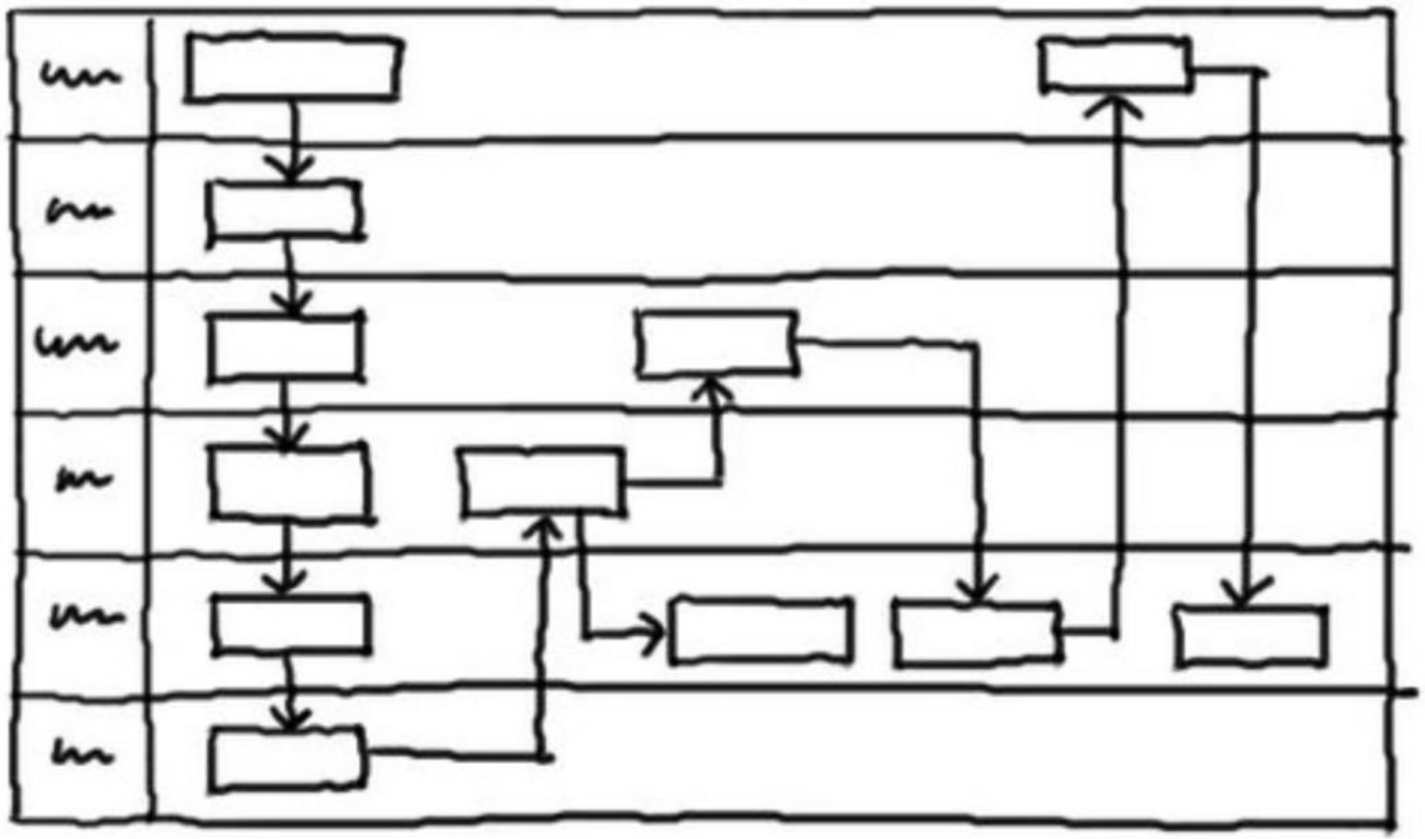
# EXAMPLE: SEMICONDUCTOR PRODUCTION

- Cycle Time: ~ 6 Months
- > 1000 Steps
- Product routes reiterate processes to add more layers on top of raw silicon wafer
- 30 floor “sky” scraper built with nanometer precision



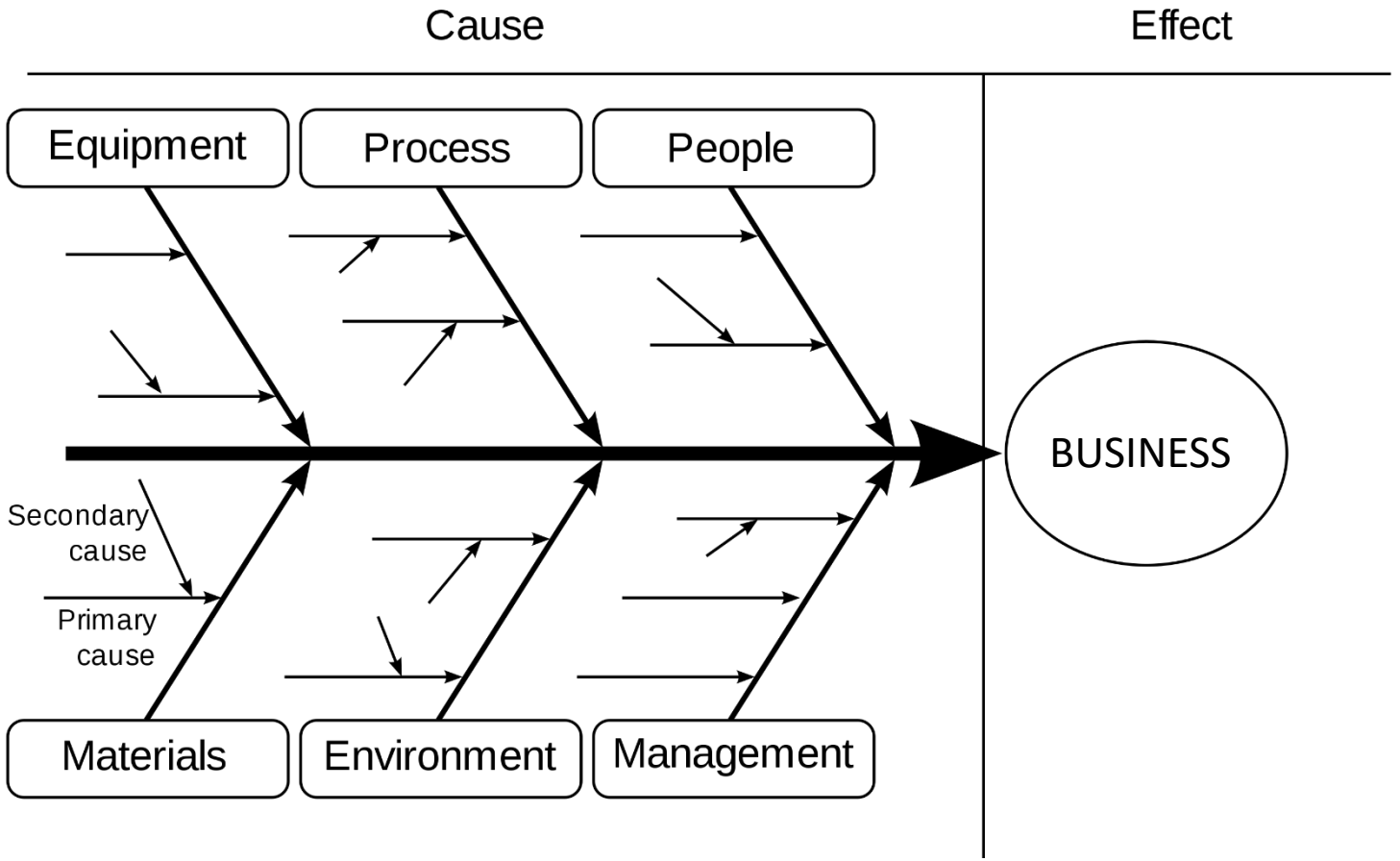

HOW TO MAKE MORE MONEY?

# MODEL THE BUSINESS PROCESS!



NICE, BUT WHERE IS THE BOTTLENECK?

# CAUSE AND CONSEQUENCE

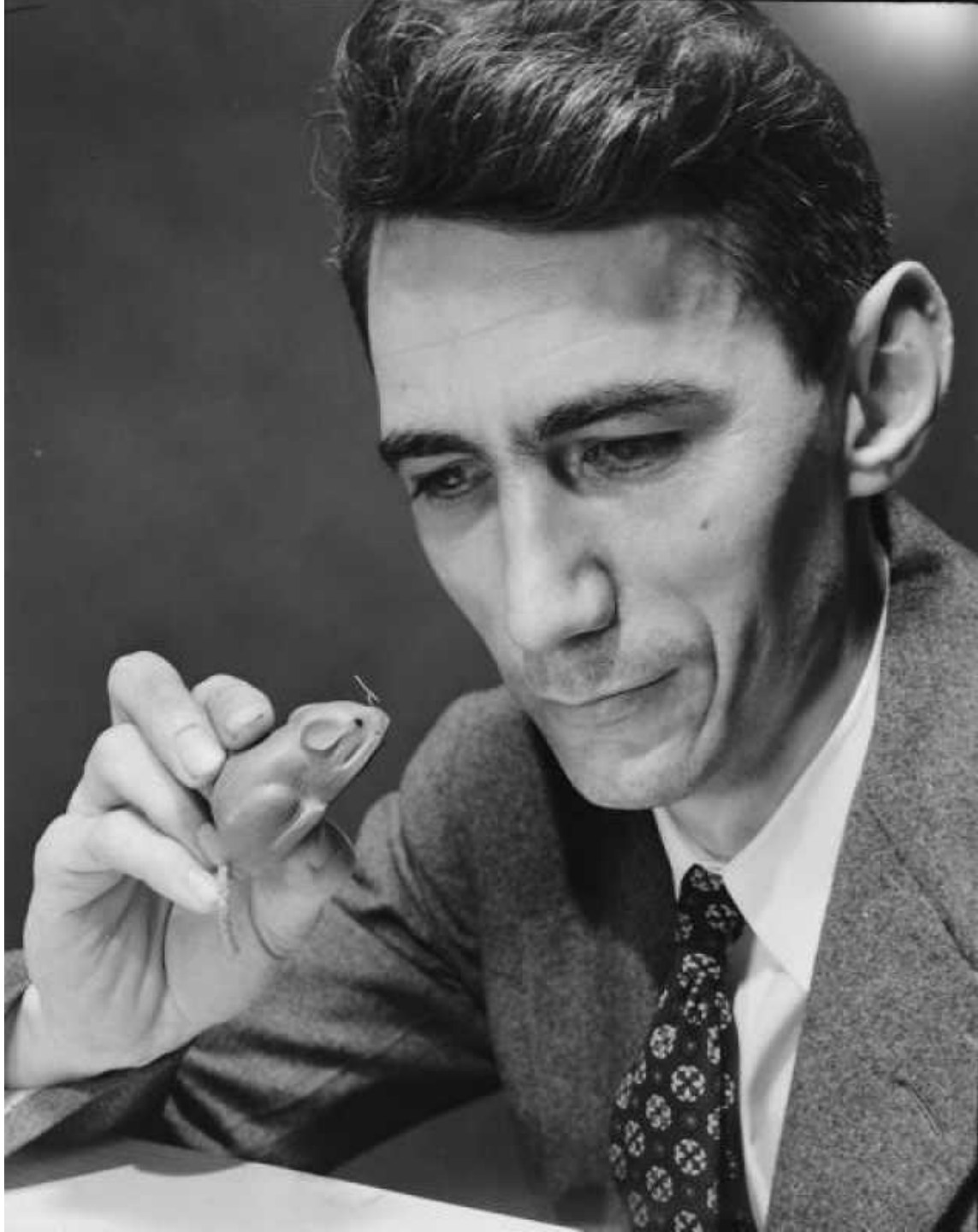



- Complex interplay of human, machine, material and methods
- Not always following intuition
- Applies to all businesses from service industry to manufacturing

# WHAT IS SIMULATION?

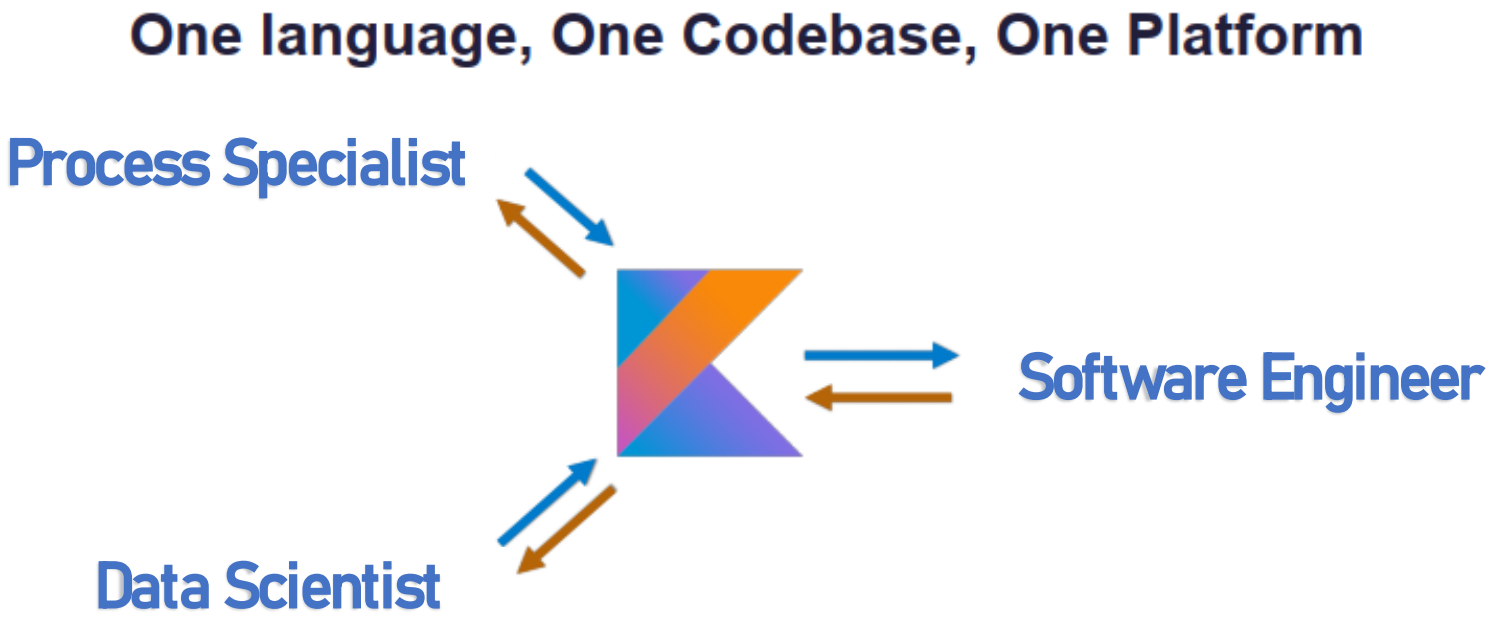
A simulation is the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behavior of the system or of evaluating various strategies (within the limits imposed by a criterion or a set of criteria) for the operation of the system.

Claude Shannon (1975)



# WHY ANOTHER DISCRETE SIMULATION ENGINE?

- Simmer, Salabim, SimJulia, SimPy, DSOL, …
- Low-code commercial tooling such as  AnyLogic or SIMIO



# Welcome to `kalasim`


`kalasim` is a discrete event simulator. It provides a statically typed API, dependency injection, modern persistence, structured logging and automation capabilities.

kalasim is designed for simulation practitioners, process analysts and industrial engineers, who need to go beyond the limitations of existing simulation tools to model and optimize their business-critical use-cases.

In contrast to many other simulation tools, `kalasim` is neither low-code nor no-code. It is *code-first* to enable change tracking, scaling, refactoring, CI/CD, unit-tests, and the rest of the gang that makes simulation development fun.

**Table of contents**

# holgerbrandl / kalasim

Public

Unpin | Unwatch 1 | Fork 5 | Starred 21

Code | Issues 13 | Pull requests | Discussions | Actions | Projects | Security | Insights | Settings

master | 4 branches | 31 tags | Go to file | Add file | Code

holgerbrandl Continued call-center article — e505375 23 hours ago — 673 commits

| | | |
|---|---|---|
| .github/workflows | Improved documentation | 12 months ago |
| .idea/codeStyles | Fixed #27: More informative resource events | 17 days ago |
| docs | Continued call-center article | 23 hours ago |
| gradle/wrapper | Matched letsplot display support API as much as possible | 2 months ago |
| modules | Continued elevator animation | yesterday |
| simulations | Fixed naming | 6 days ago |
| src | Continued call-center article | 23 hours ago |
| .gitattributes | override github language stats | 17 days ago |
| .gitignore | share codestyle in repo | last month |
| CHANGES.md | Improved documentation | 12 months ago |

## About

Discrete Event Simulator

data-science | simulation | optimization | agent-based-modeling | process-modeling | visulization | discrete-event-simulation

Readme

MIT License

21 stars

1 watching

5 forks

## Releases 29

v0.7.95 Latest

yesterday

+ 28 releases

# CORE FEATURES

`kalasim` is a generic process-oriented discrete event simulation engine

- Simulation entities have a generative **process description** that defines the interplay with other entities
- There is a well-defined rich process **interaction vocabulary**
- An **event trigger queue** maintains future action triggers and acts as sole driver to progress simulation state
- Built-in **monitoring and statistics** gathering across the entire API

# FIRST EXAMPLE

Key Types

- `Component`
- `Resource`
- `State`

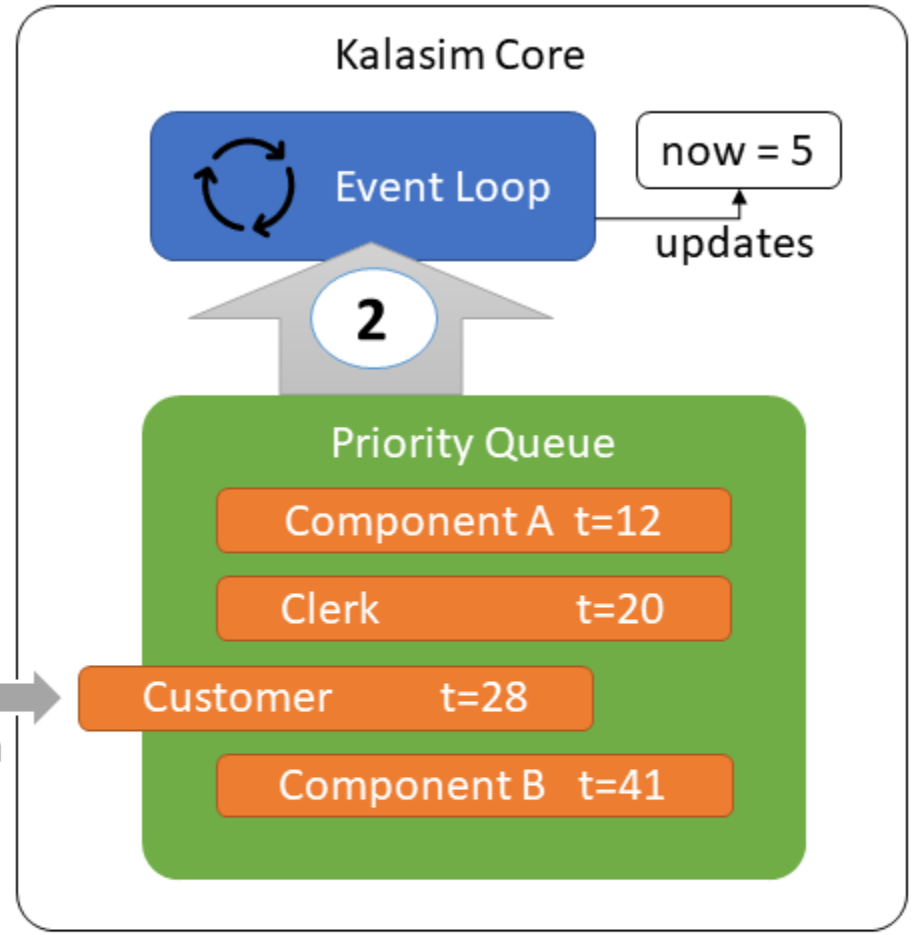
Used Interaction Methods

- `request()` – Ask (and wait) for resource
- `hold()` – Suspend execution
- `wait()` – Wait for specific state

```
////Cars.kts
import org.kalasim.*

class Driver : Resource()
class TrafficLight : State<String>("red")

class Car : Component() {

    val trafficLight = get<TrafficLight>()
    val driver = get<Driver>()

    override fun process() = sequence {
        request(driver) {
            hold(1.0, description = "driving")

            wait(trafficLight, "green")
        }
    }
}

createSimulation(enableConsoleLogger = true) {
    dependency { TrafficLight() }
    dependency { Driver() }

    Car()
}.run(5.0)
```

# EVENT LOOP ARCHITECTURE

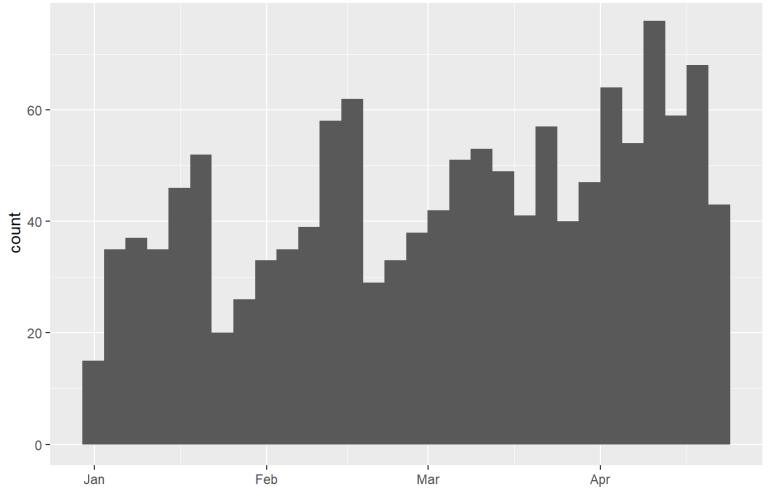
```
class Customer(val clerk: Resource) : Component() {

    override fun process() :Sequence<Component> = sequence {
        // do shopping
        hold(ticks = 23.0)

        // wait for an empty counter
        request(clerk)

        // billing process
        hold(ticks = 2.0, priority = HIGH)
    }
}
```

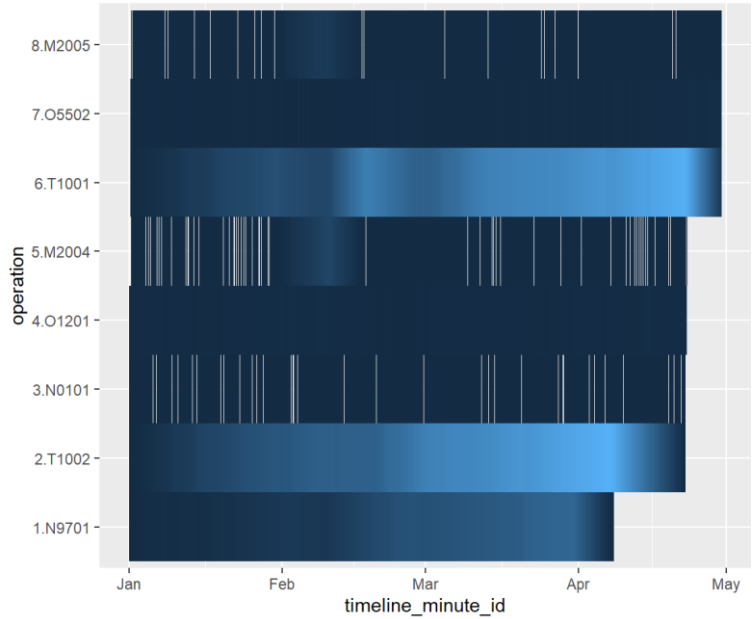



1. Stall customer's `process` execution and insert it into event queue by time (and optional priority to resolve ambiguities)
2. Poll, while queue is not empty and resume (or start) `process`
3. Event loop will continue execution here. Components are terminated once `process` end has been reached

# CAN WE MODEL AN ENTIRE FACTORY WITH `kalasim`?

- Yes, we can!
- Can we study its dynamics? Sure.
- Can we discuss it here? Nope.

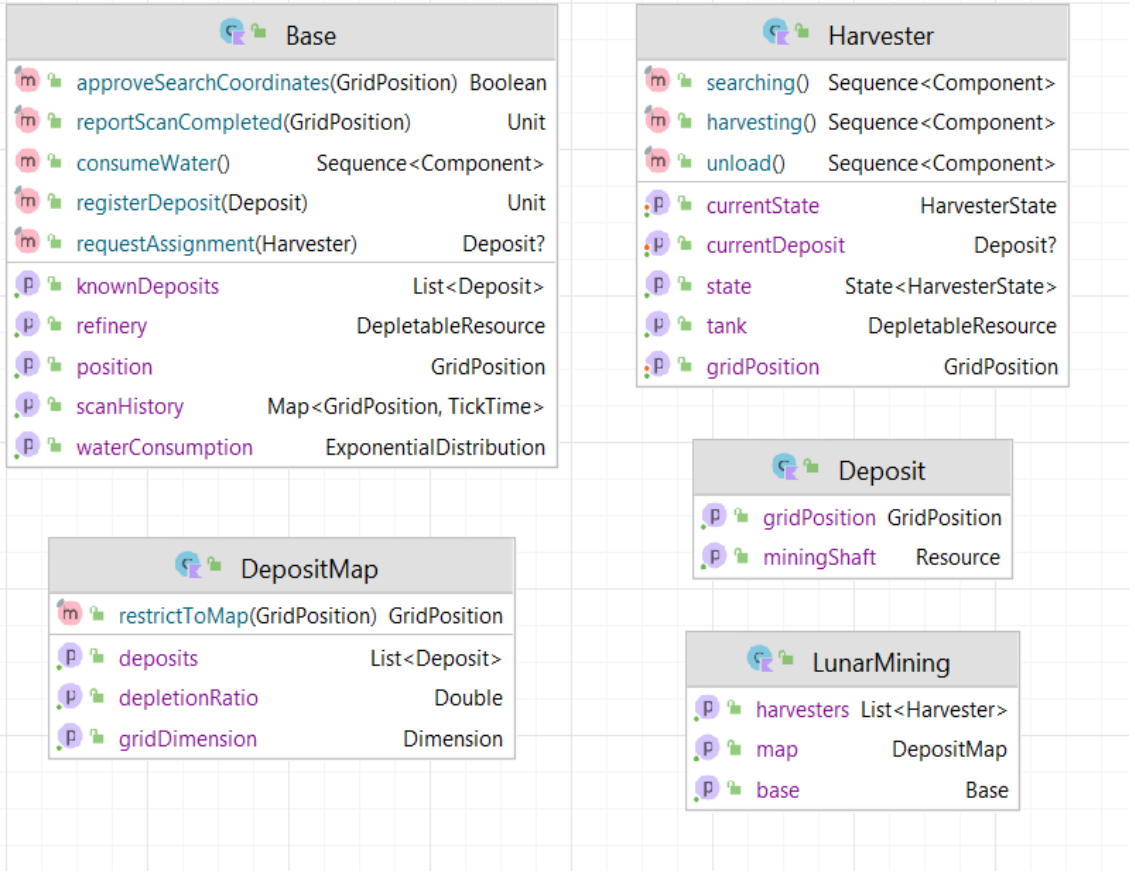



https://www.kalasim.org/animation/lunar_mining/

# LUNAR MINING PROCESS

Mining robots scan the surface of the moon for depletable water ice deposits.
It is a complex business process, and the lives of thirsty astronauts are at stake.




```
fun unload() = sequence {
    moveTo(base.position)

    val unloadingUnitsPerHours = 20  // speed of unloading

    // unloading time correlates with load status
    currentState = UNLOADING
    hold((tank.level / unloadingUnitsPerHours).roundToInt().hours,
         "Unloading ${tank.level} water units")

    // put the water into the refinery of the base
    put(get<Base>().refinery, tank.level)

     // empty the tank
    take(tank, tank.level)

    activate(process = Harvester::harvesting)
}
```

# PROCESS ANALYTICS TOOLBOX INCLUDED

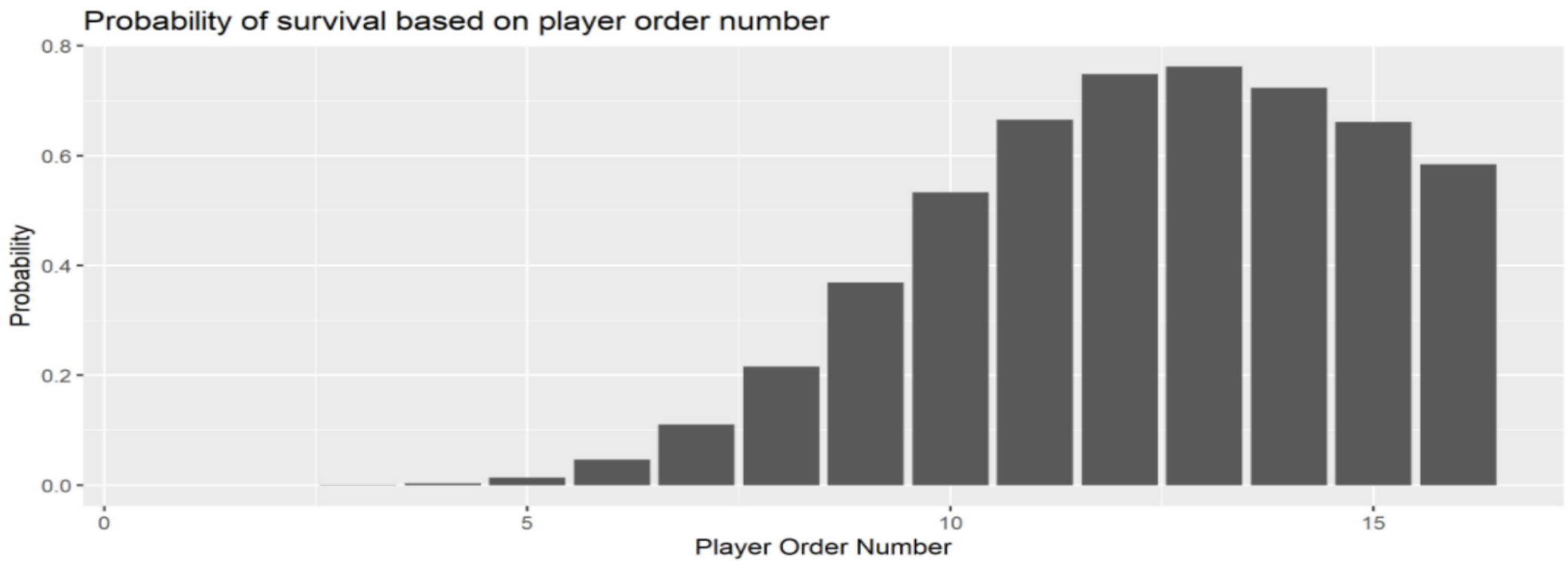
`kalasim` offers various means to analyze data created by a simulation

- The **Event Log** tracks events in a simulation
- **Monitors** track state and statistics of the basic elements within a simulation, and may be used for domain-specific entities as well
- **Lifecycle Records** summarize a component's states history
- **Visualization** to inspect complex spatio-temporal patterns

https://www.kalasim.org/analysis/

In [92]:
```
val survivalProbByNo = (1..manyGames.first().numPlayers).map { playerNo ->
    playerNo to manyGames.count { it.playerSurvived(playerNo) }.toDouble() / manyGames.size
}

survivalProbByNo.plot(x = { it.first }, y = { it.second }).geomCol().labs(
    title = "Probability of survival based on player order number",
    x = "Player Order Number",
    y = "Probability"
)
```

Out[92]:

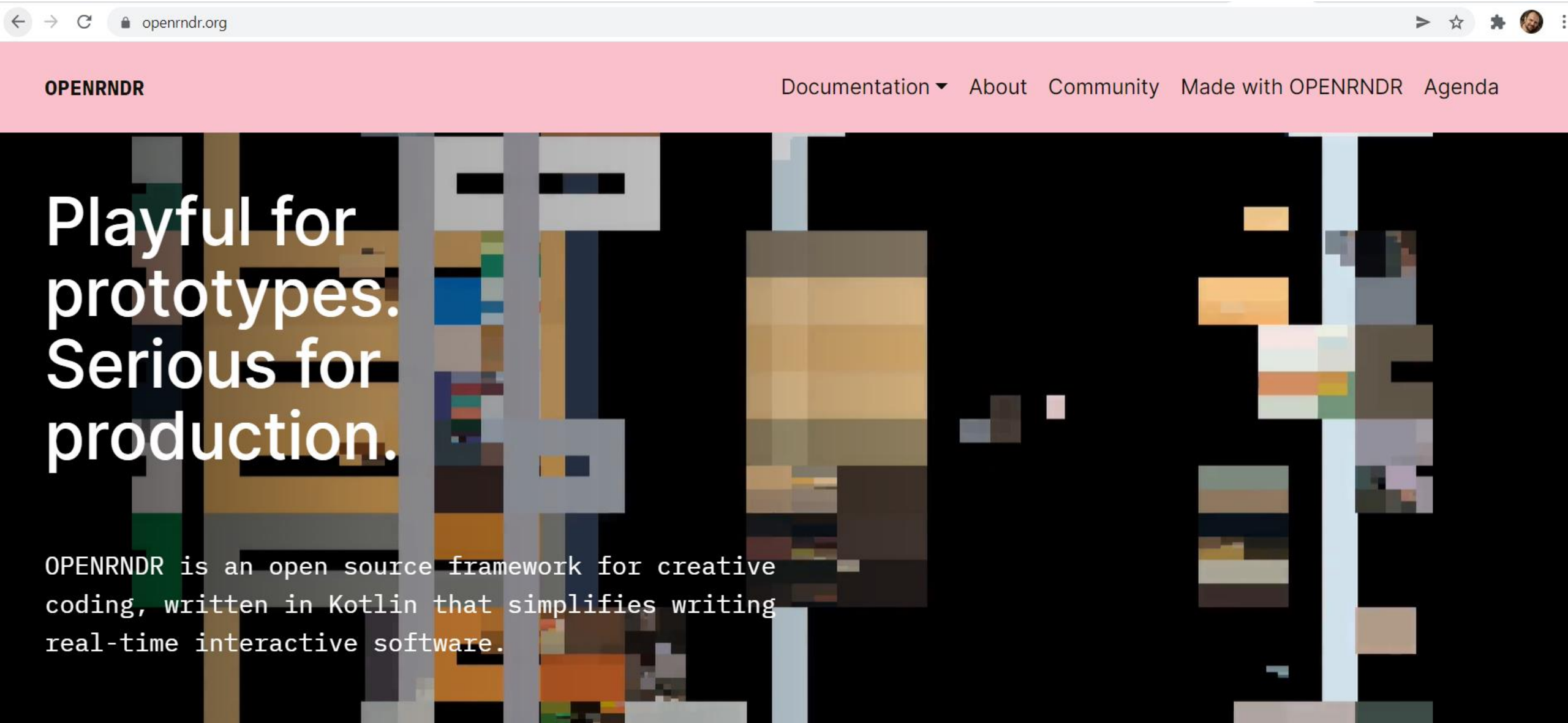



So indeed there seems a strategy to maximize your odds of survival in the game. Simply pick **No13**, and you may live more likely compared to any other starting number.

https://nbviewer.org/github/holgerbrandl/kalasim/blob/master/docs/userguide/docs/animation/lunar_mining.ipynb

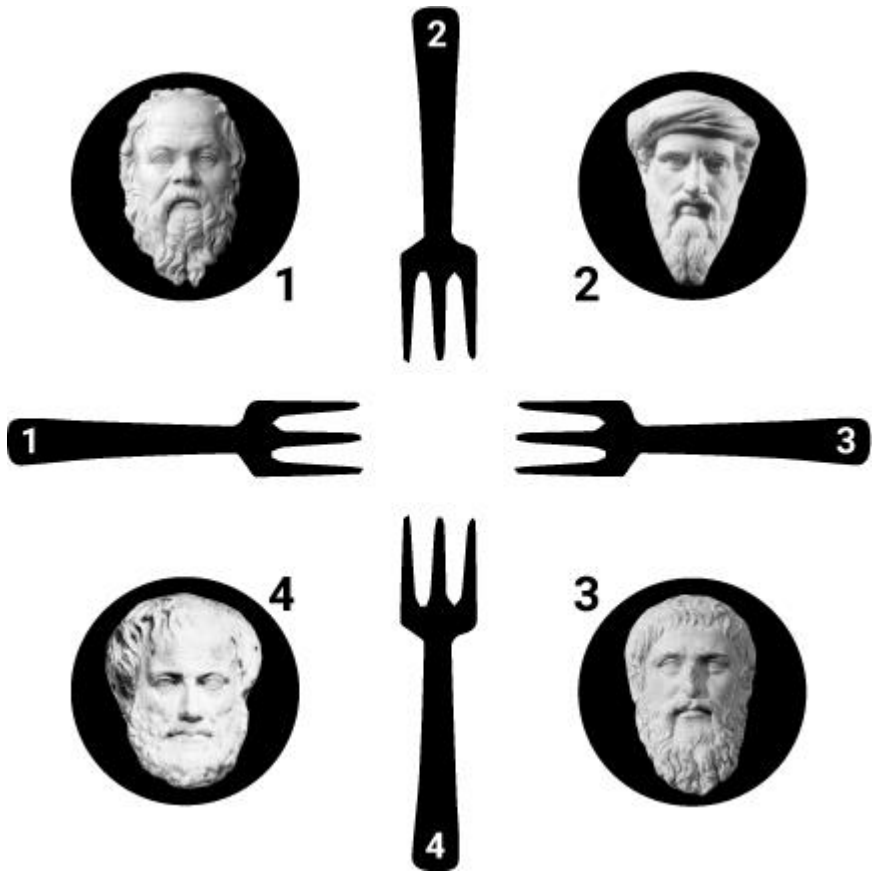
https://www.youtube.com/watch?v=r2igemRWk7s

# PROCESS ANIMATION → OPENRNDR



LIVE-CODING SESSION

# HAVE FOOD & FUN WITH DINING PHILOSOPHERS




@

CASUS
CENTER FOR ADVANCED
SYSTEMS UNDERSTANDING

https://www.kalasim.org/examples/dining_philosophers/

https://www.youtube.com/watch?v=GzcpJ_7pWvk

# KALASIM APPLICATIONS

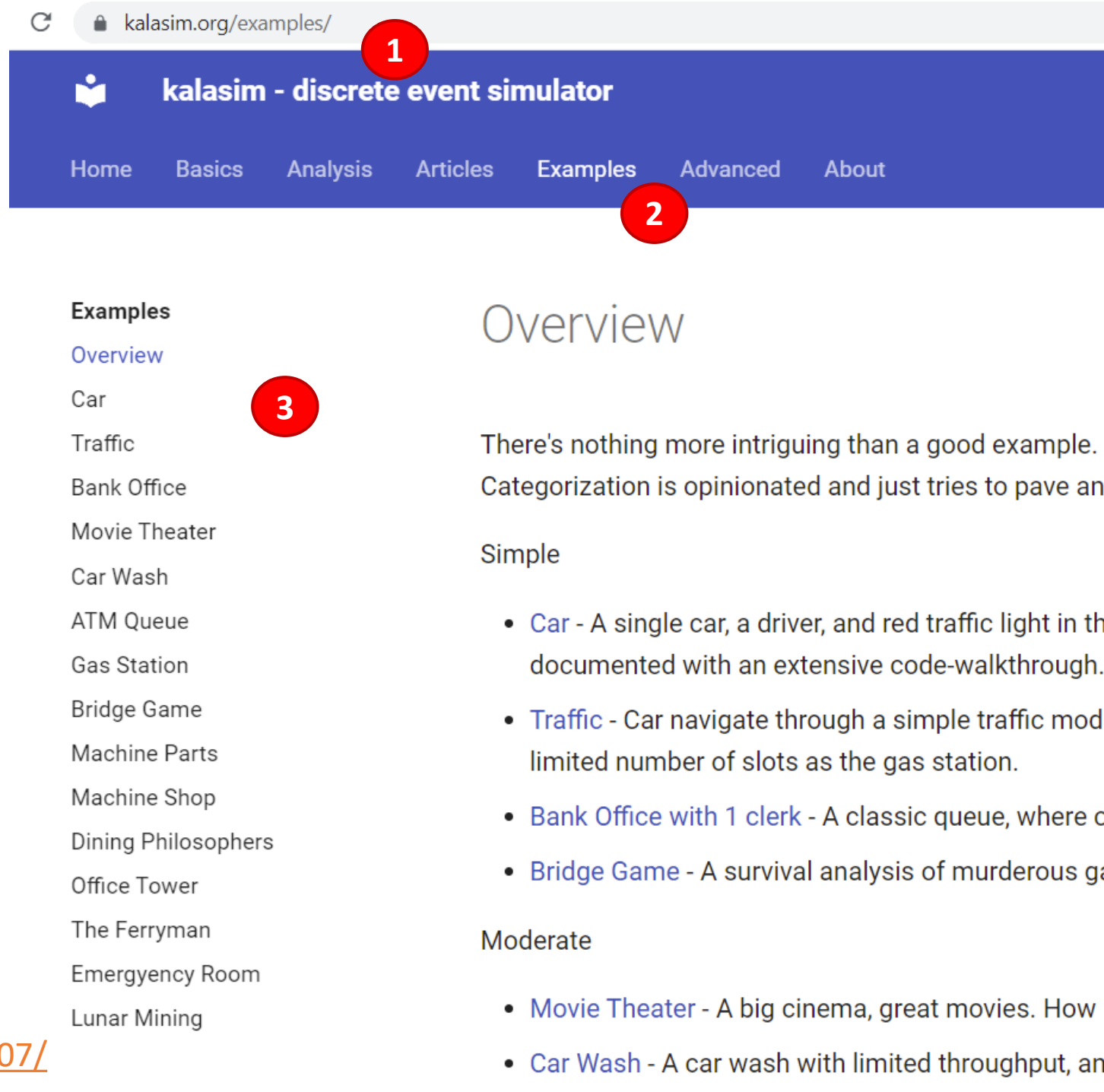
- Optimization
- AI Research
- Bottleneck Analytics
- Integration Testing
- Games & Fun
- Capacity Planning

# WHAT'S NEXT?

Me

- Evolve Process Animation API
- Environment snapshotting & branching (based on kryo)
- Agent-based modeling
- Evolve process mining toolbox
- (Help others to) Make more money

You:




https://www.kalasim.org/articles/2021-11-27-kalasim-v07/

# IMAGE ATTRIBUTIONS

- The moon (CC BY-SA 3.0) https://en.wikipedia.org/wiki/Moon
- Chip (CC BY-SA 4.0) https://commons.wikimedia.org/wiki/File:Samsung_Galaxy_Tab_2_10.1_-_Texas_Instruments_TWL6032-3960.jpg
- Business Process 1 (CC BY-SA 4.0) https://www.flickr.com/photos/davegray/5630708345
- Business Process 2 (CC BY-ND 2.0) https://commons.wikimedia.org/wiki/File:Gamification-in-business-illustration-web.jpg
- OPENRNDR http://openrndr.org
- Kotlin Features https://github.com/thomasnield/kotlinconf-datascience-talk
- Claude Shannon https://commons.wikimedia.org/wiki/File:Claude_Shannon_1776.jpg
- Ishikawa Fishbone Diagram (CC BY-SA 3.0) https://commons.wikimedia.org/wiki/File:Ishikawa_Fishbone_Diagram.svg
- Money (CC0) https://www.maxpixel.net/Market-Currency-Money-Europe-Euro-Coins-Yellow-1353420

THANK YOU FOR JOINING!

THANKS TO THE ORGANIZERS OF FOSDEM-2022

LET'S STAY IN TOUCH!

DO YOU WANT TO MAKE MORE MONEY WITH YOUR BUSINESS? YOU KNOW WHAT TO DO! ☺

@holgerbrandl