Xebia

Software Development Done Right

# Lightning Fast Learning ….using simulations

Pieter Rijken

# Pieter Rijken

Job

Agile Coach / Trainer

Clients

KPN, ING NL, Rabobank, Nationale Nederlanden, Aegon

Contact

prijken@xebia.com
+31 6 83036743
pieterjrijken
pieter_rijken

ACCREDITED KANBAN TRAINER

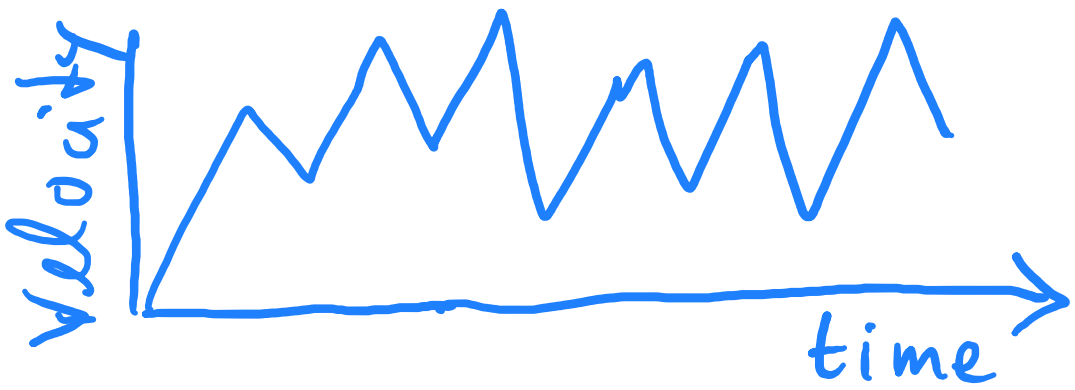
Xebia

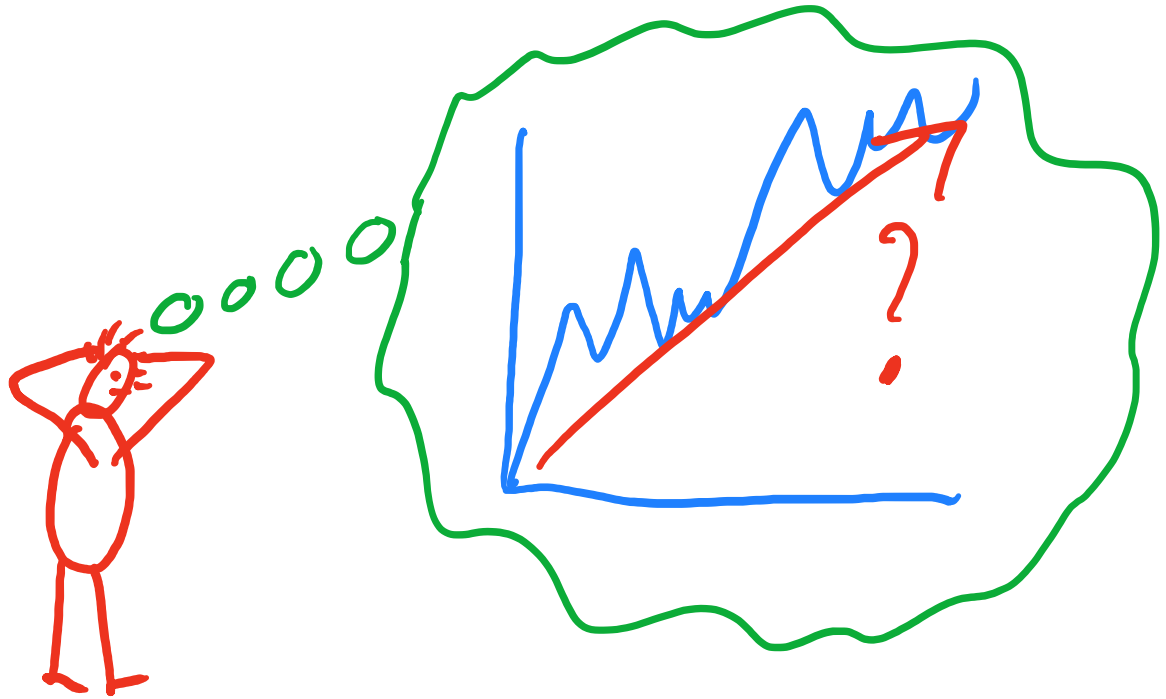
# Case: introduction

- Large telecoms company
- Scrum team
- Complex IT landscape
- Team distributed over The Netherlands and Romania
- Develops application to support internal business units
- Dependencies with other teams

# After a couple of sprints

€?
Lead time?




Xebia


- ❑ At the start of sprint the board is reset
- ❑ Improvement action result is known
- ❑ ….or not?

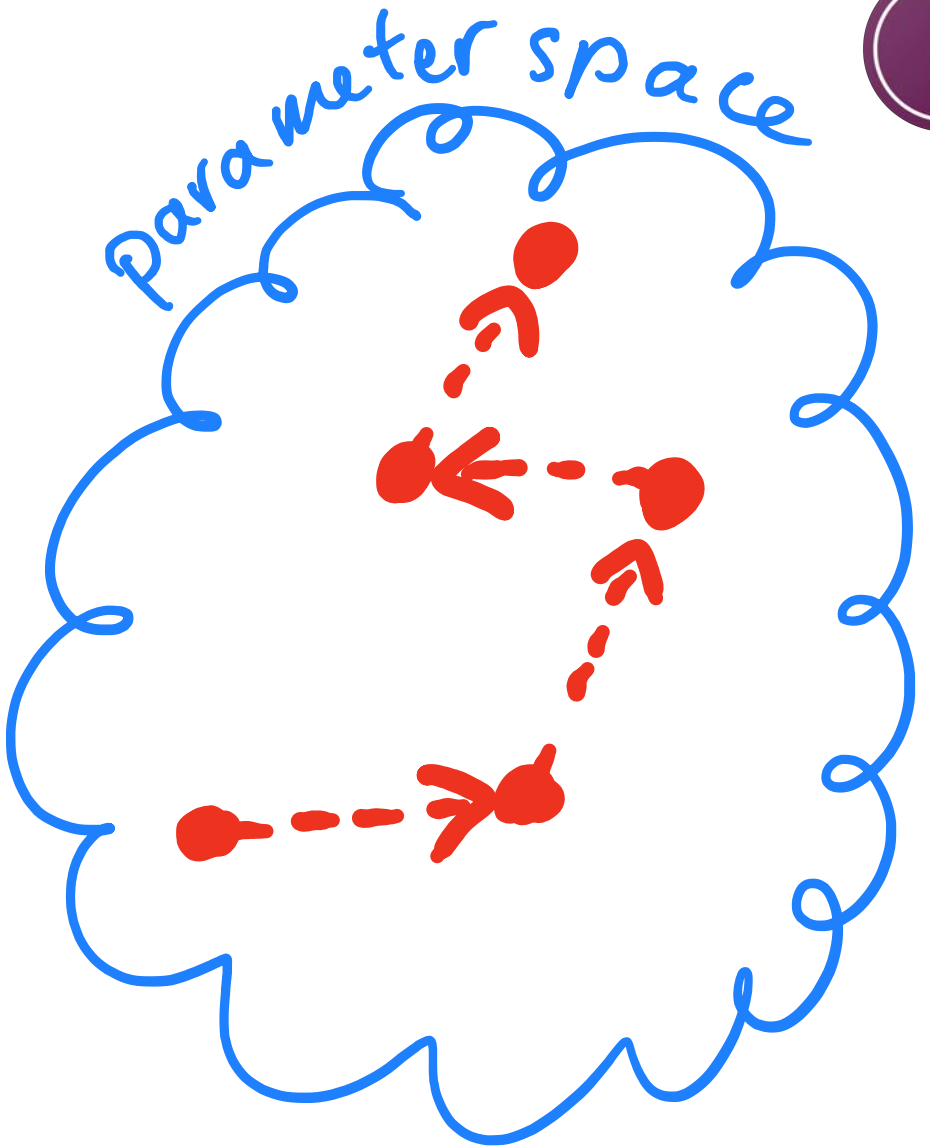
Xebia

# There is more!!


Xebia

# Learning

- Every sprint (2 week cycle)
- Sequential
- Using partial information

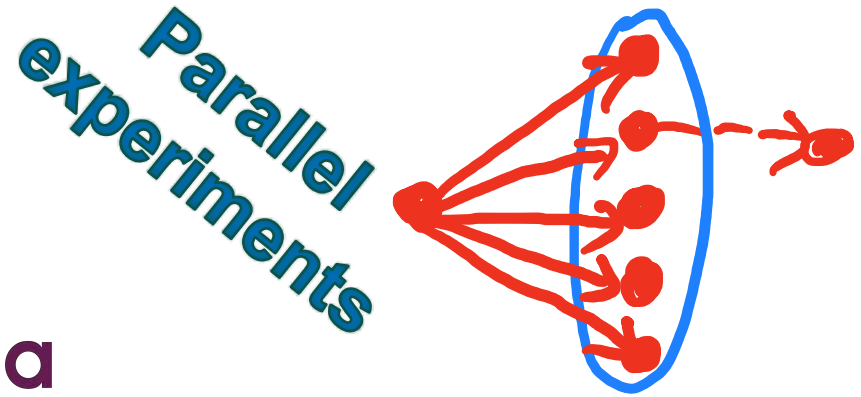


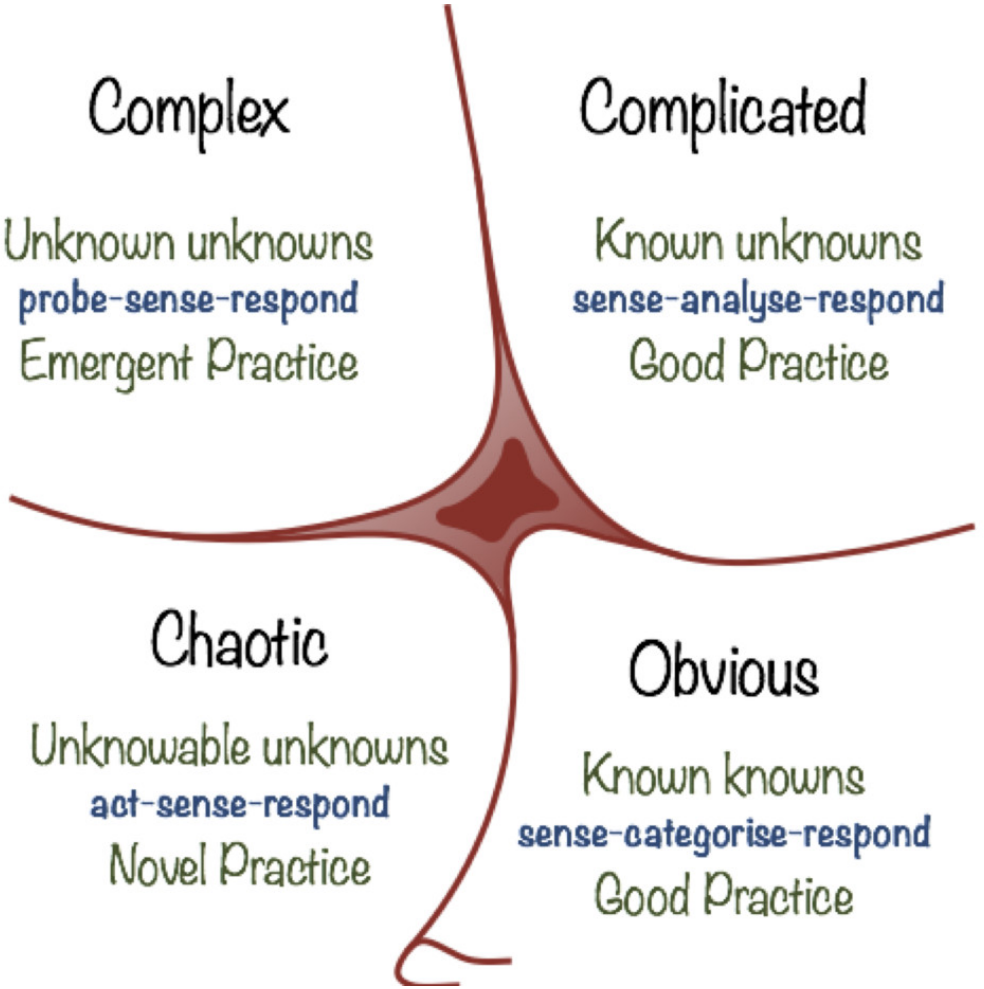

Xebia

# Cynefin framework

- Obvious & complicated
  - Sequential

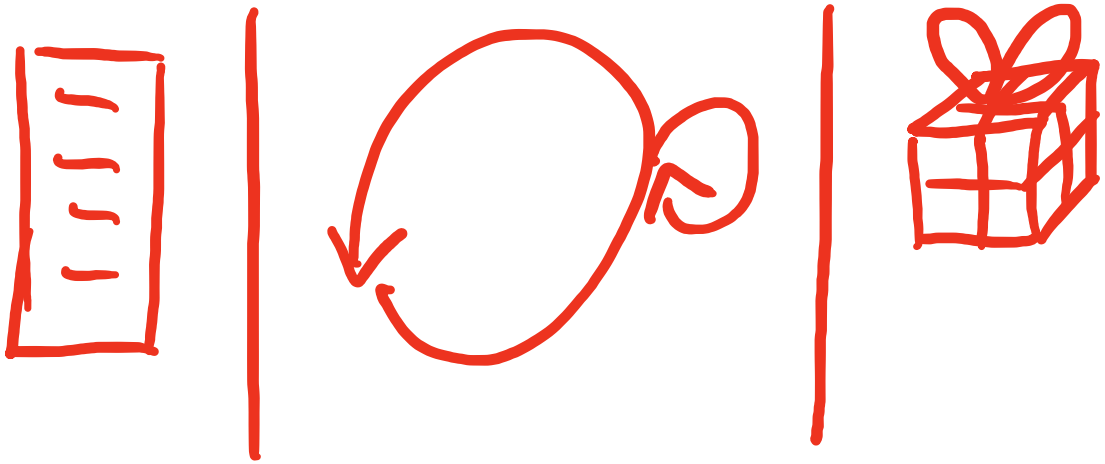
- Complex
  - Selectionism



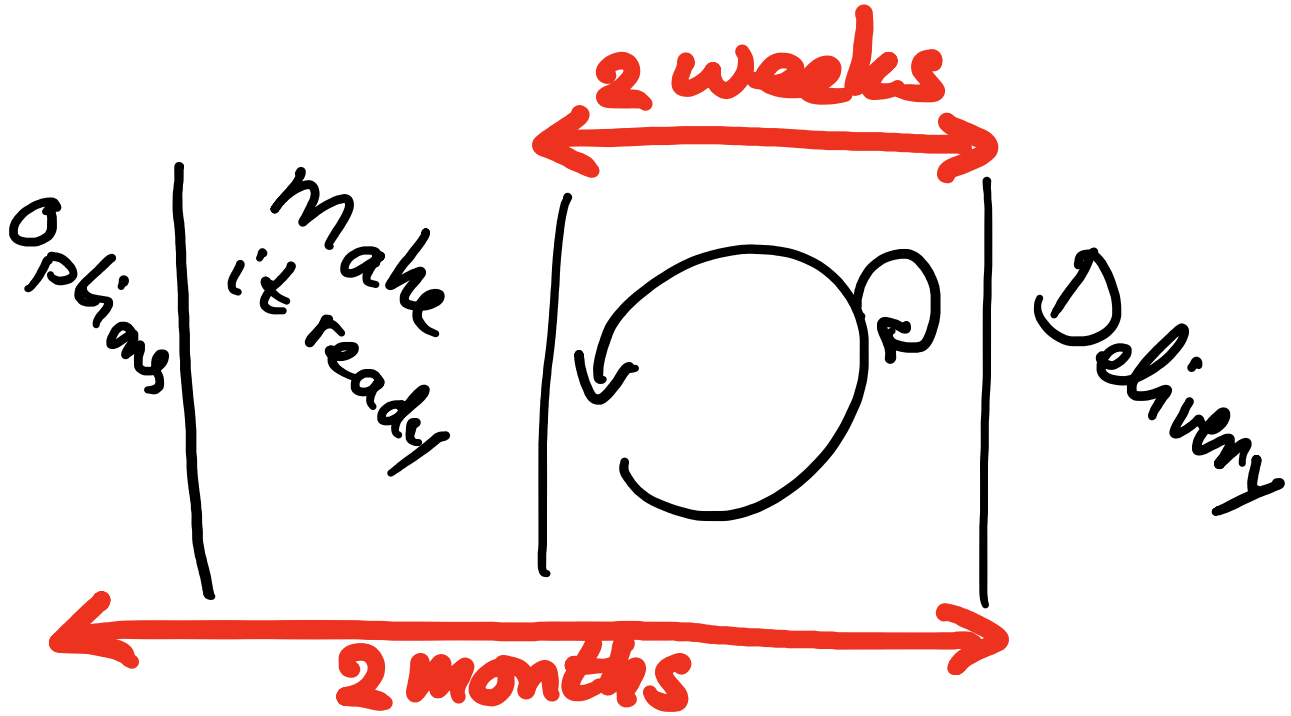

Xebia

# Partial information



Simple: Effect of improvement is known at the end of the sprint

Complex: Actual situation is more complicated and effect is known much later

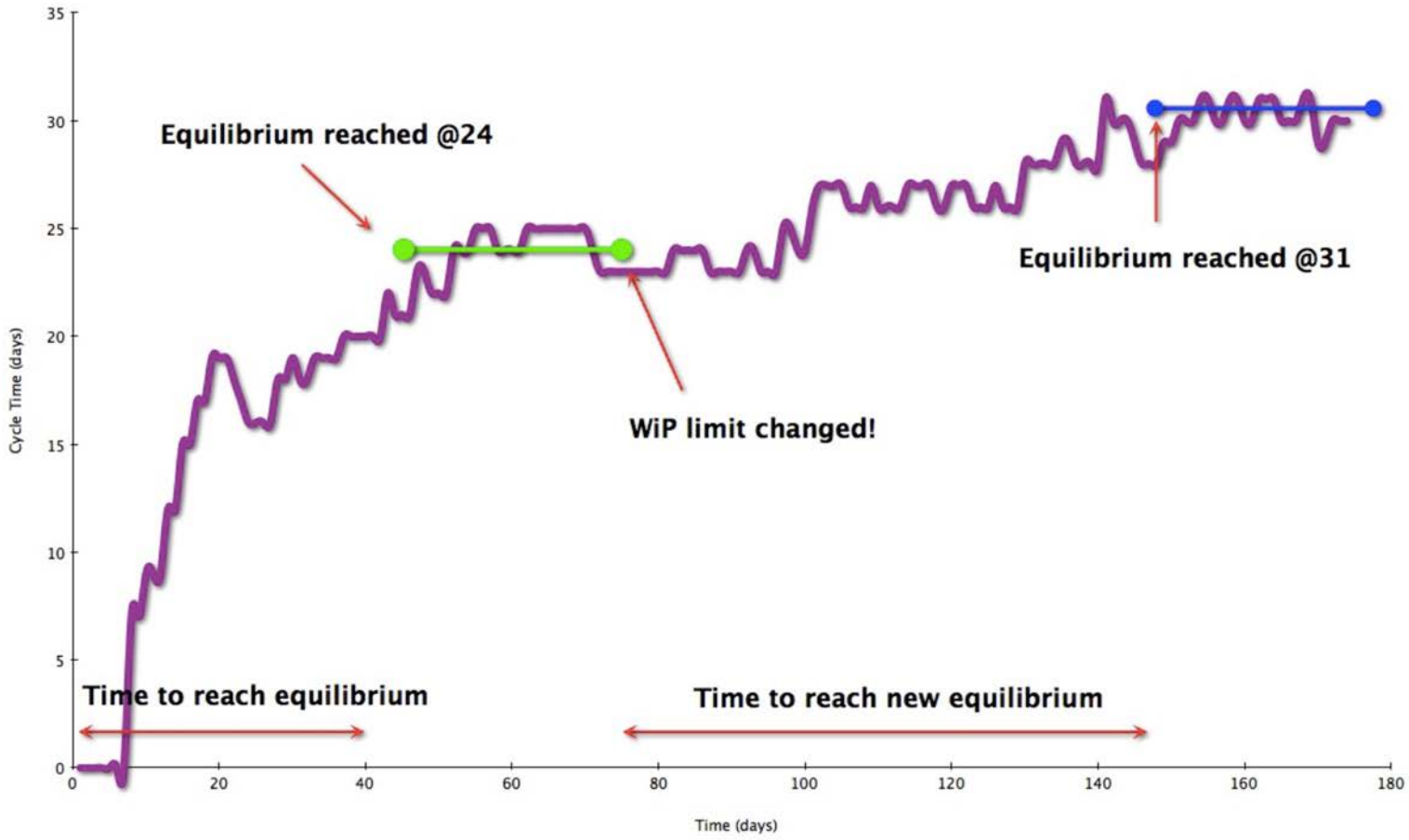



Xebia

# How much later?

Second and parallel experiments give unreliable results and based on partial understanding of the system

Limited number of learnings and improvements



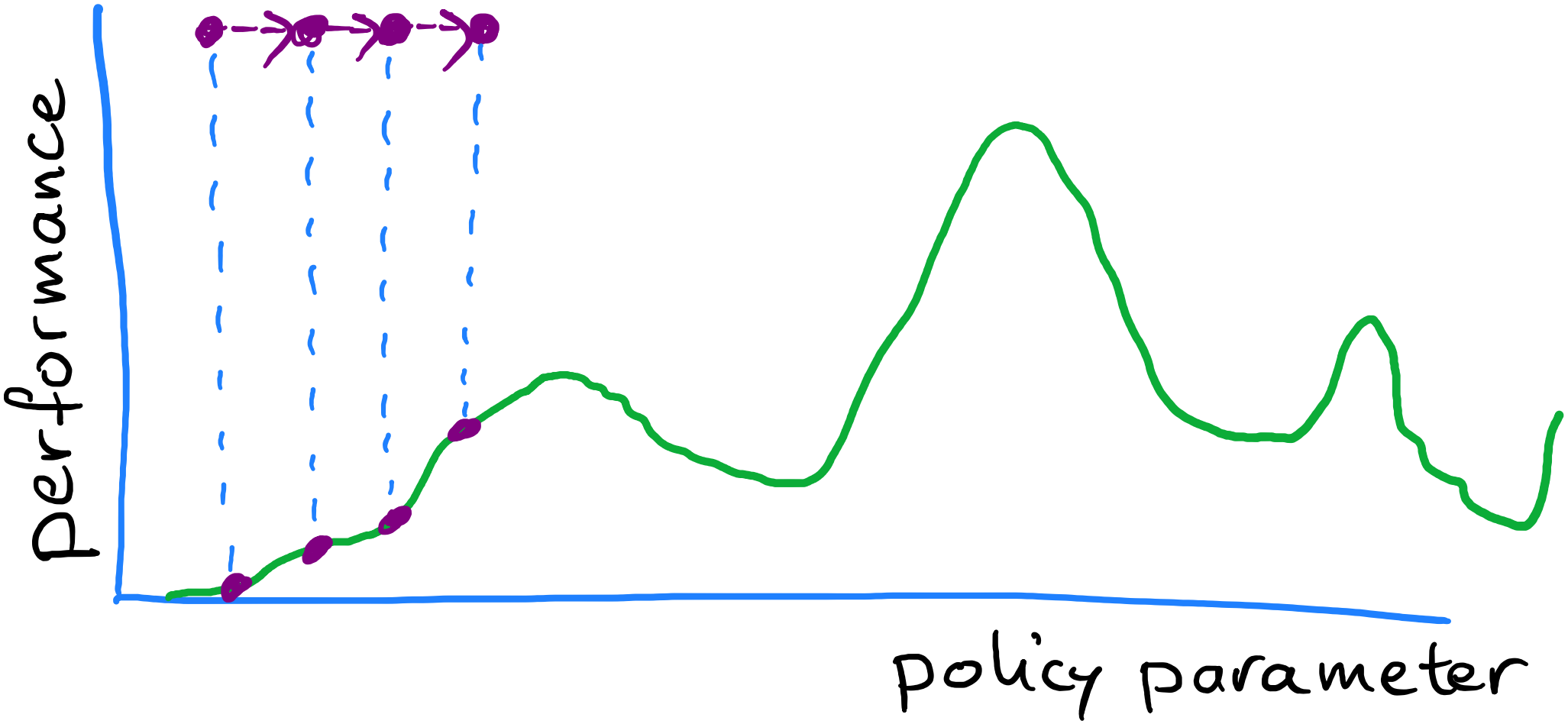

Xebia

# So, why should we care?

Walking the fitness landscape 

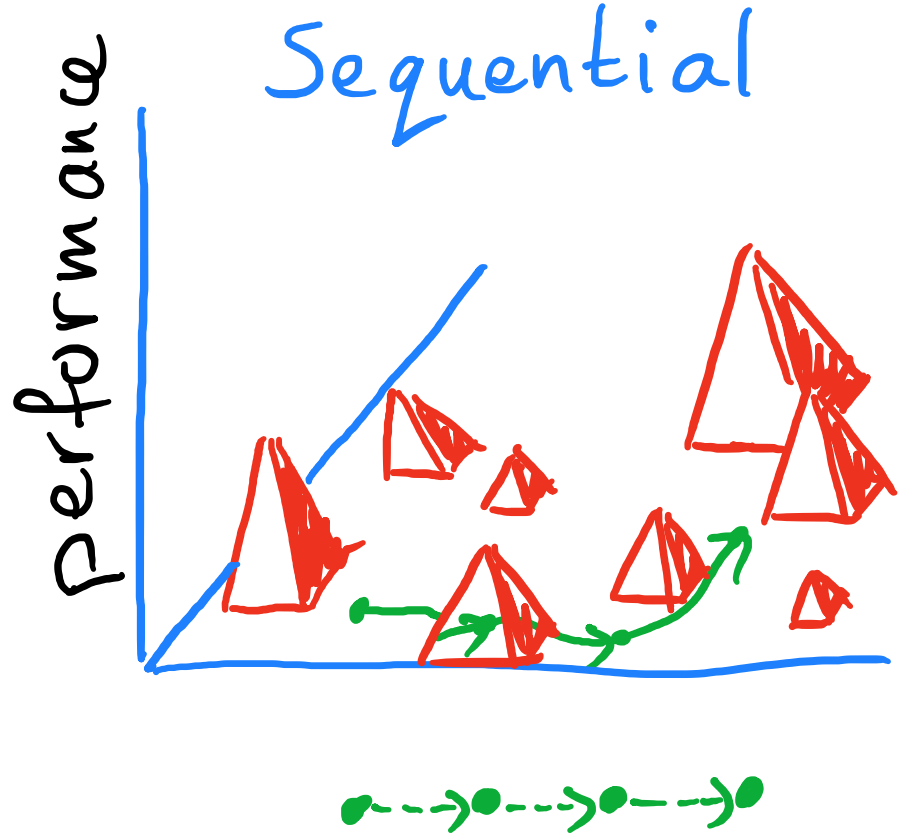
Xebia

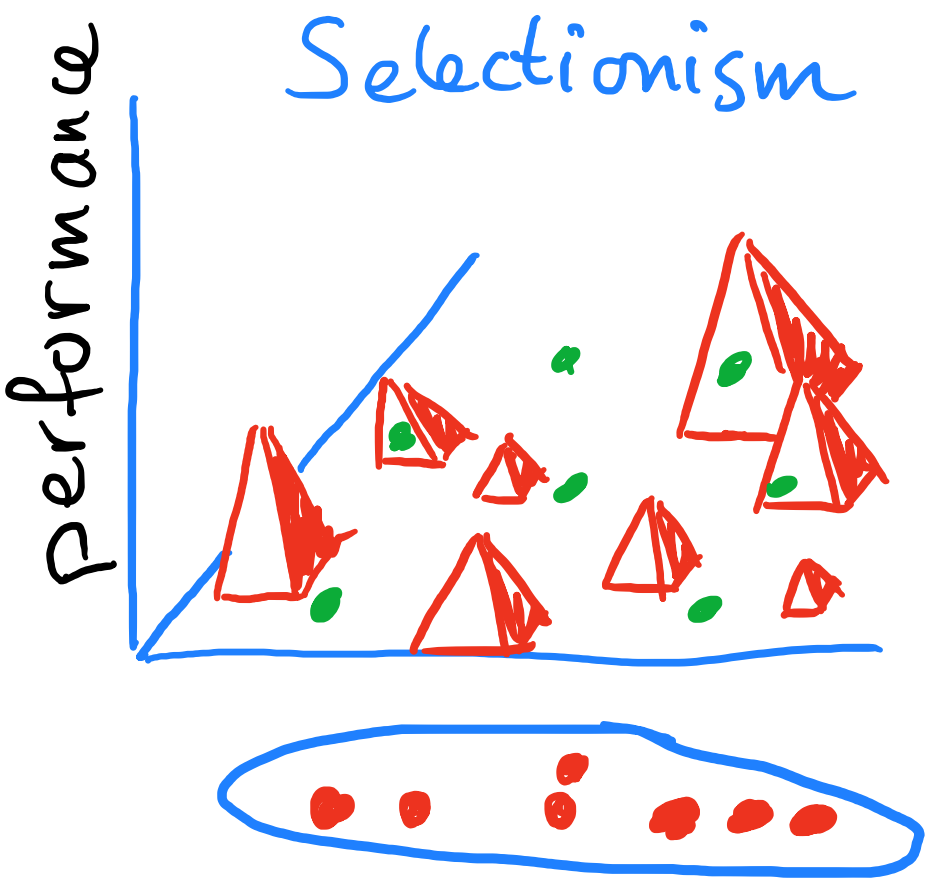
# Fitness landscape




Xebia

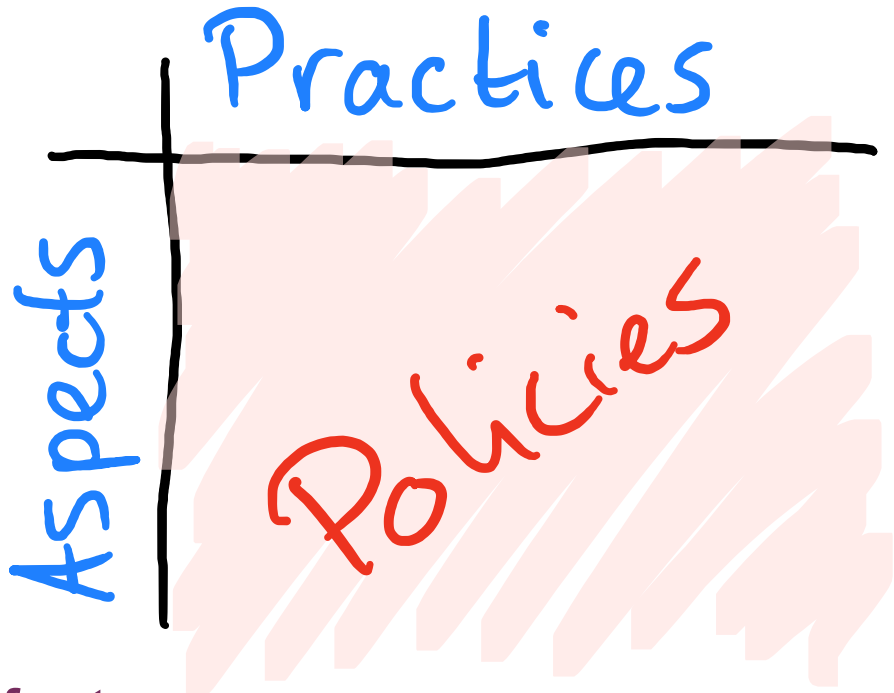
# Fitness landscape: more policies




Xebia

# More complicated




- Aspects
  - Up & downstream
  - More services & dependencies
  - Types of work
  - Scaling work item, eg tasks, stories, features

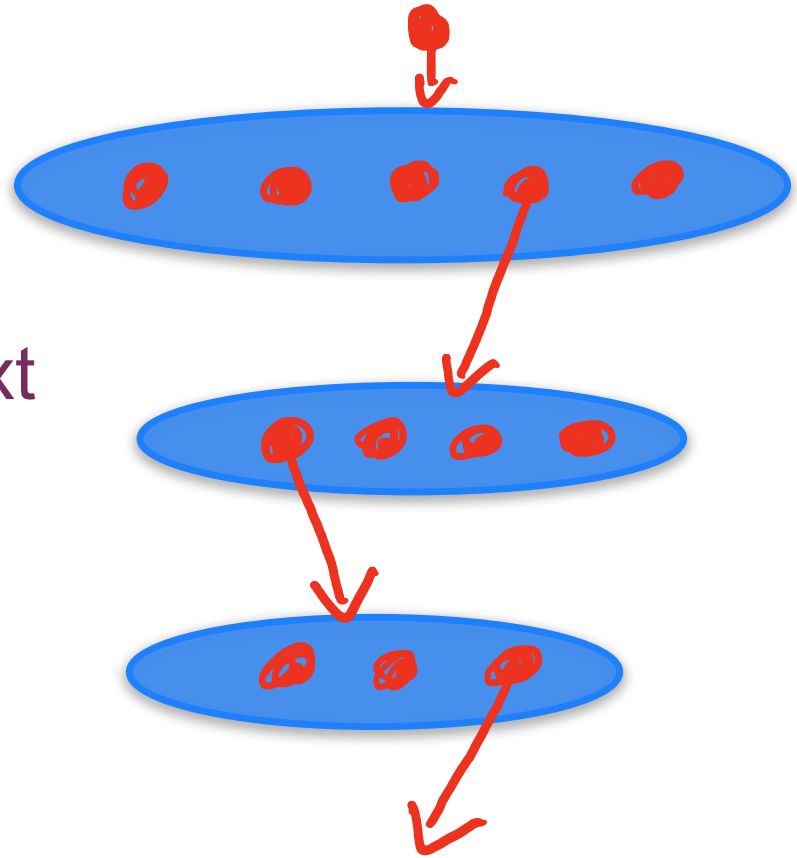
- 6 Practices of the Kanban Method
  - Visualize
  - Limit wip
  - Manage flow
  - Make policies explicit
  - Implement feedback loops & improve collaboratively

Xebia

# Learning in complex environments

- We need a combination of sequential and selectionism

- Run parallel experiments
- Choose the best outcome
- …as starting point for the next



Xebia

# Learn faster

- Use simulations! Especially Monte Carlo based methods
- Examples



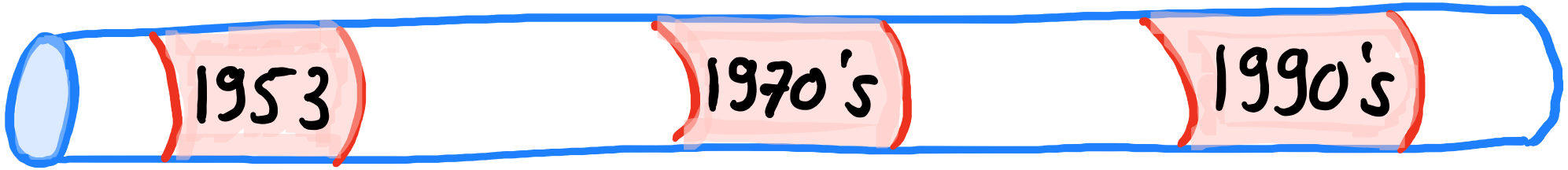
Enterprise Services Planning

Xebia

# Monte Carlo Methods

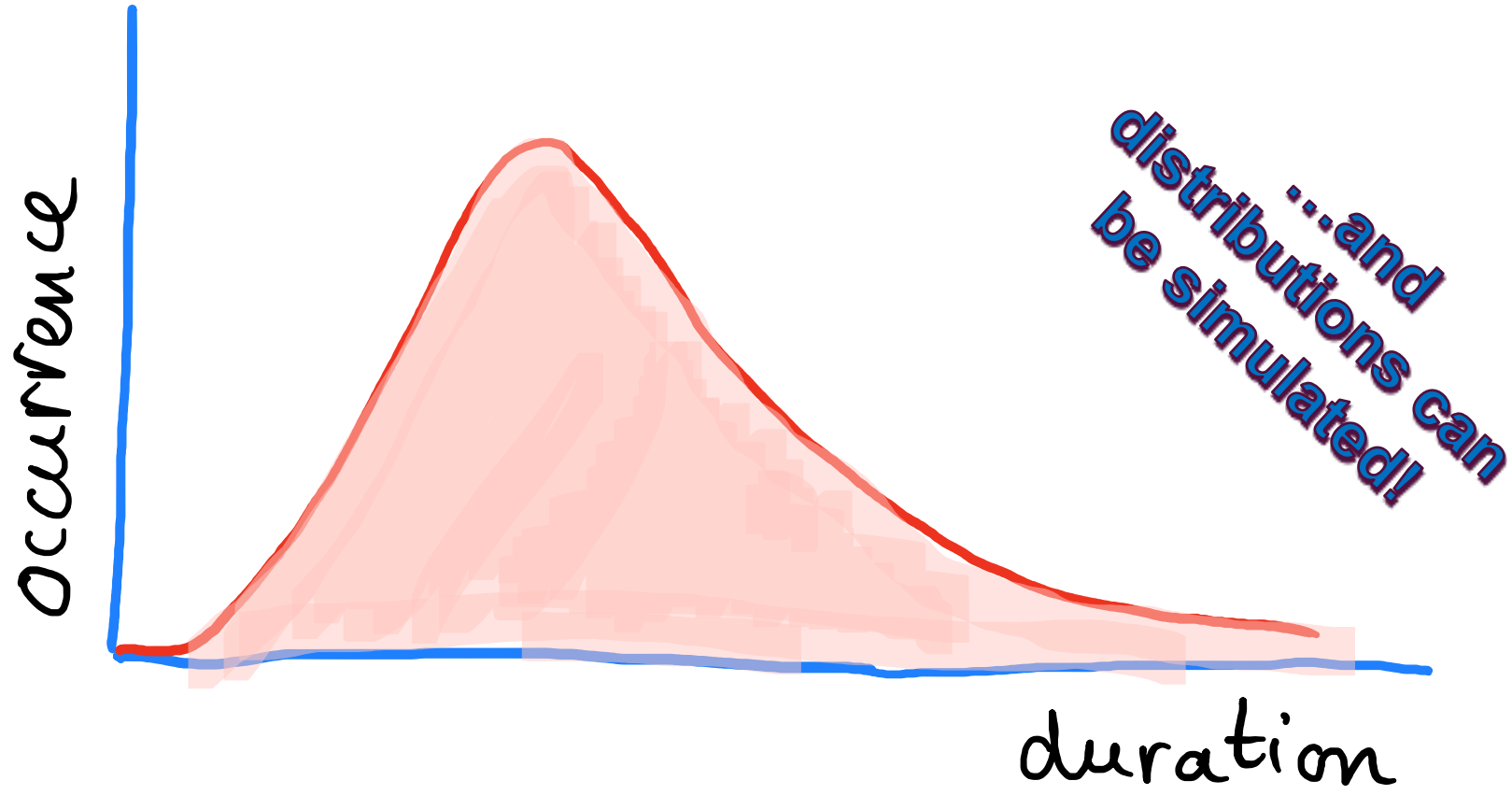
Xebia

# History

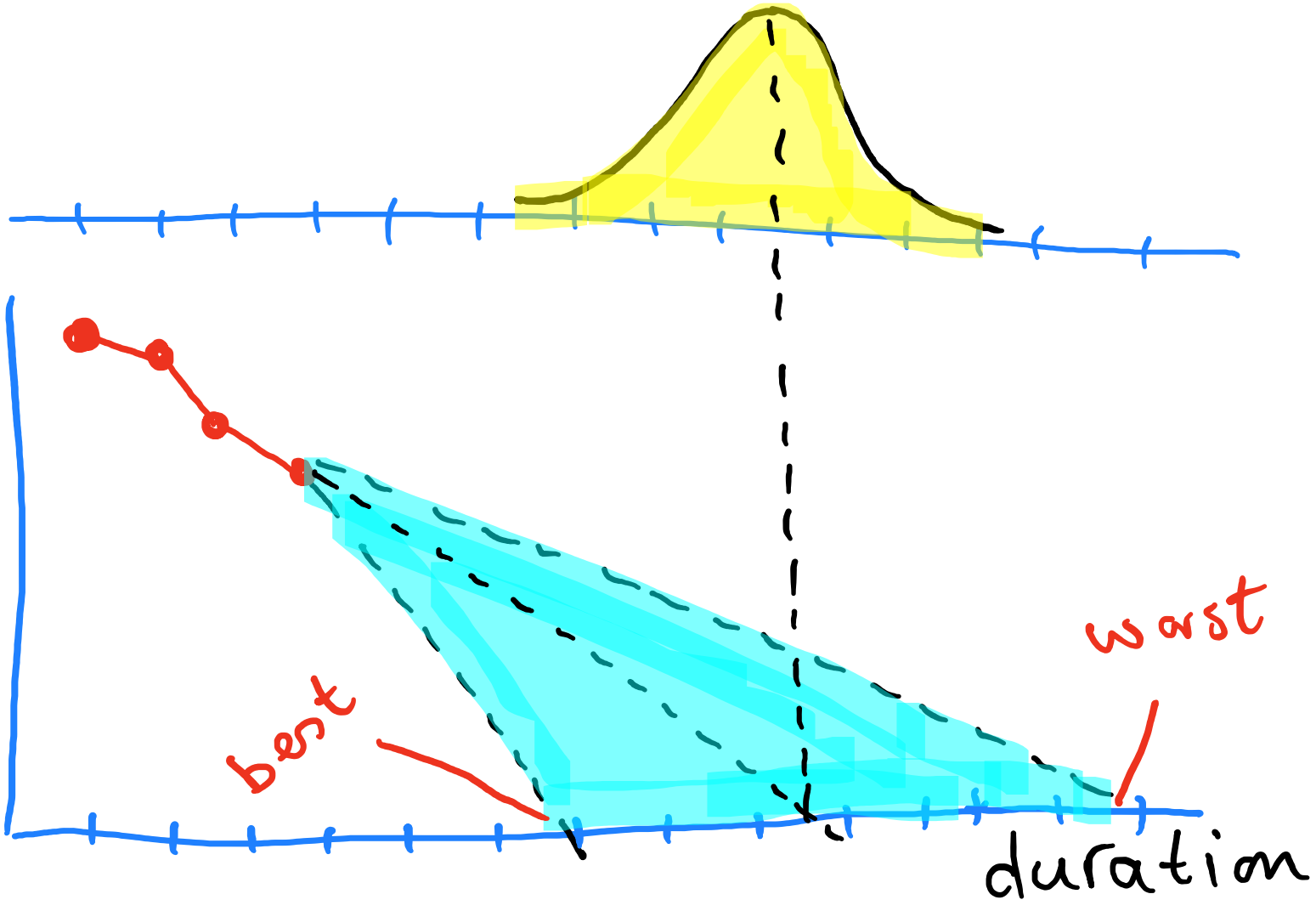

Xebia

# Tasks are distributions



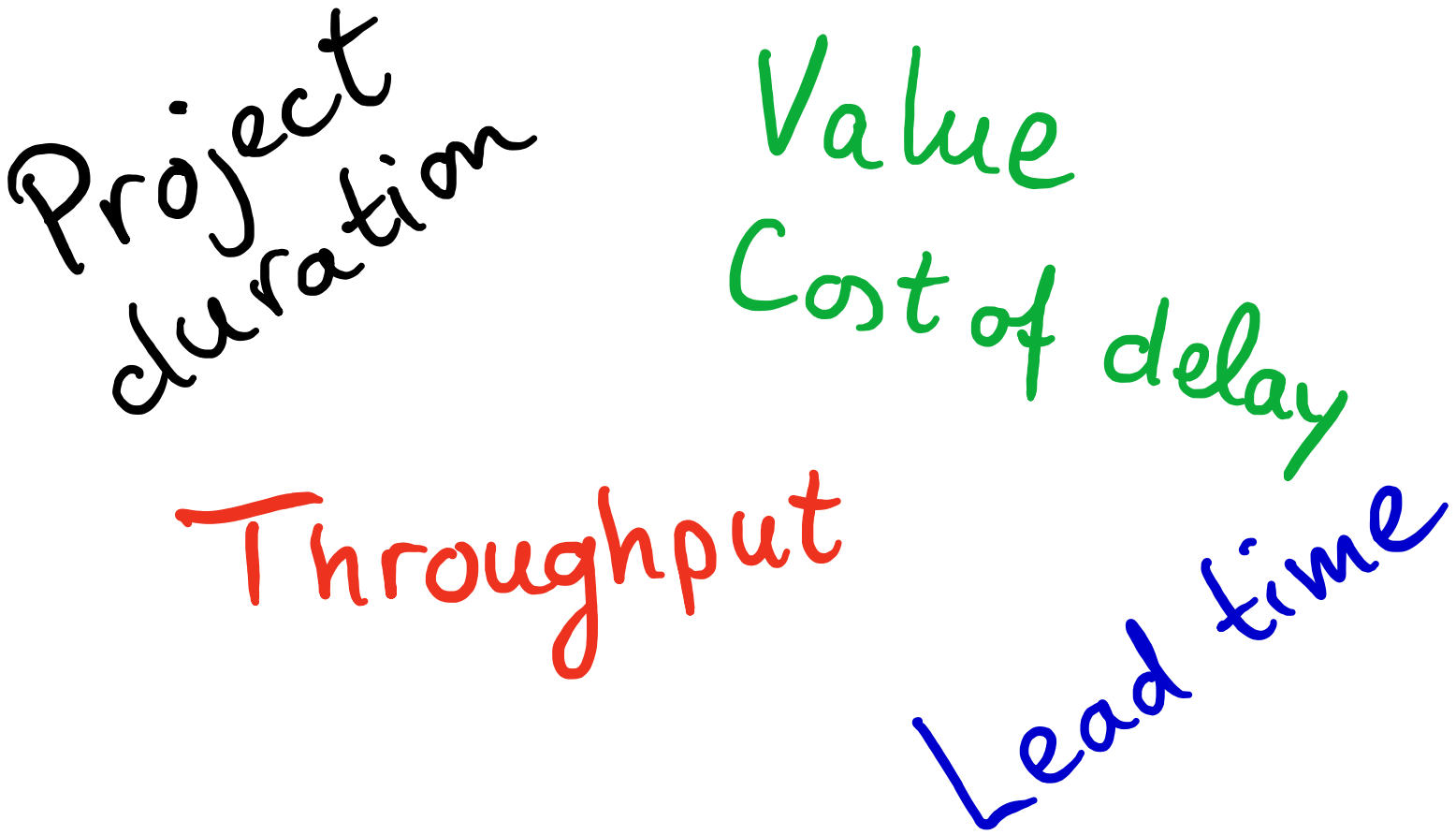

Xebia

# Forecast project duration




Xebia

# What are we trying to optimize?




Xebia

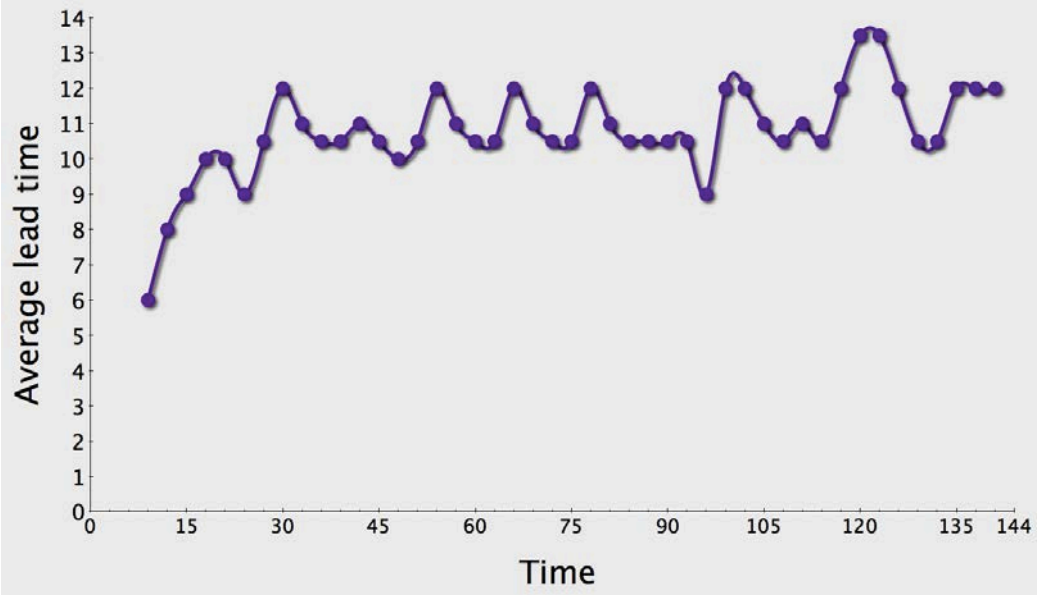
# Let's do some simulations

Xebia

# Fun with policies! GetKanban


Xebia

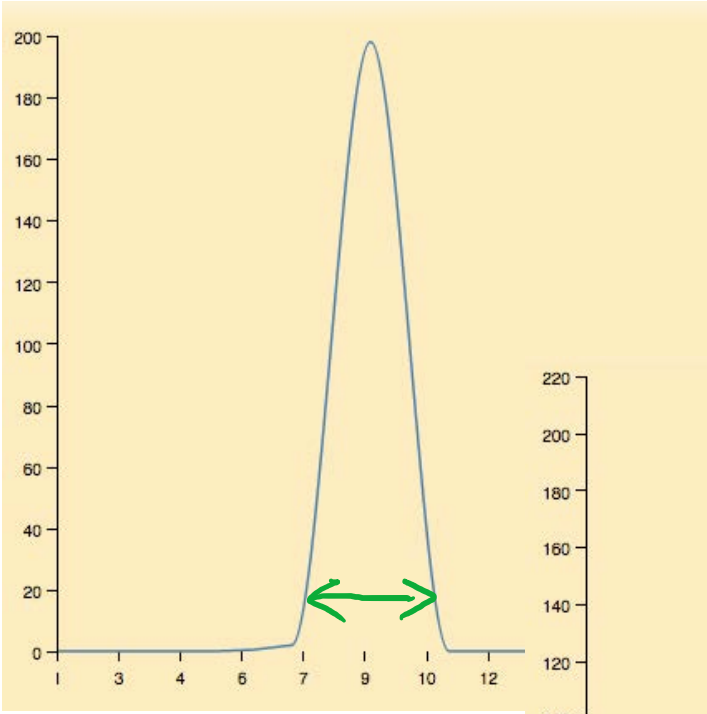
# Policies: wip limits



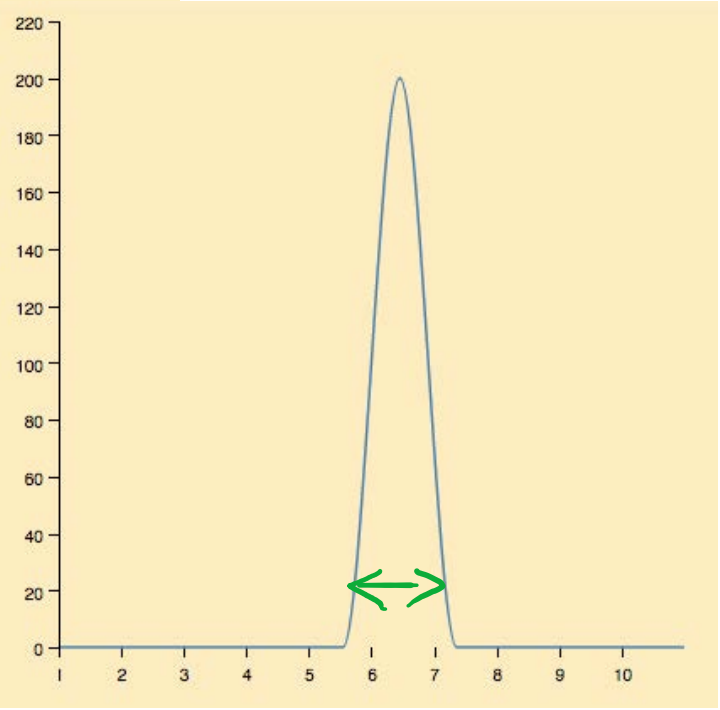

One of many possible runs

wip : 4-2-4-3

Monte Carlo simulation 200 runs



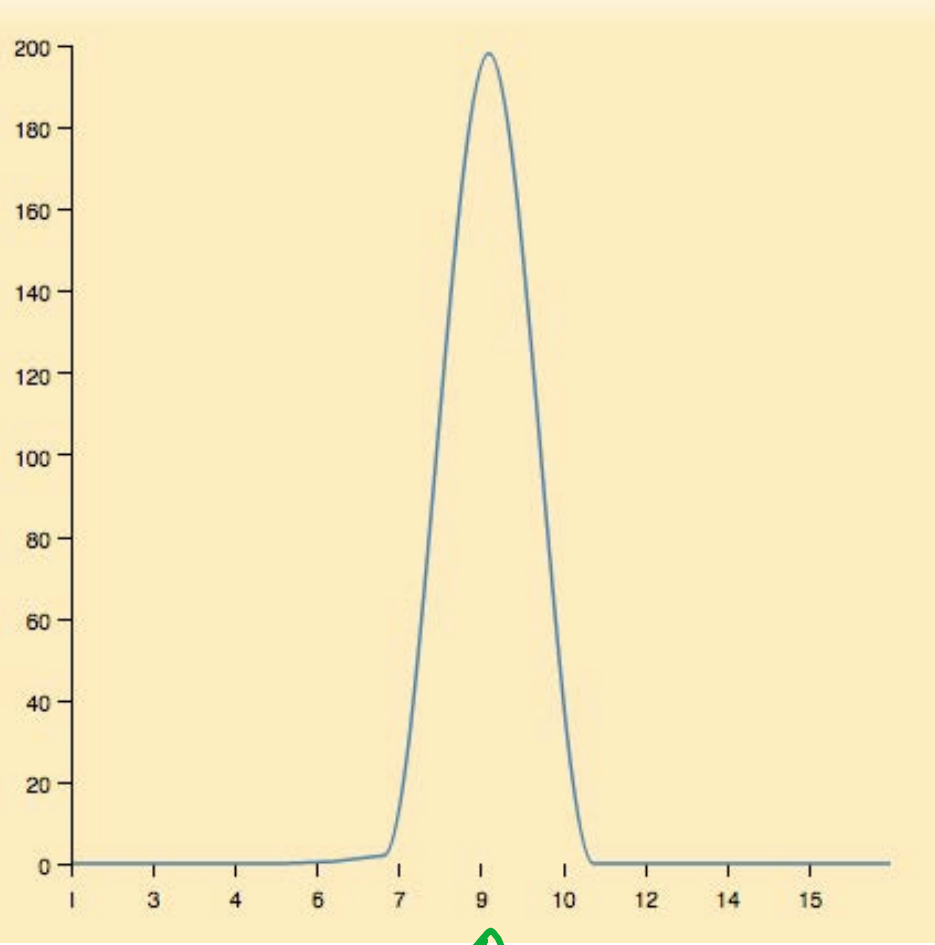
3-2-2-3

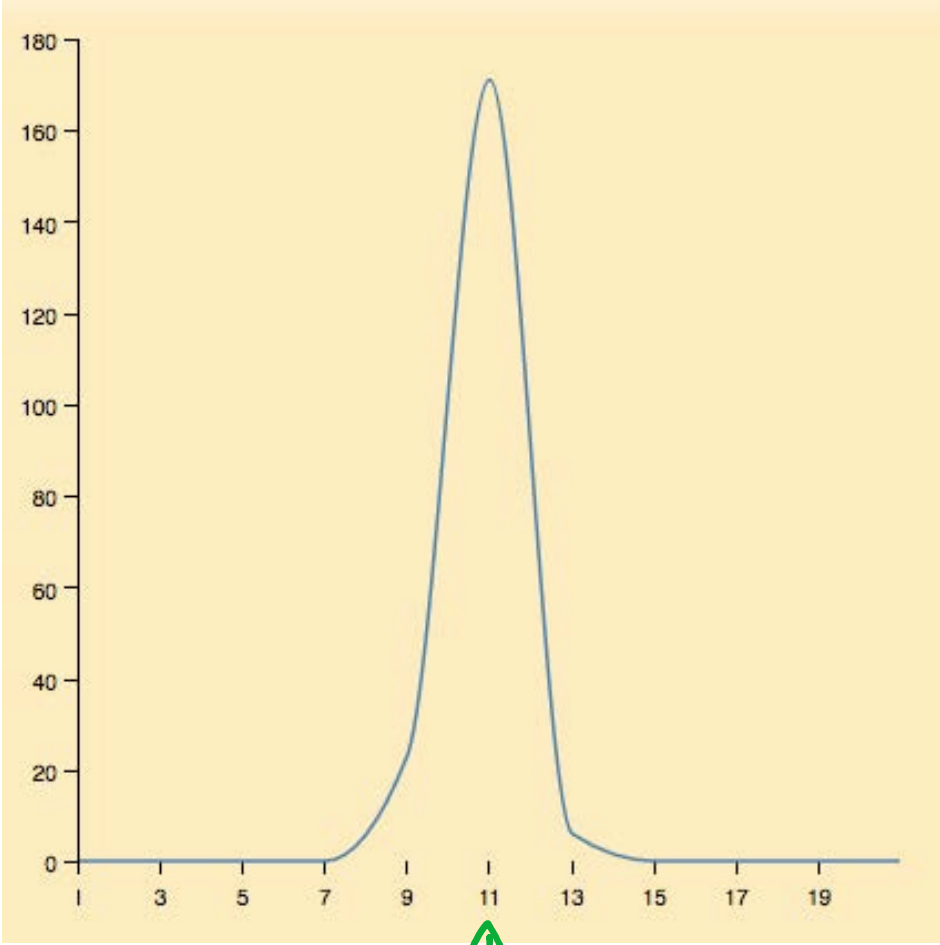


Xebia

# Policies: Carlos

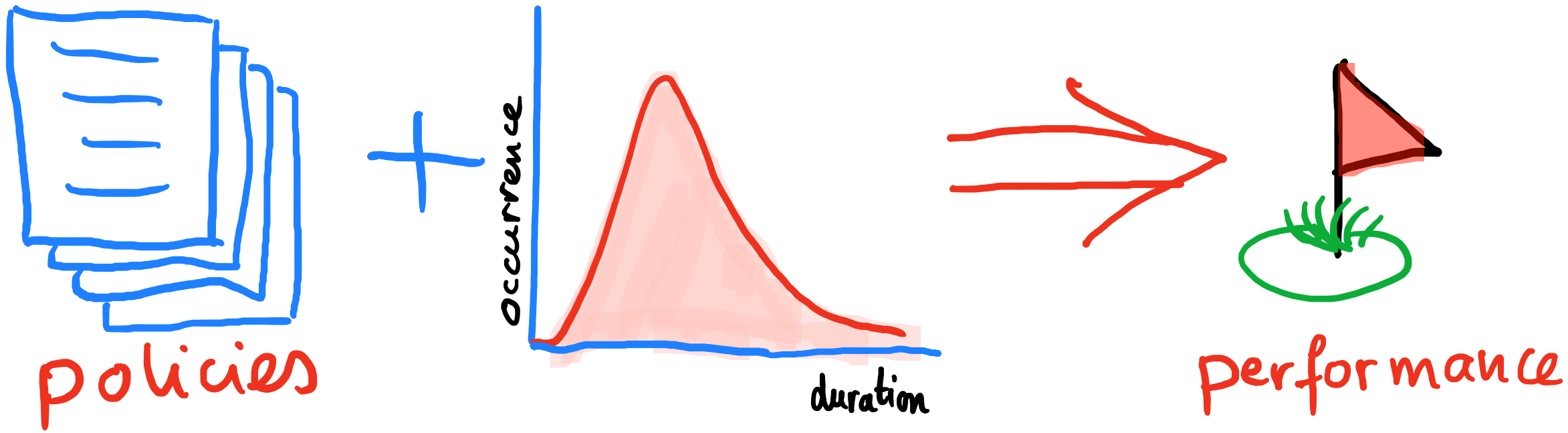
Carlos




Xebia

# Benefits of Monte Carlo




Xebia

# Monte Carlo: errors caused by

- Lack of understanding
  - Unknown policies
  - Policies wrongly modeled
  - Policies not correctly followed in practices (lack of discipline)
  - Changing environment
- Monte Carlo
  - Statistical, outliers
  - Artifacts from the used algorithm/method
  - Random number generators

Xebia

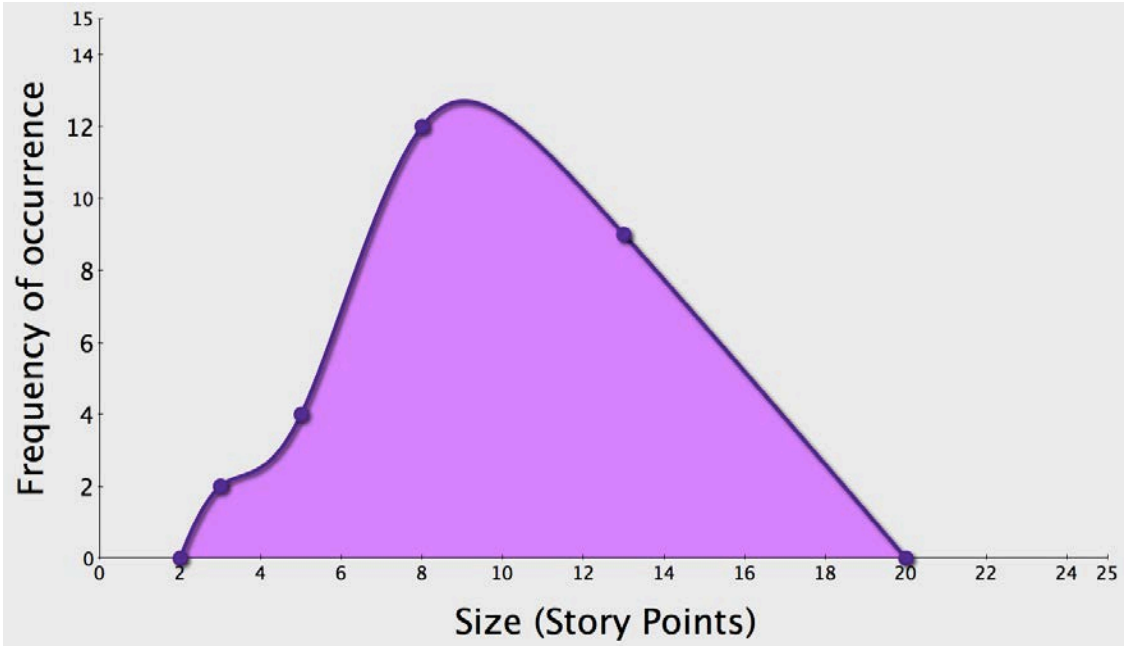
# Understanding the system

Relevant policies ›

Xebia

# Case: project duration

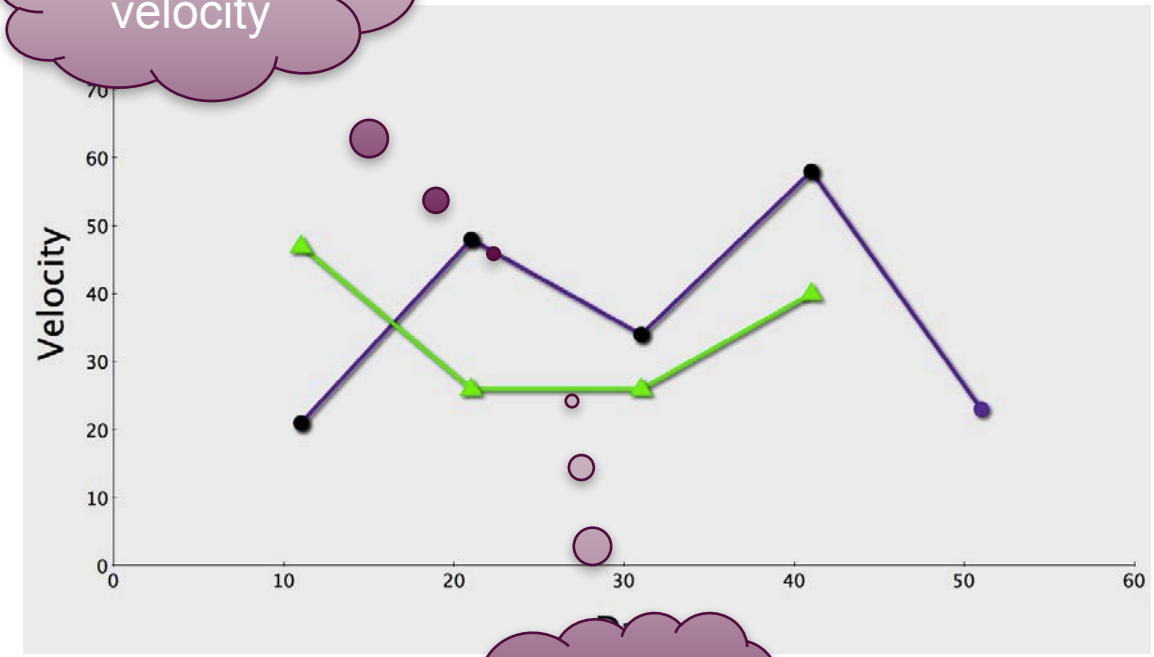
- Velocity: 47, 26, 26, 40 SP
- Sprint = 2 weeks
- Sprint backlog size = 60 SP



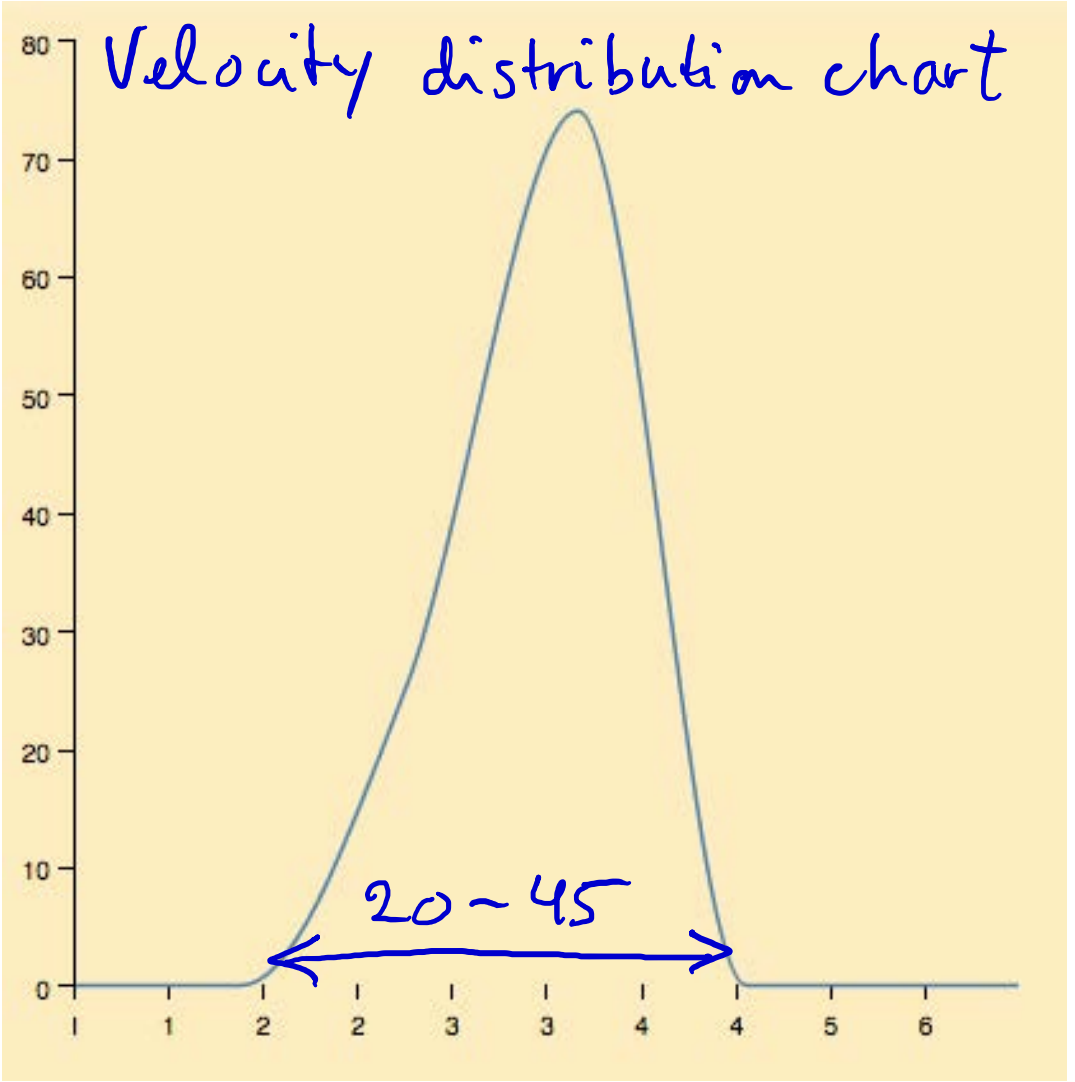

Xebia

# Case: understanding velocity

Simulated velocity




Team velocity




Xebia

# Relevant policies

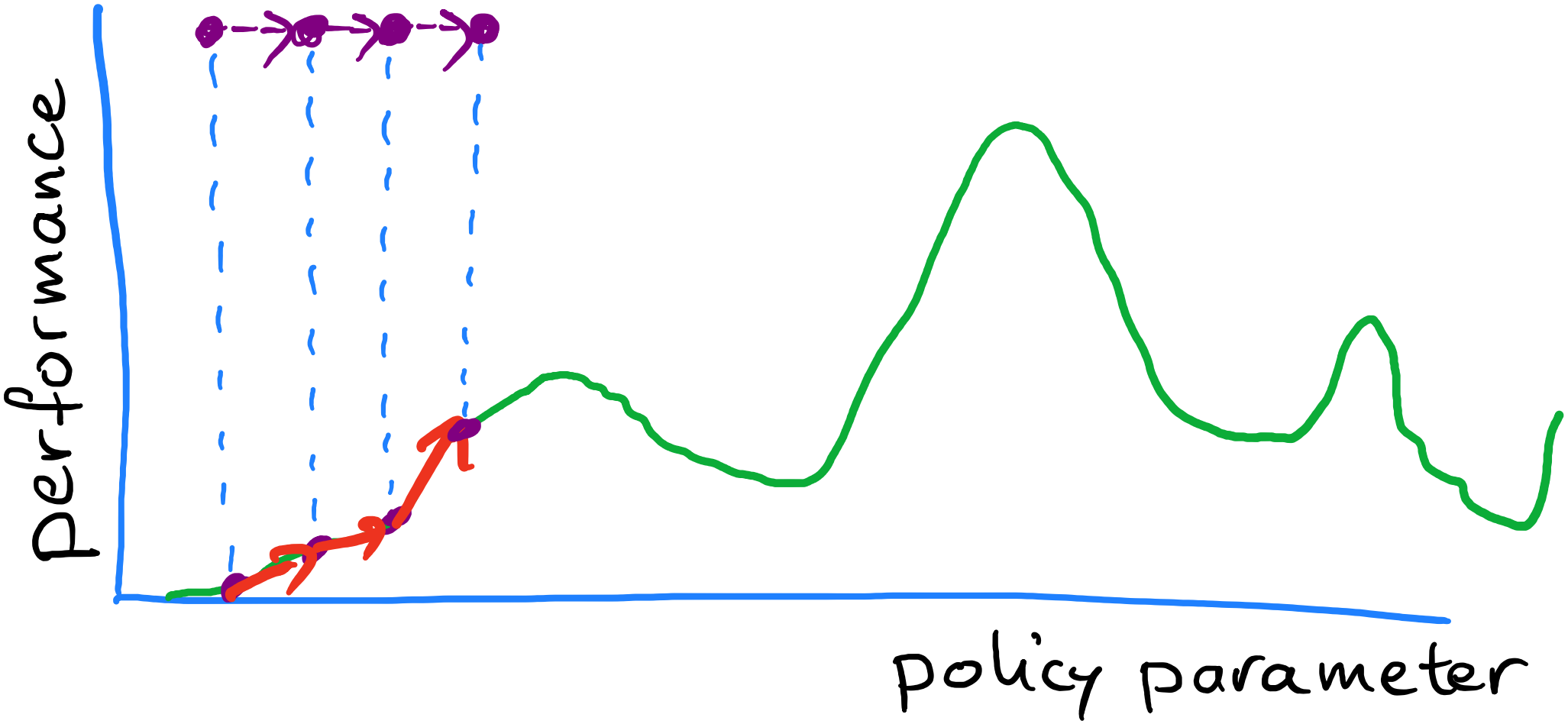
- Distribution of story sizes
- Distribution of work accomplished every day
- Sprint length, replenishment, and delivery frequencies
- Capability of the team (velocity)
- Dealing with unfinished stories
- Taking up additional work during sprint (replenishment)
- …..

# Optimal policies?

Walking the fitness landscape 

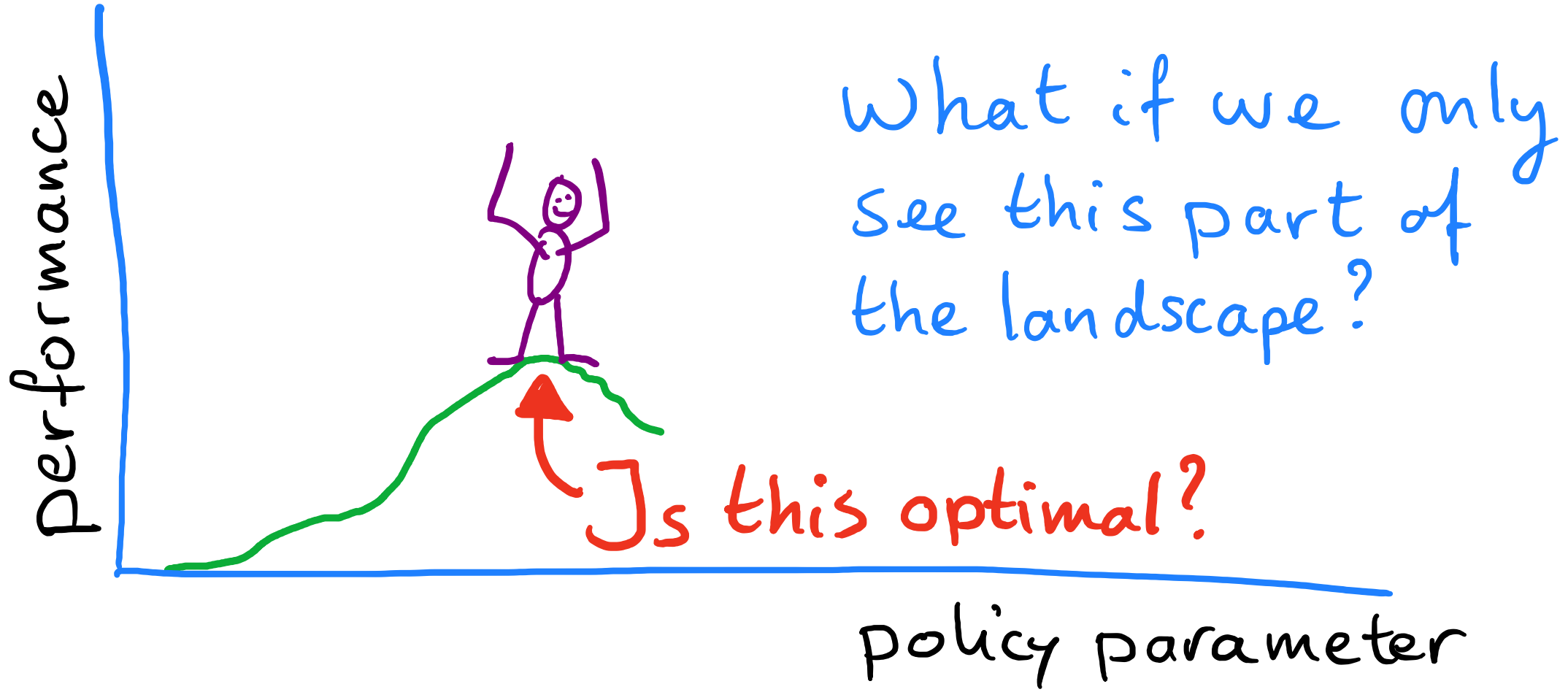
Xebia

# Fitness landscape



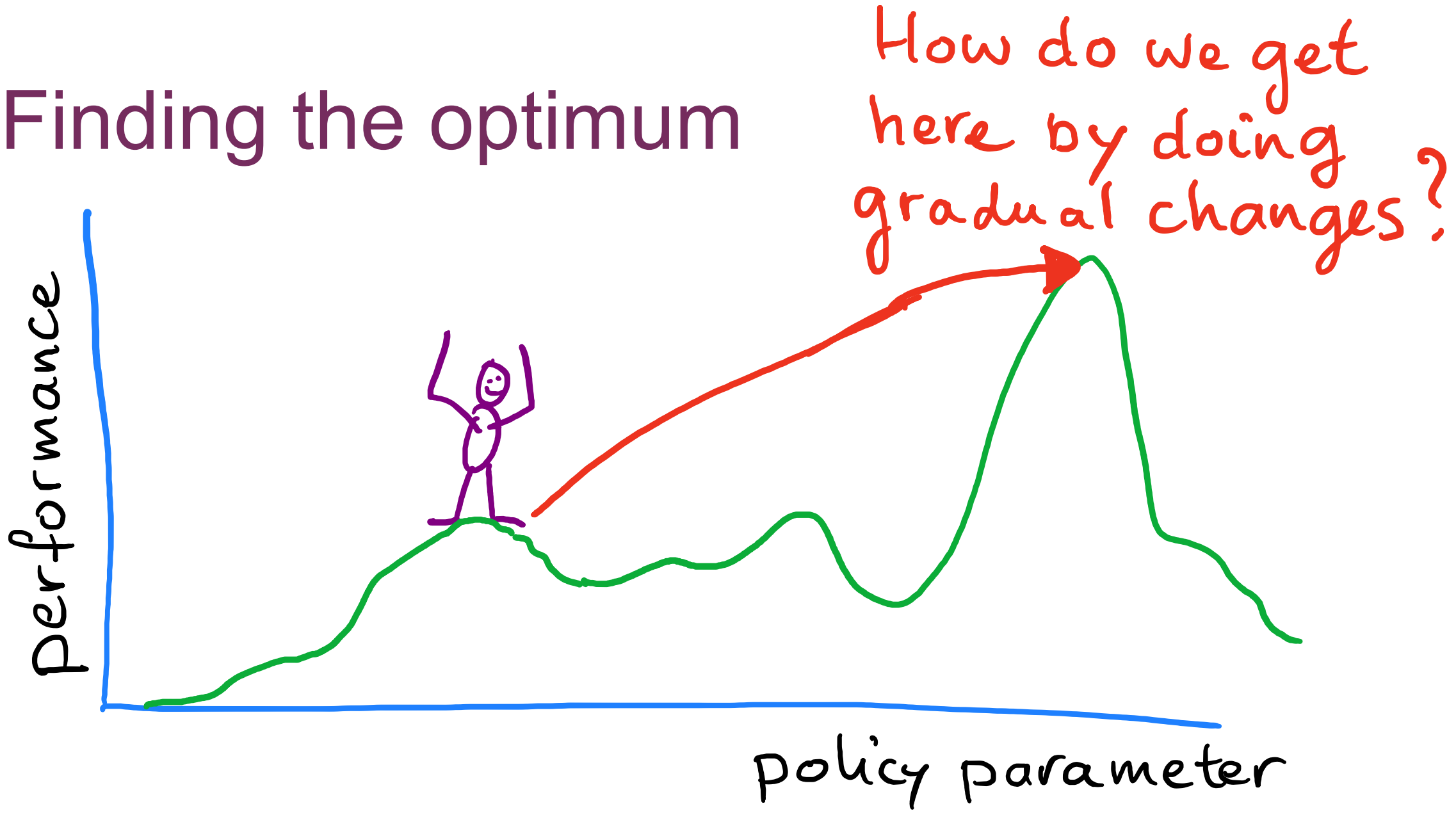

Xebia

# Finding the optimum



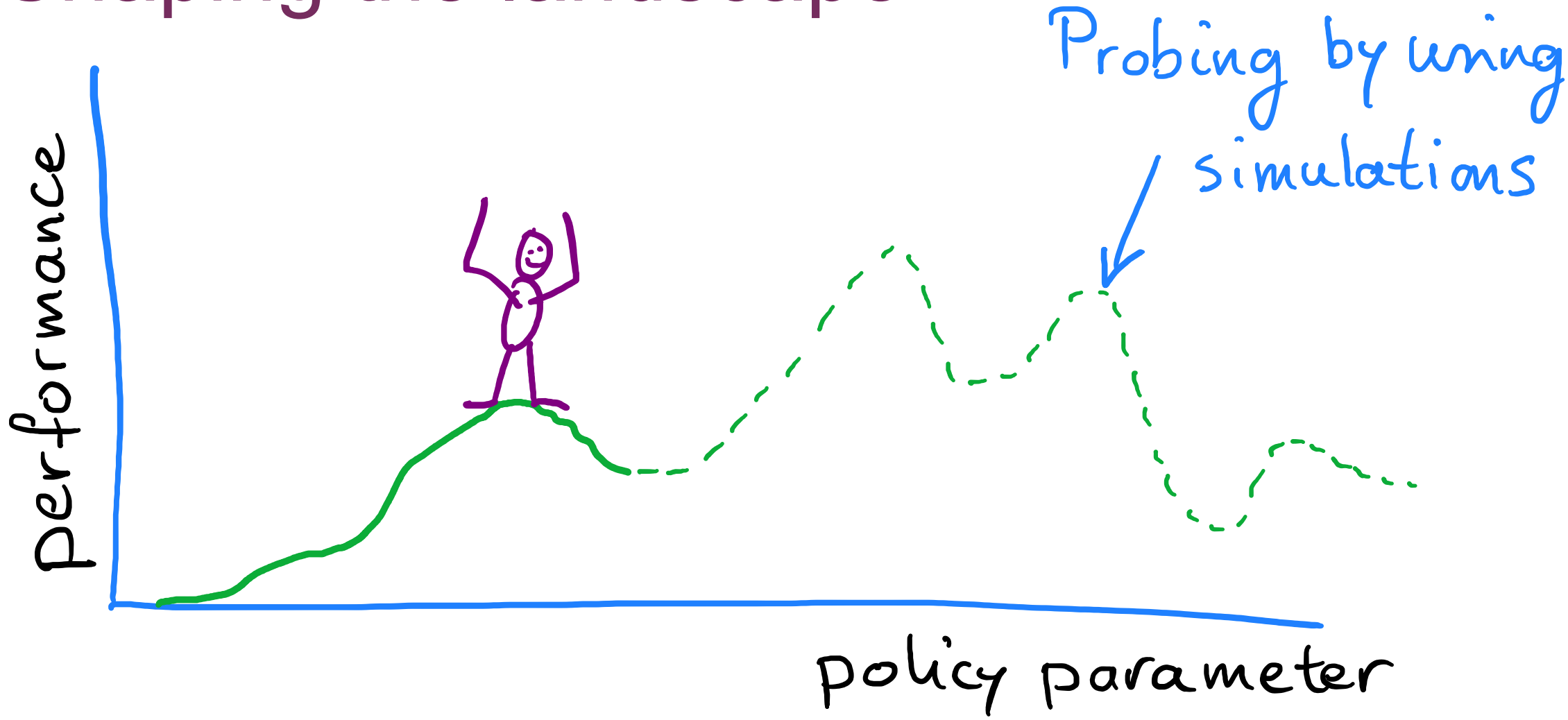

Xebia

# Finding the optimum




Xebia

# Shaping the landscape

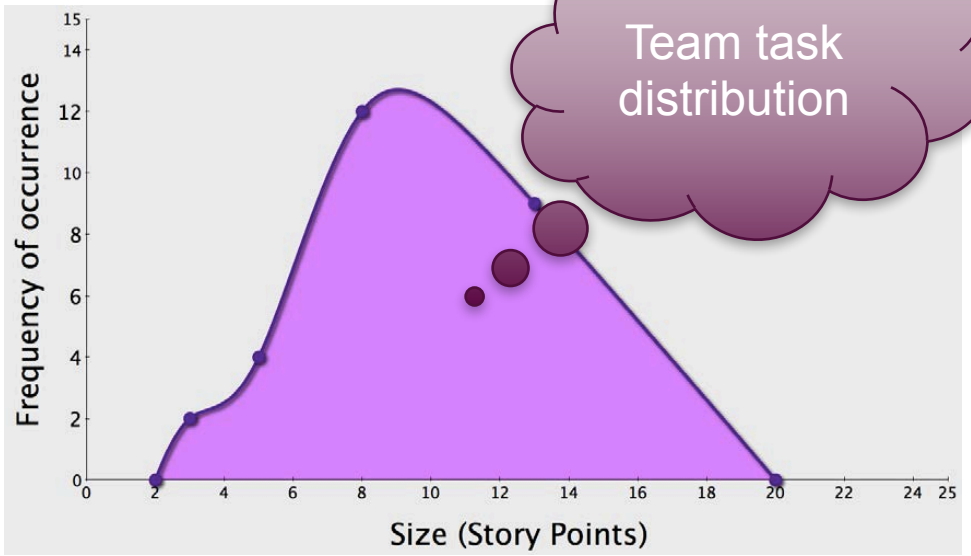


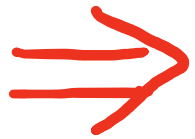

Xebia

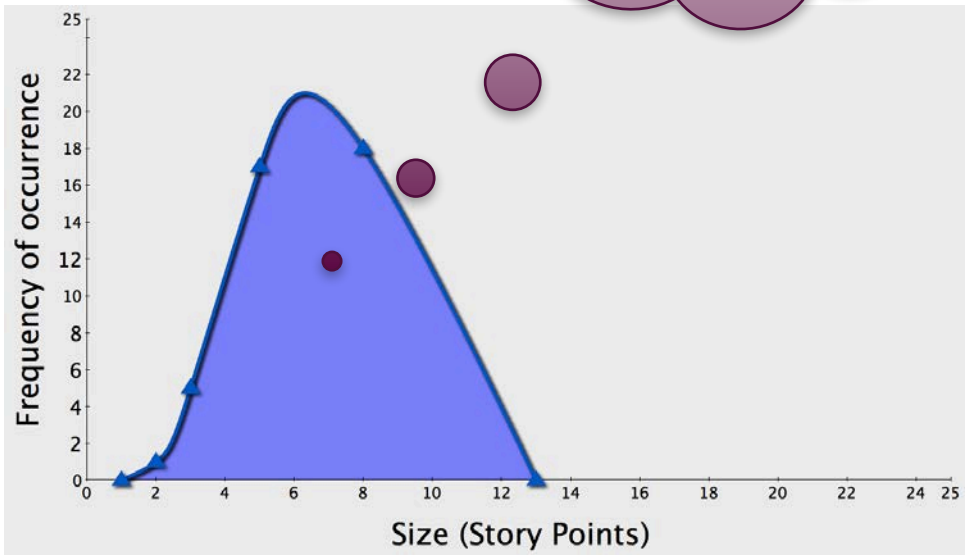
# Fitness landscape

- Issue of local optimum
  - How do you know whether you're in local optimum only?
  - How to get to the global optimum?
- With gradual steps, first results get worse, before they improve
- Use simulations to guide what will work
  - Reinforcement learning techniques, or
  - Monte Carlo methods

# Case: story size policy



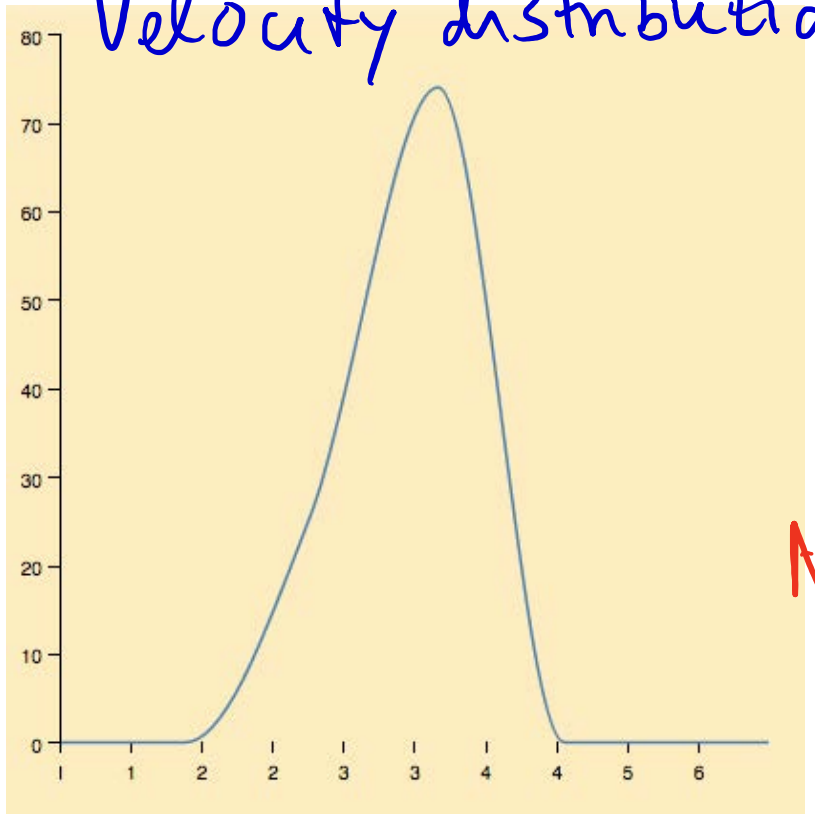



Policy : no more 13!!

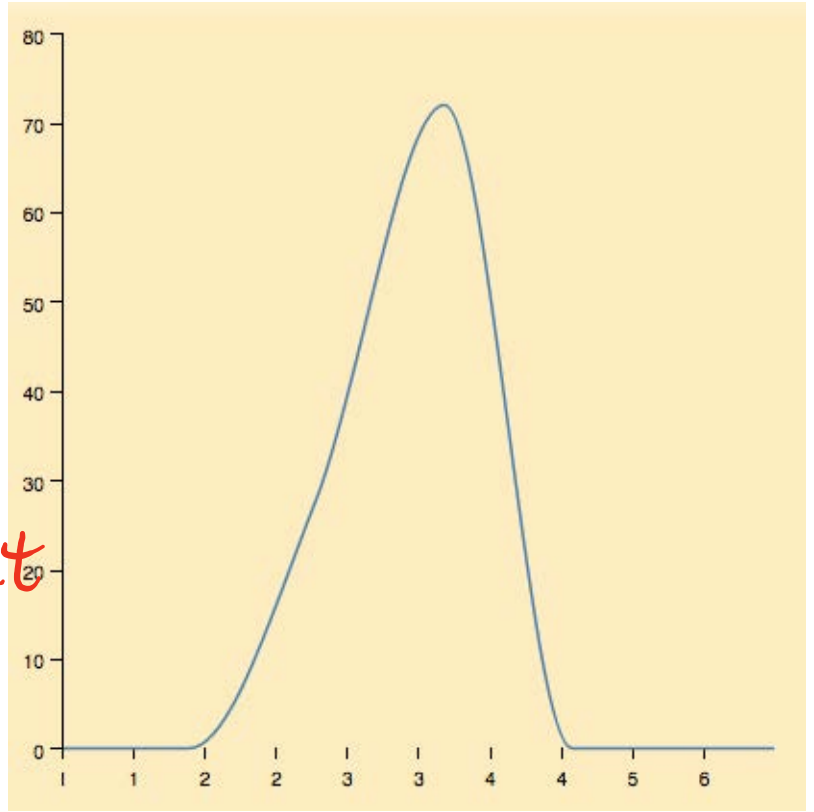
Xebia

# Case: 'story size < 13SP' policy

Velocity distribution chart

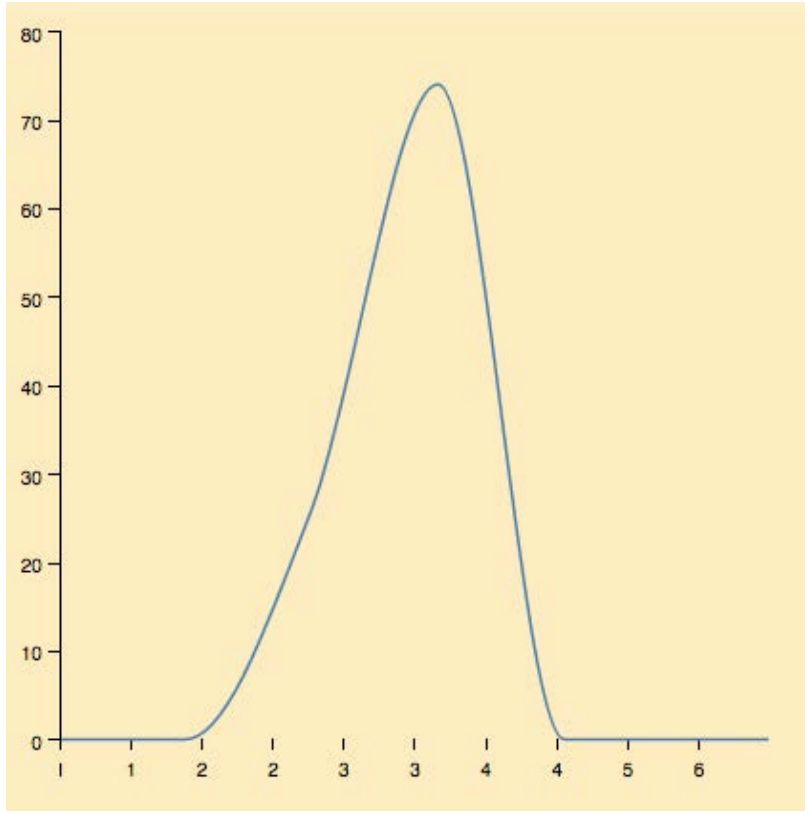



No significant improvement

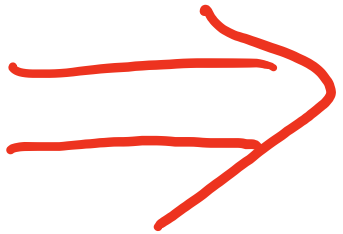


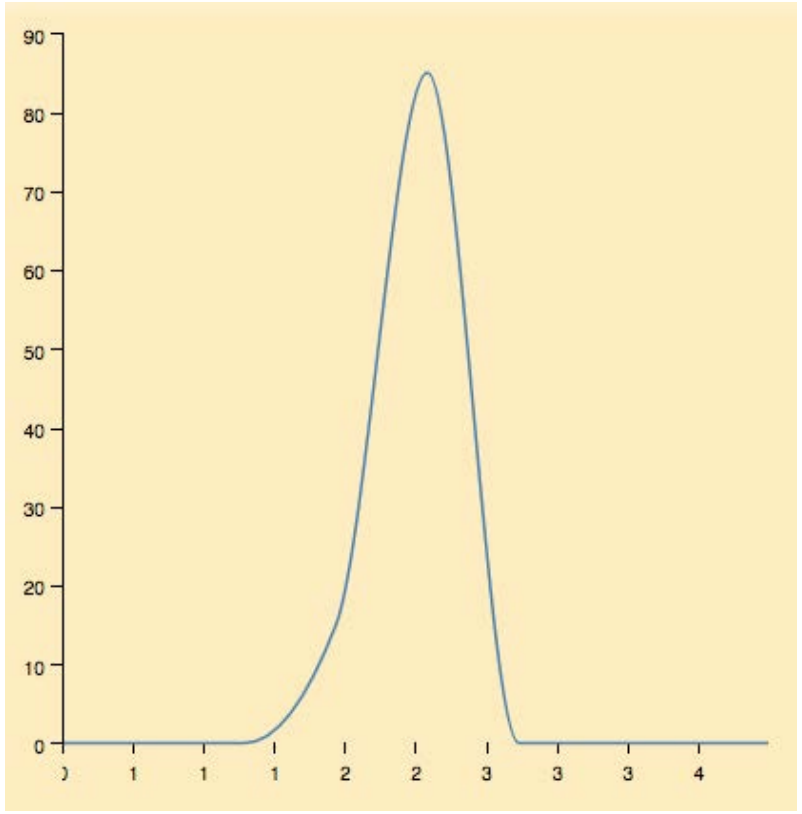

Xebia

# Case: 'no more than 4 stories' policy




velocity: 20-45

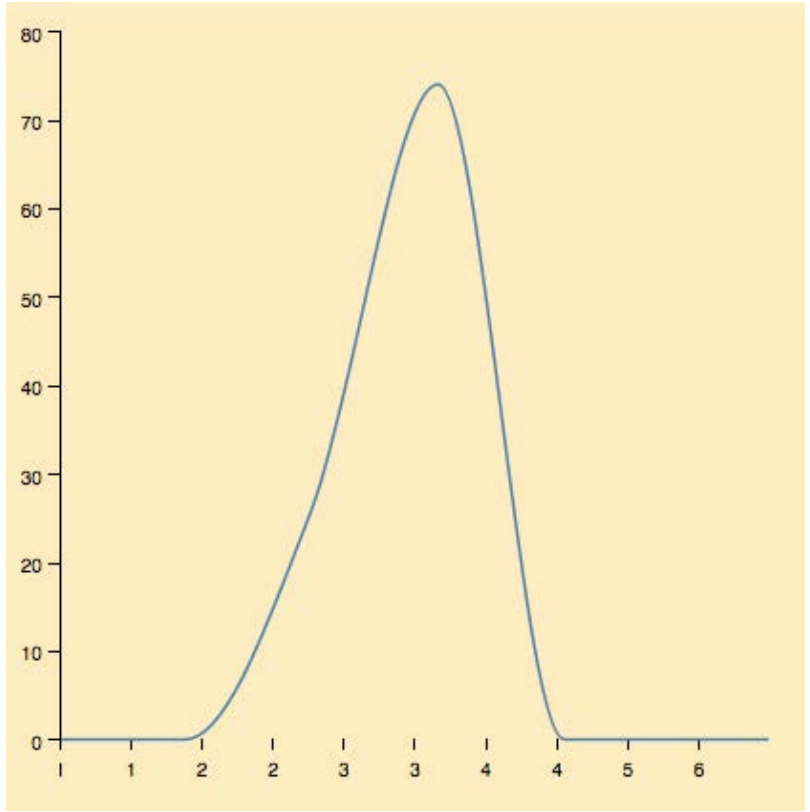




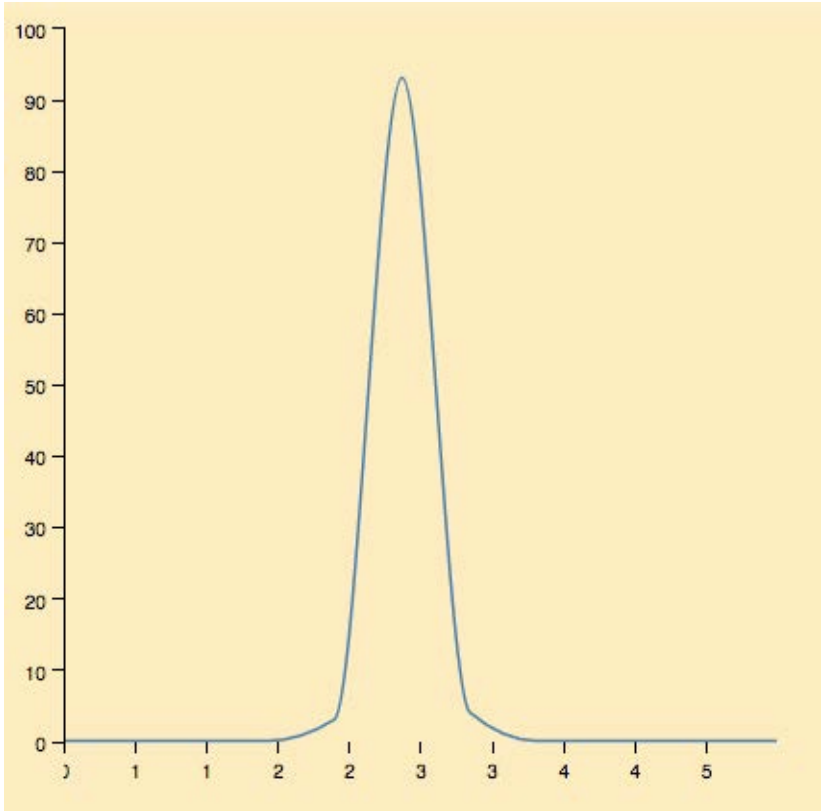

velocity: 15-30

Xebia

# Case: 'take up only 35 SP' policy




velocity: 20 - 45




velocity: 25 - 35

Xebia

# Summary

Xebia

# Summary

- Learning takes time for complex problems
- Speed up by running simulated experiments

- Benefits:
  - From months to seconds
  - Helps to understand the system
  - Helps to explore the fitness landscape
  - Helps to choose most promising change

- Beware of systematic errors and interpretation of results

Xebia