Scrum? Agile?

KTH, Sep 2018

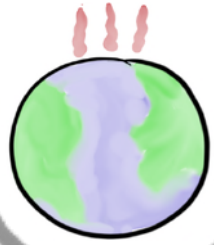
Henrik Kniberg

henrik.kniberg@crisp.se
@HenrikKniberg

Consultant

crisp
www.crisp.se

Climate guy



Dad

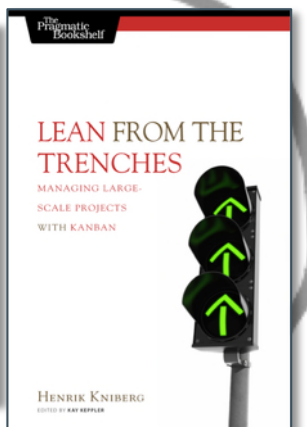
Organizational coach & Change Instigator

Spotify

LEGO

MOJANG

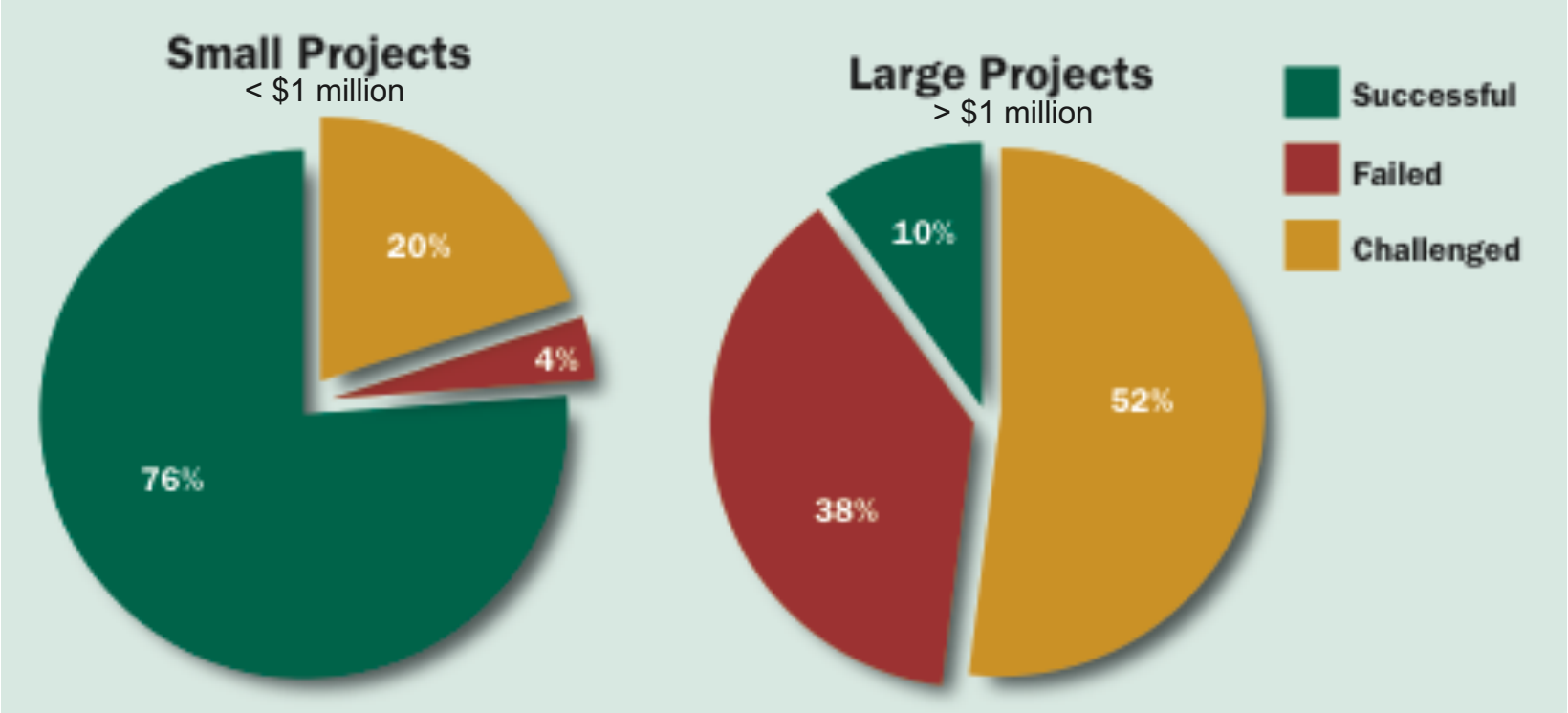
Author




Scrum
Retrospective
Continuous Integration
Kanban
Agile
Sprint
Lean
Velocity
User stories
TDD
XP
Pair programming
Daily standup
Henrik Kniberg

A history of failure

Big Projects usually fail. Regardless of process.



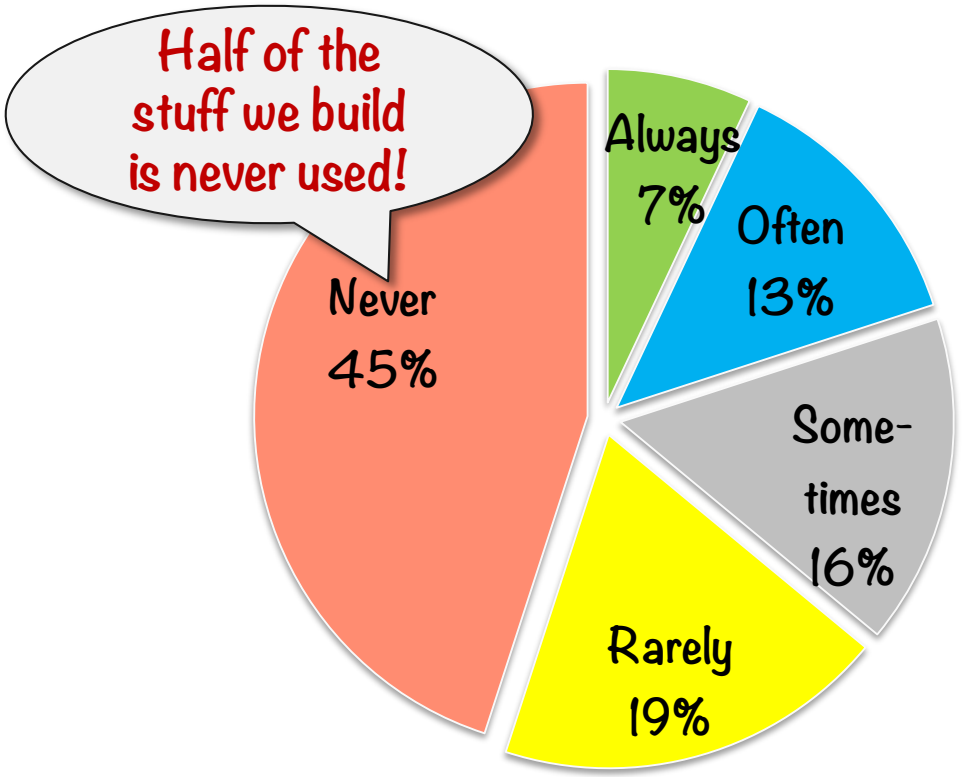
Source: Chaos Manifesto 2013

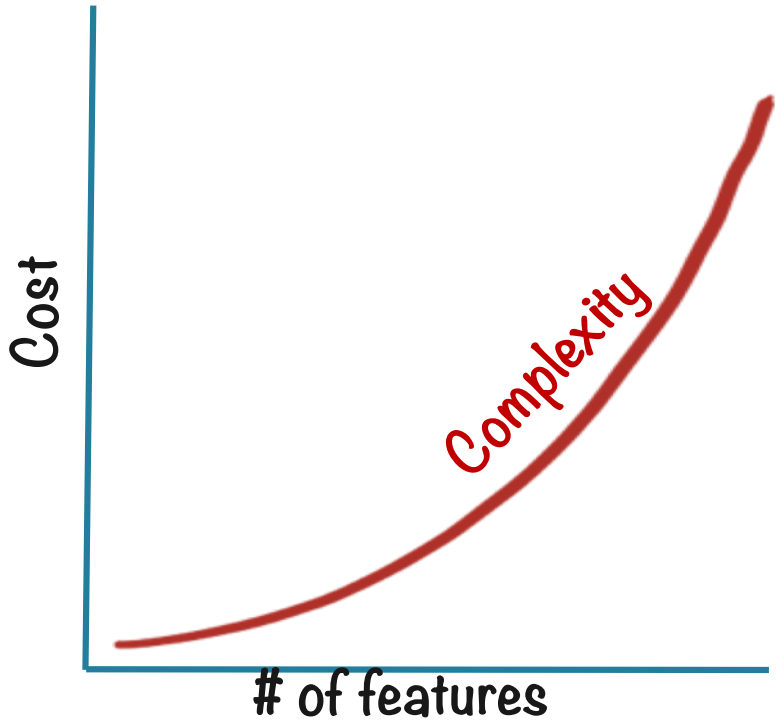
"half of all large IT projects (>$15 million) massively blow their budgets. On average, large IT projects run 45 percent over budget, while delivering 56 percent less value than predicted. Software projects run the highest risk of cost and schedule overruns"

http://www.mckinsey.com/business-functions/business-technology/our-insights/delivering-large-scale-it-projects-on-time-on-budget-and-on-value

We tend to build the wrong thing

Features and functions used in a typical system





Sources:
Standish group study reported at XP2002 by Jim Johnson, Chairman

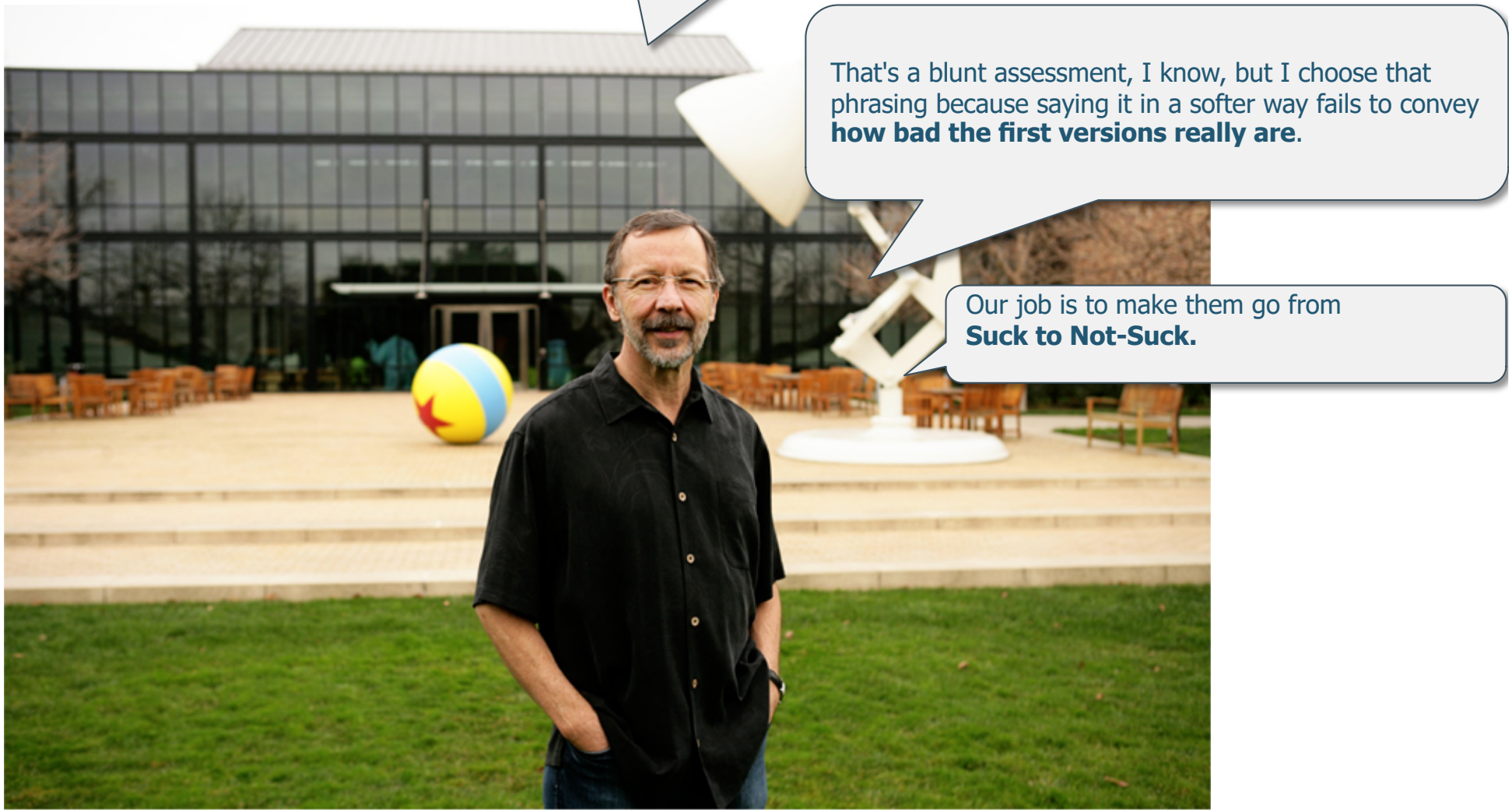
The right-hand graph is courtesy of Mary Poppendieck

How successful products are developed

Henrik Kniberg

Henrik Kniberg

Example: Pixar



Ed Catmull
President of Pixar & Disney Animation Studios

Henrik Kniberg

In the early stage of making a movie, we draw storyboards (a comic-book version of the story) and then edit them together with dialogue and temporary music.

The first versions are very rough, but they give a sense of what the problems are, which in the beginning of all productions are many.

We then iterate, and each version typically gets better and better.

Henrik Kniberg

Henrik Kniberg

Henrik Kniberg

Lego Universe Spider Cave

Brian Tyler

MINECRAFT

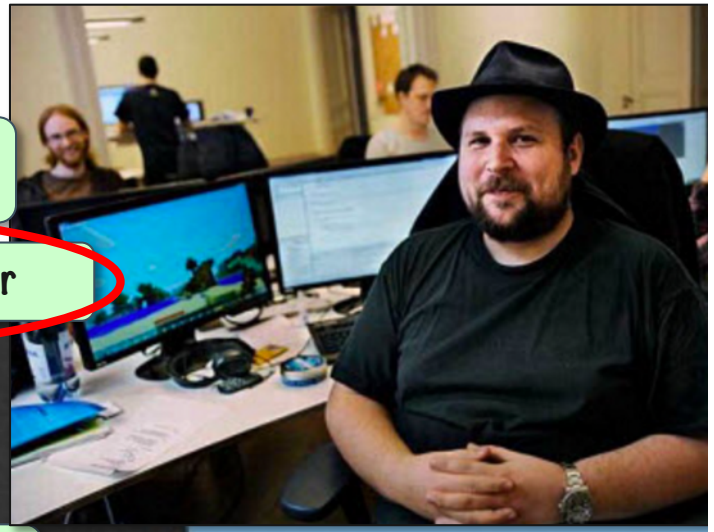
Built by 1-2 people

6 days to first public release

> 100 releases within first year

$80 million revenue within first 15 months

Sold to MS for $2.5 Billion!



Henrik Kniberg

4 years of development – 1000 man years!

Super Beautiful!
Kinda fun.
Low revenue.

2 years later...

Dead!

Few days of development

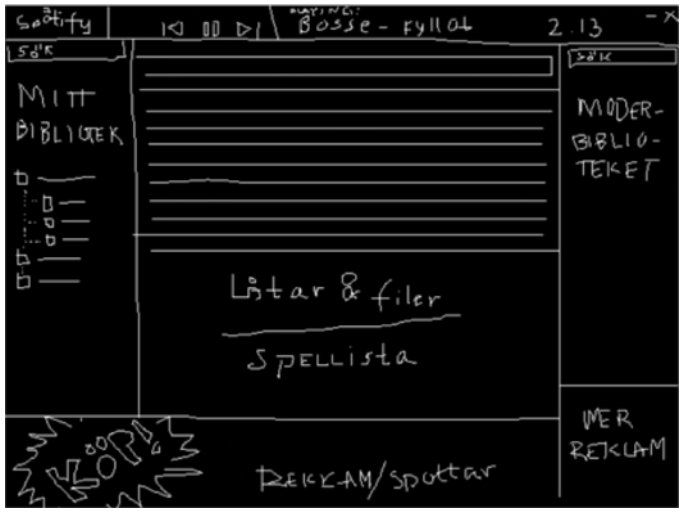
Ugly, kinda fun.

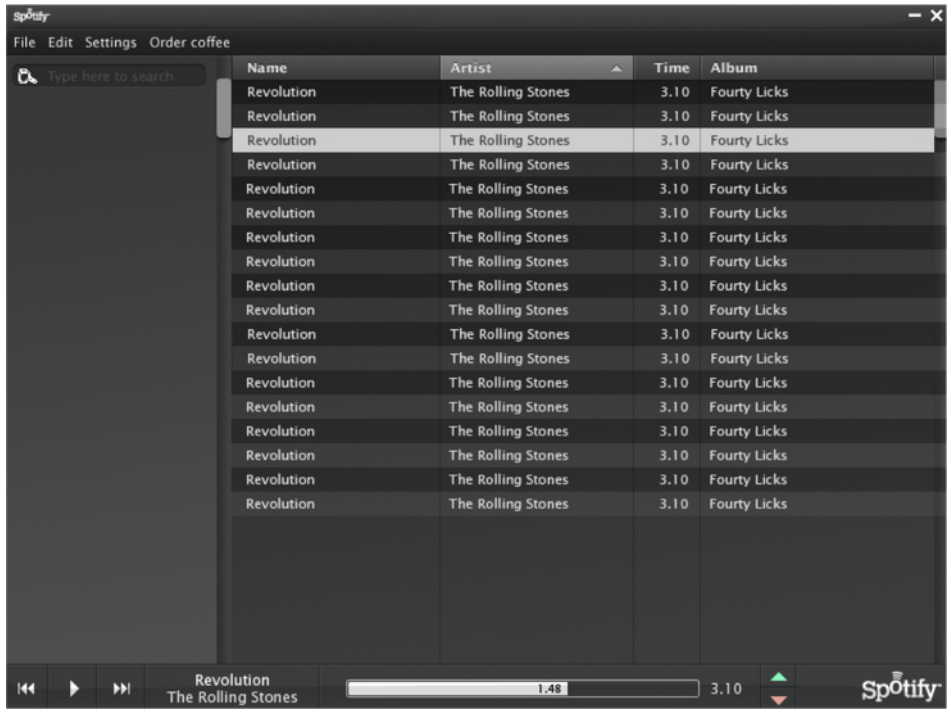
100s of releases....

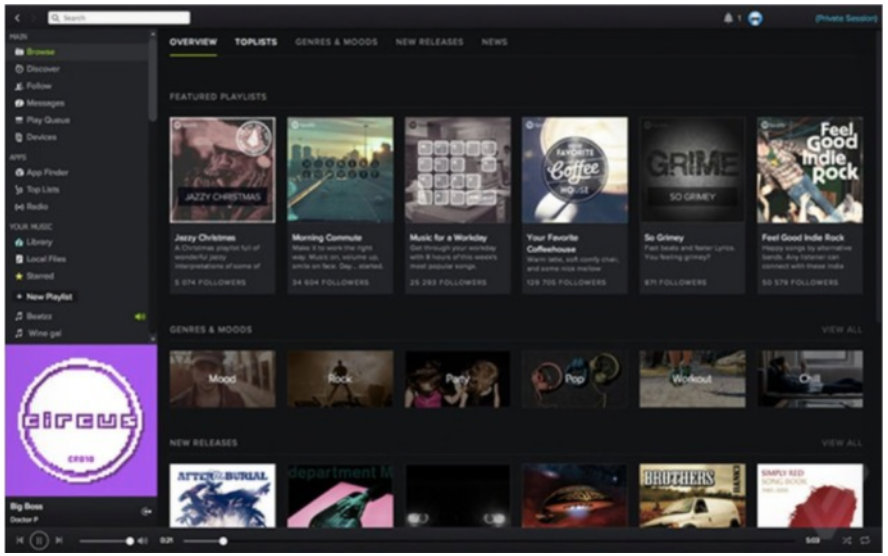
Beautiful enough.
SUPER fun!
LOTS of revenue!

Fame & Glory &
Riches & Happy players!

Henrik Kniberg







Henrik Kniberg

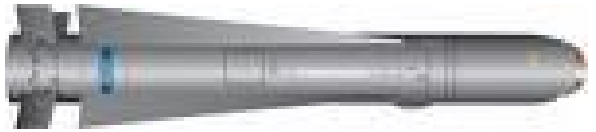
Predictive vs Adaptive processes

Predictive process = cannon ball

Assumptions:

- The customers knows what they need
- The teams know how to deliver it
- Few things change along the way

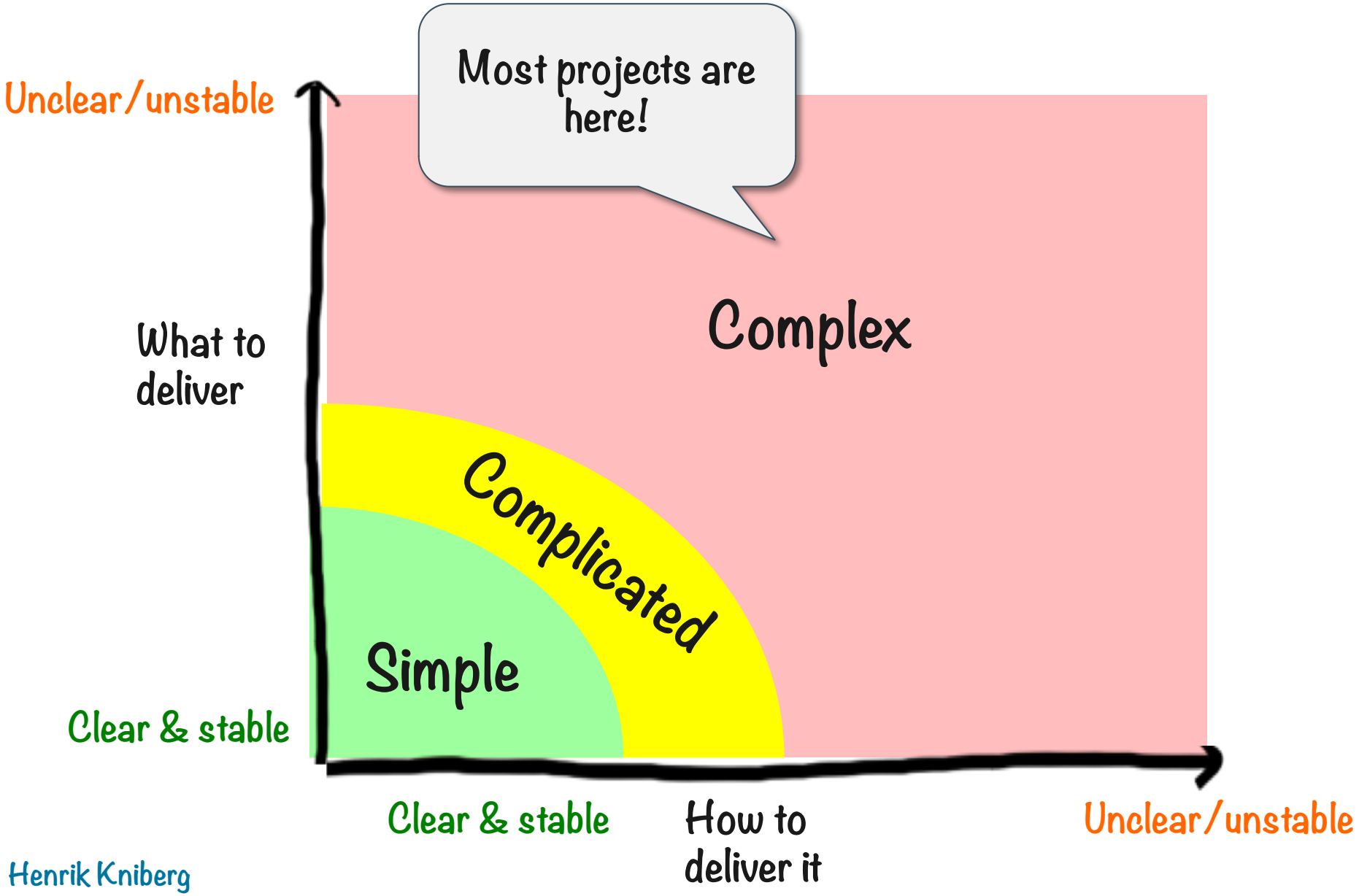
Adaptive process = homing missile



Assumptions:

- The customer discovers what they need
- The teams discover how to deliver it
- Many things change along the way

Henrik Kniberg


Most projects are here!
Unclear/unstable
What to deliver
Complex
Complicated
Simple
Clear & stable
Clear & stable
How to deliver it
Unclear/unstable
Henrik Kniberg

Once upon a time
15 years ago...

www.agilemanifesto.org

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools

Working solutions over comprehensive documentation

Customer collaboration over contract negotiation

Responding to feedback over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.

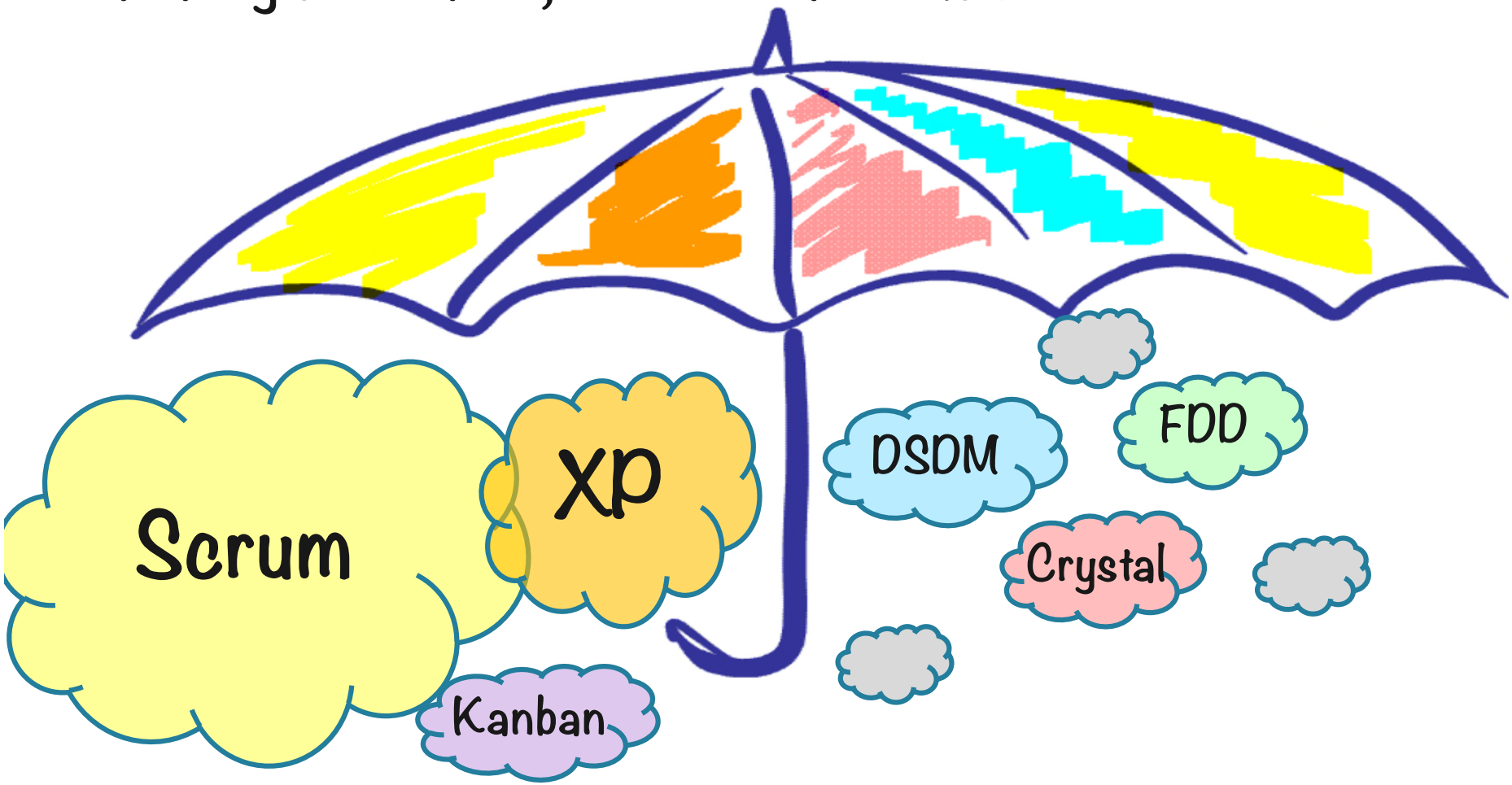
Henrik Kniberg

Principles behind the Agile Manifesto

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity--the art of maximizing the amount of work not done--is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

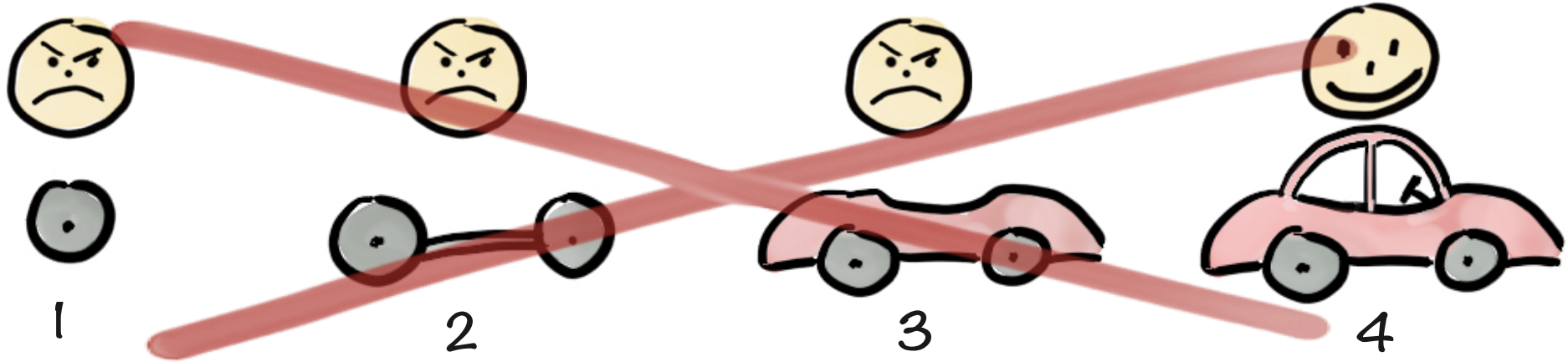
Agile "umbrella" –
a family of iterative, incremental methods



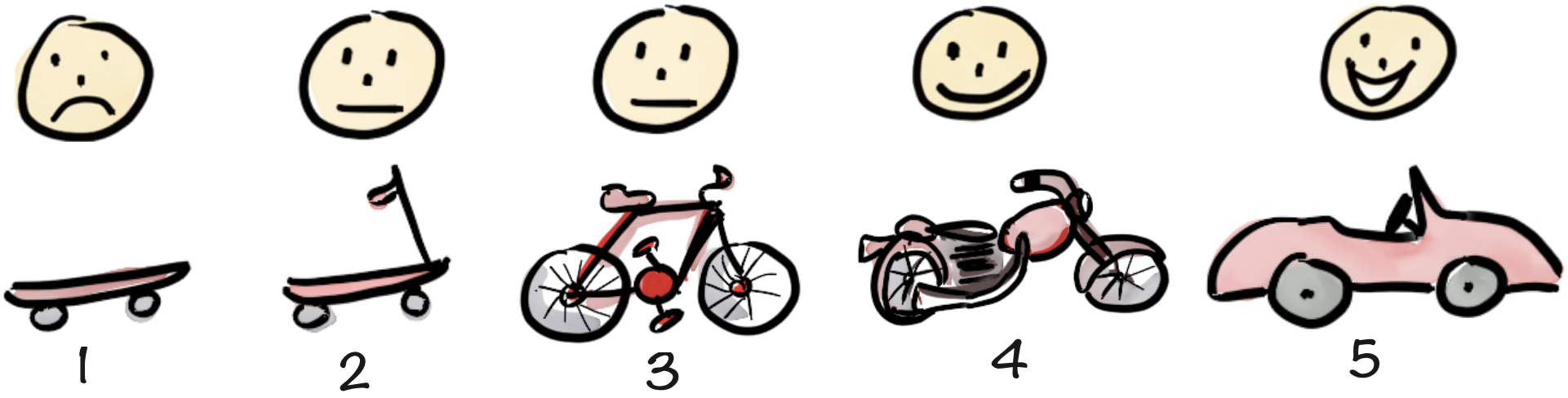
Henrik Kniberg

Slicing the elephant

Not like this....



Like this!



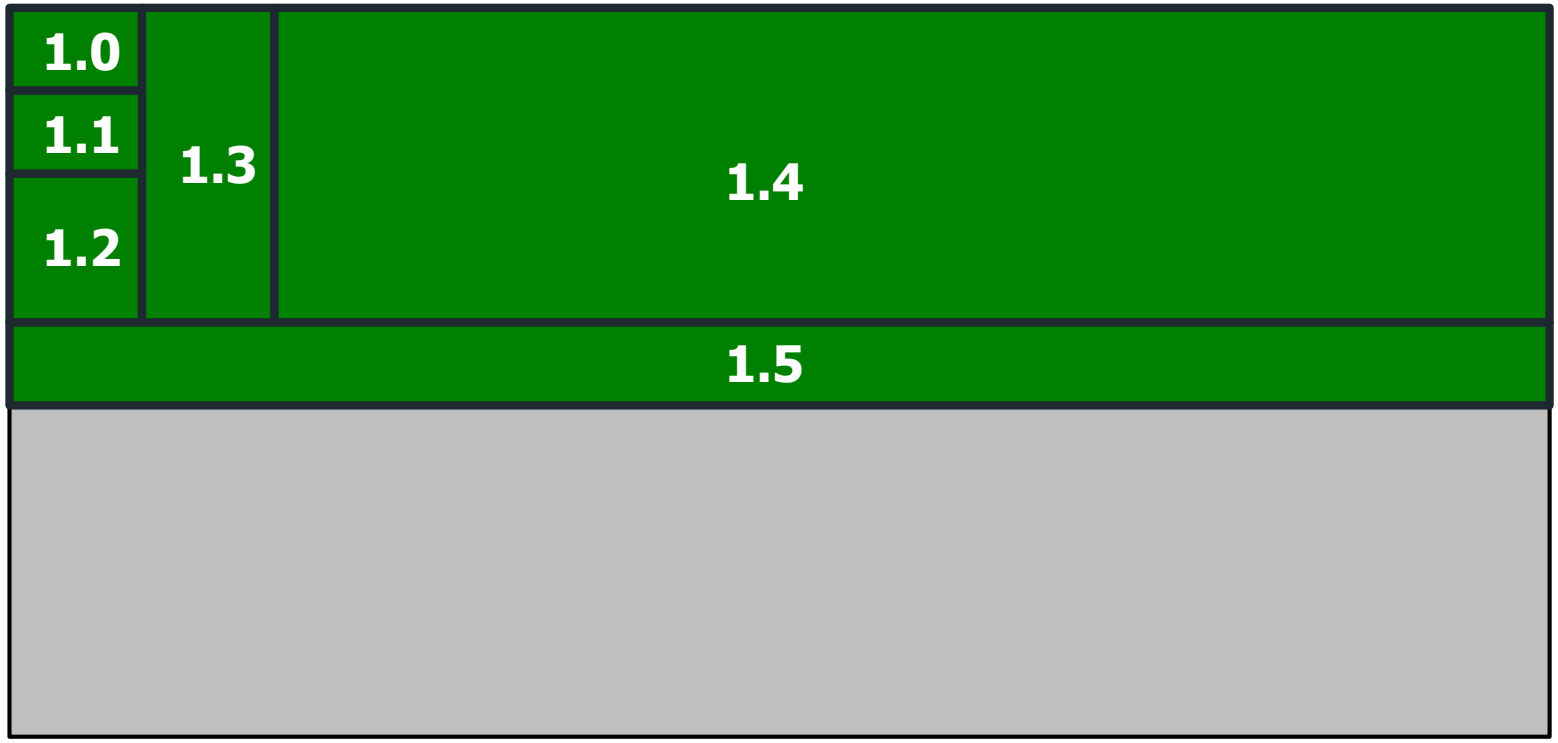
Henrik Kniberg

Slice the elephant!

Region Östergötland, Uppsala, etc

Crime types (weapon, drunk driving, shoplifting, etc)

Integrations



Optimize for flow, not resource utilization

Henrik Kniberg

Needed: 5 volunteers

100% resource utilization = 0% flow

High resource utilization

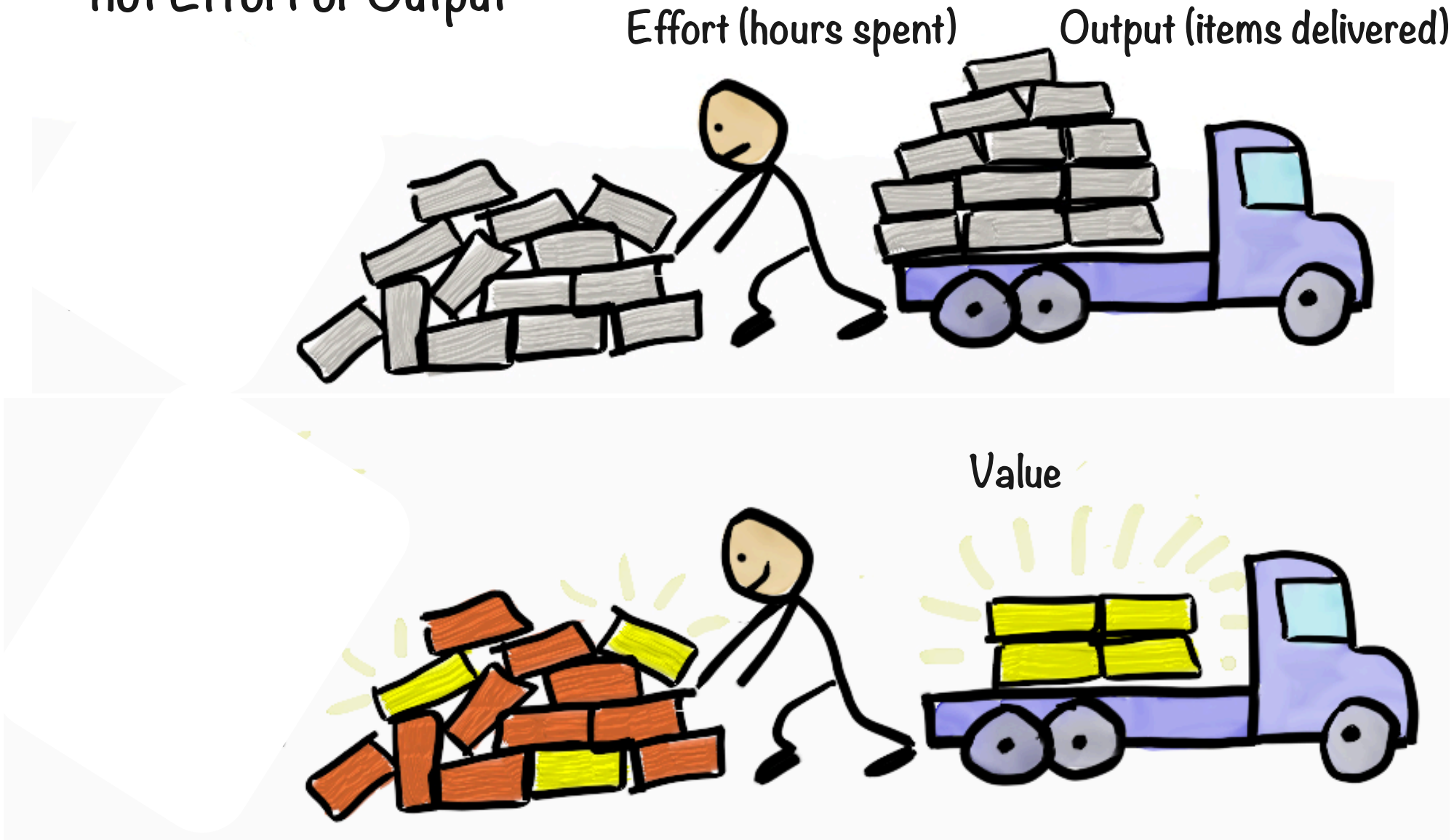
Fast flow

Optimize for value, not effort

Henrik Kniberg

Focus on Value

not Effort or Output



Henrik Kniberg

What you measure is what you get

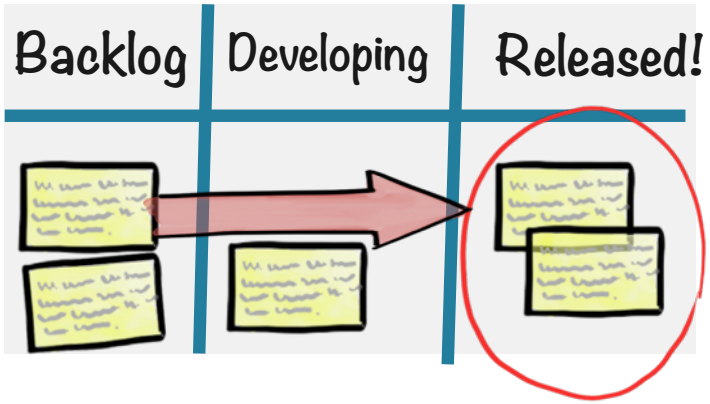
Focusing on
Effort

Hours

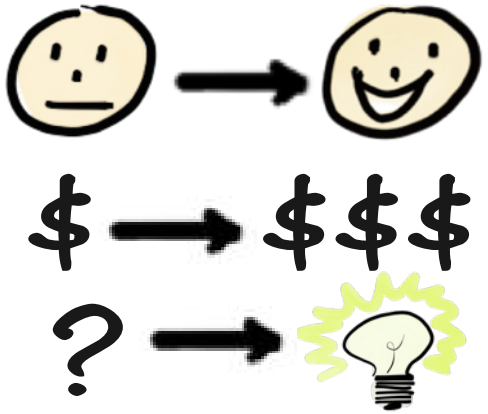
Time reports

Resource utilization

Focusing on
Output



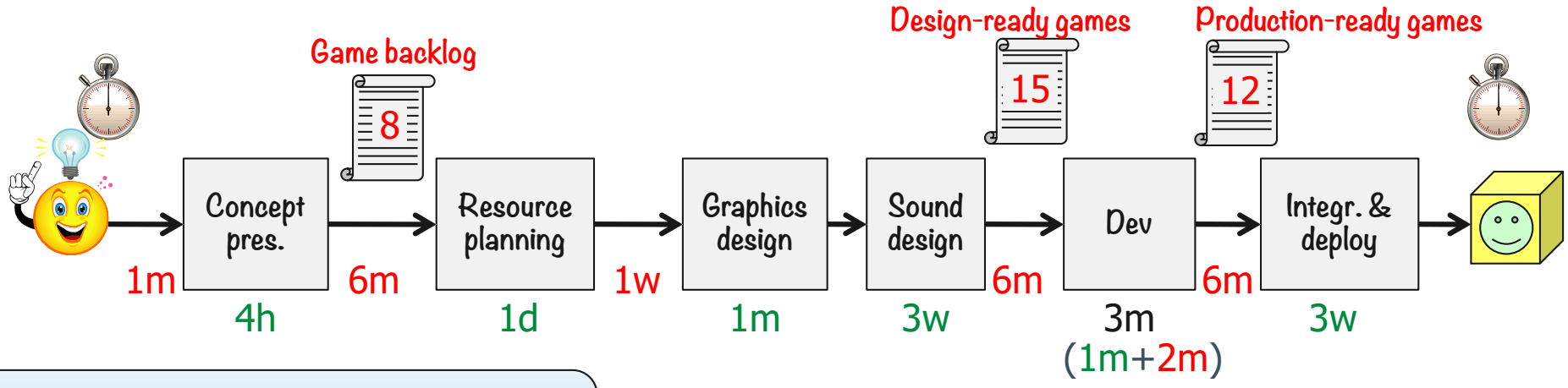
Focusing on
Value & Learning



Henrik Kniberg

Stable, cross-functional teams

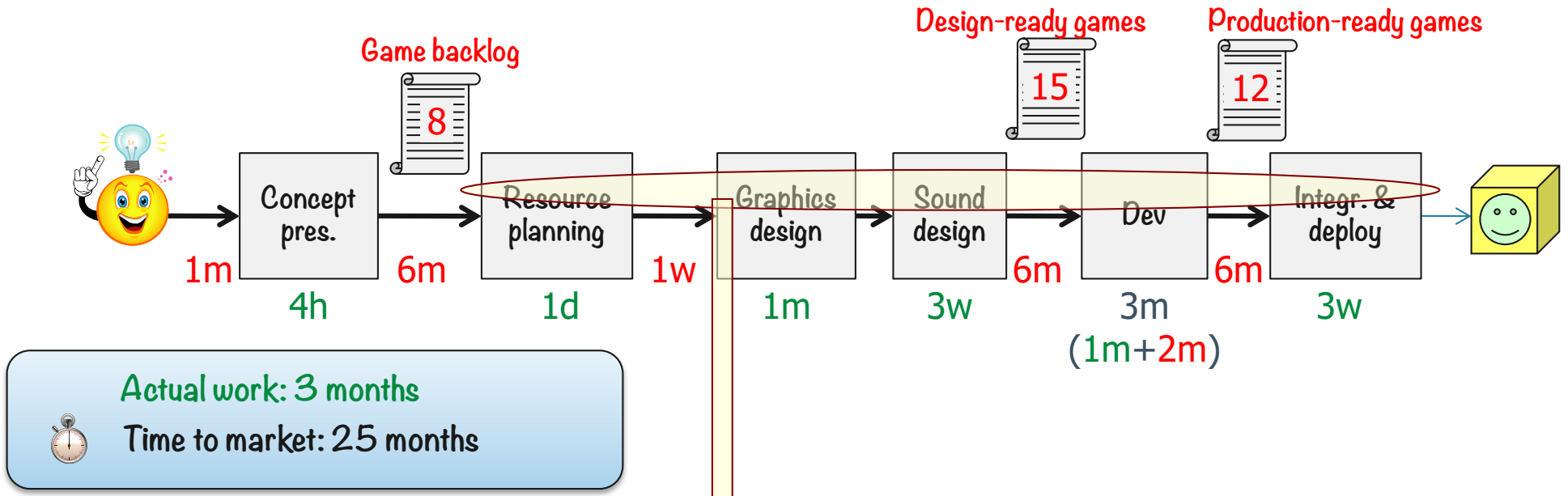
Case study: Game development company



Actual work: 3 months

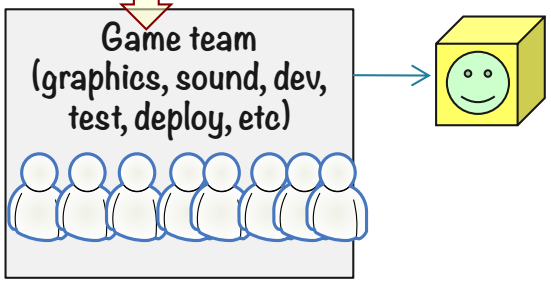
Time to market: 25 months

Before



After

Cross-functional game teams



Time to market: 3-4 months

7 times faster!

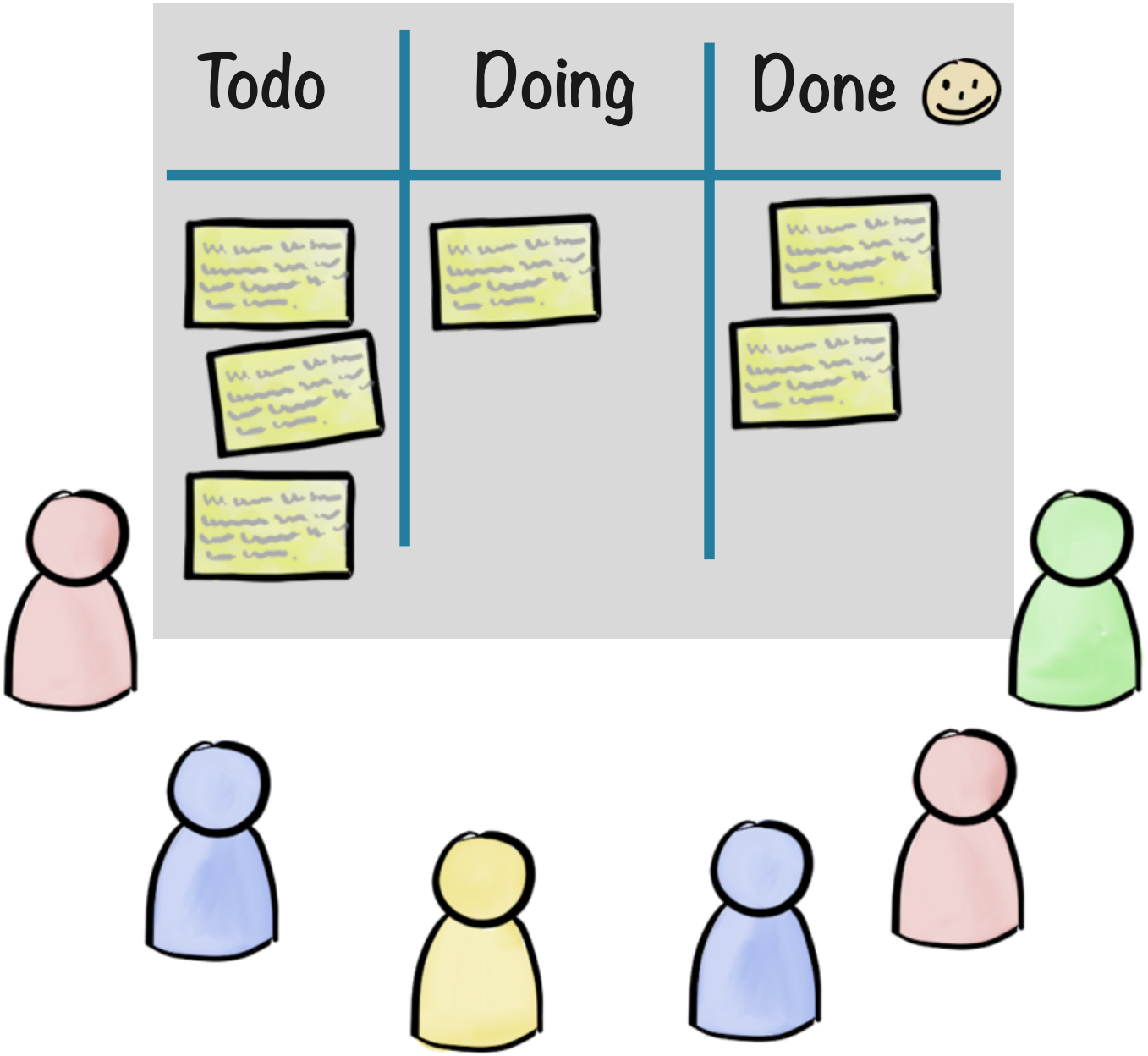
Better games!

More fun!

Agile team =

stable, small, cross-functional, self-organizing, co-located

Henrik Kniberg

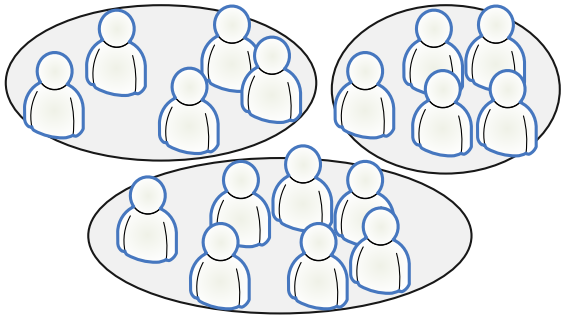


Henrik Kniberg

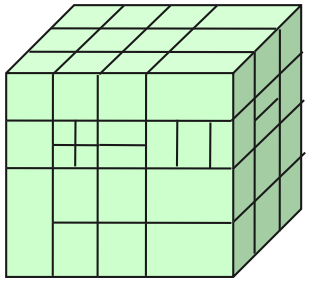
Scrum = the most popular agile framework

Scrum in a nutshell

Split your organization



Split your product

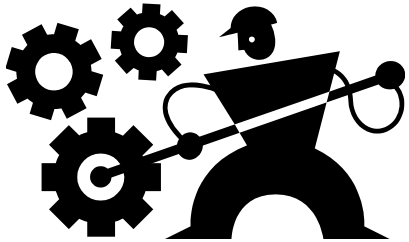


~~Large group~~ spending ~~a long time~~ building a ~~huge thing~~

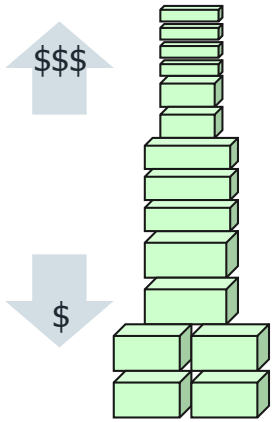
Small team spending a little time building a small thing

... but integrating regularly to see the whole

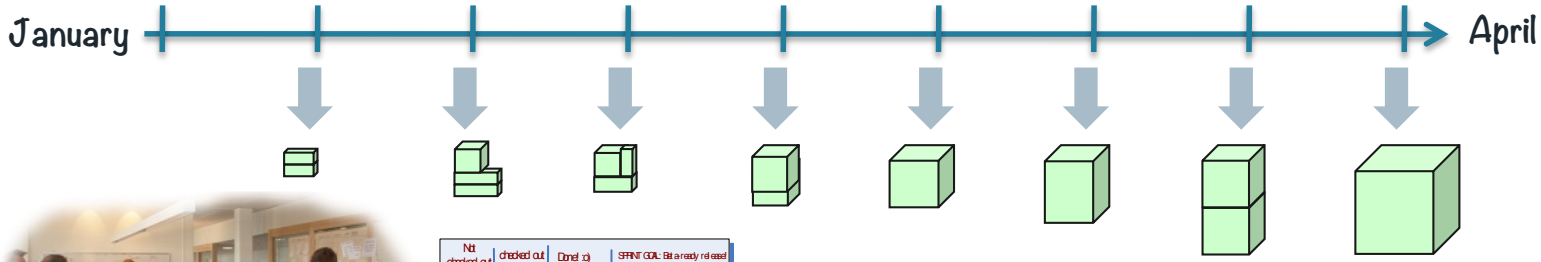
Optimize process

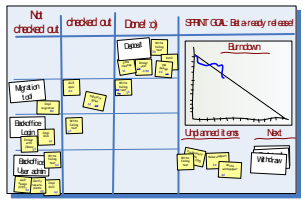


Order the backlog



Split time



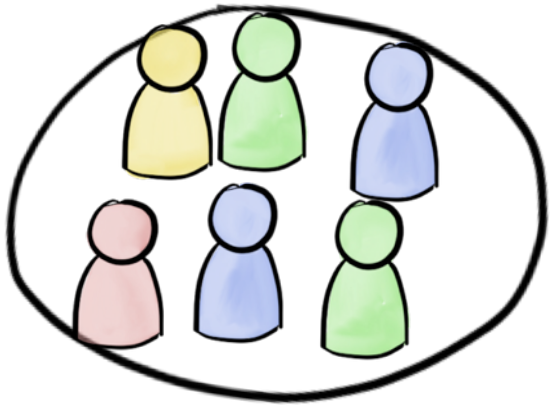


Henrik Kniberg

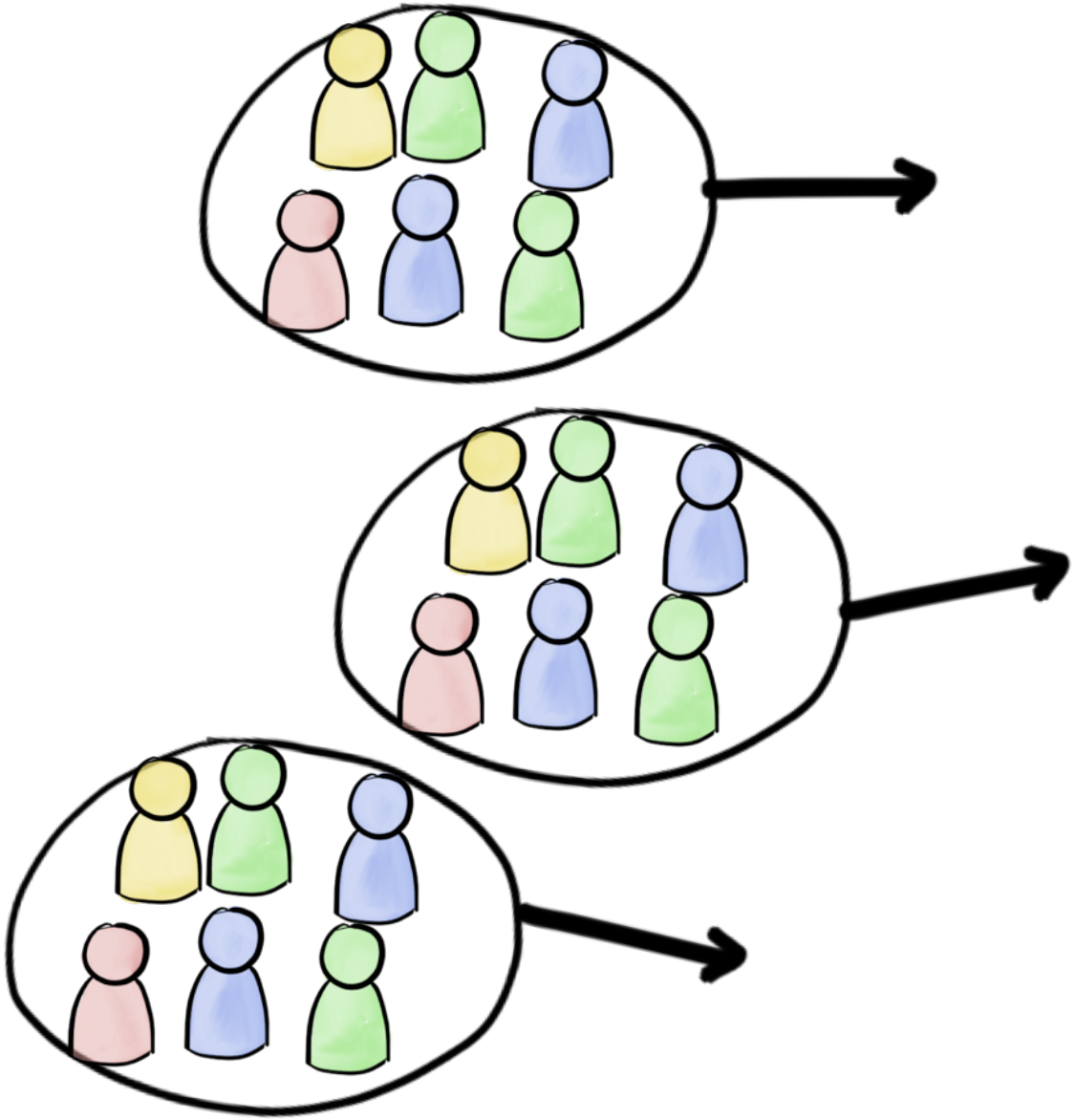
Leadership

Henrik Kniberg

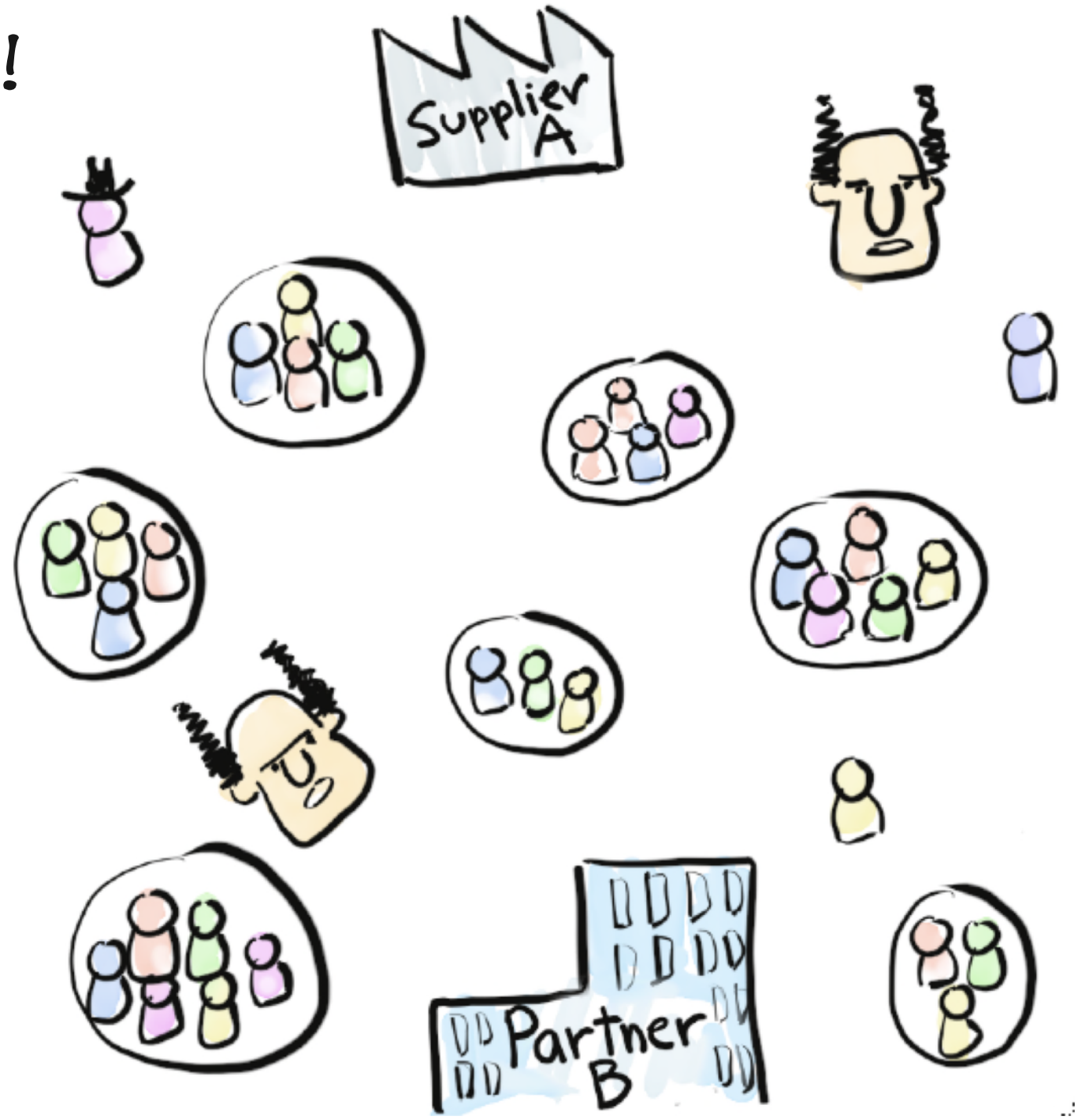
Not too hard



A bit trickier

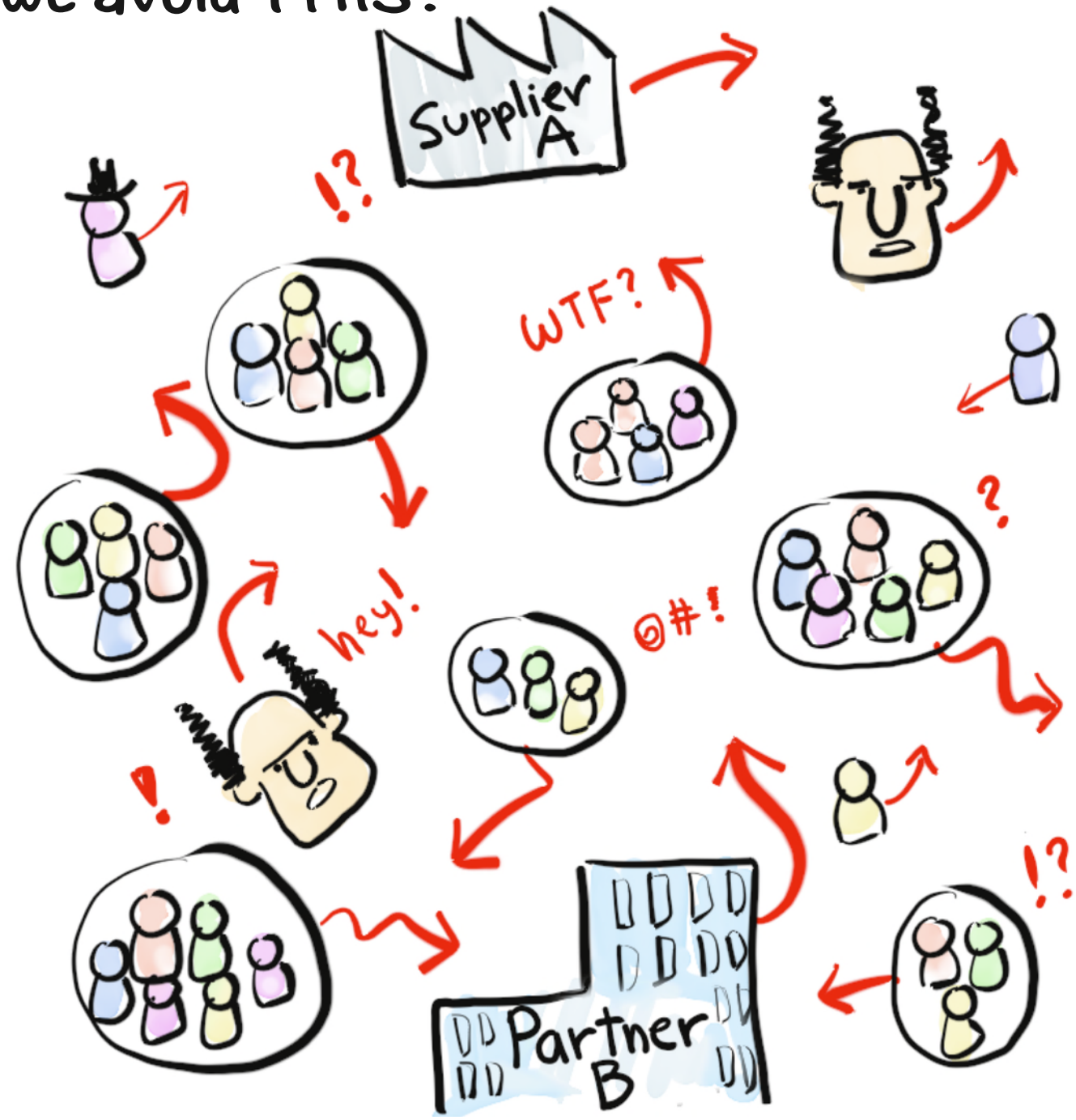


Hard!



Henrik Kniberg

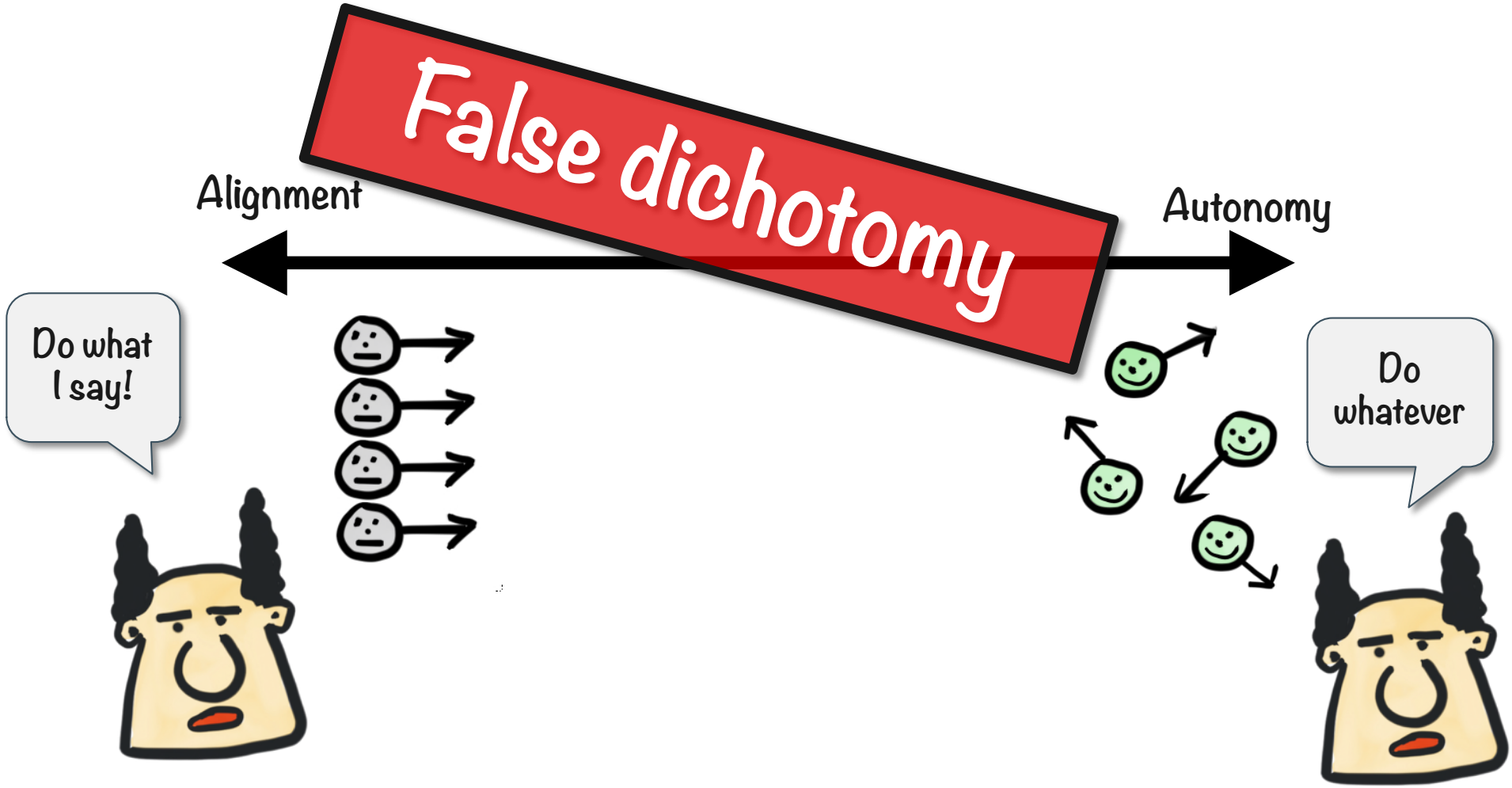
How do we avoid THIS?



Henrik Kniberg

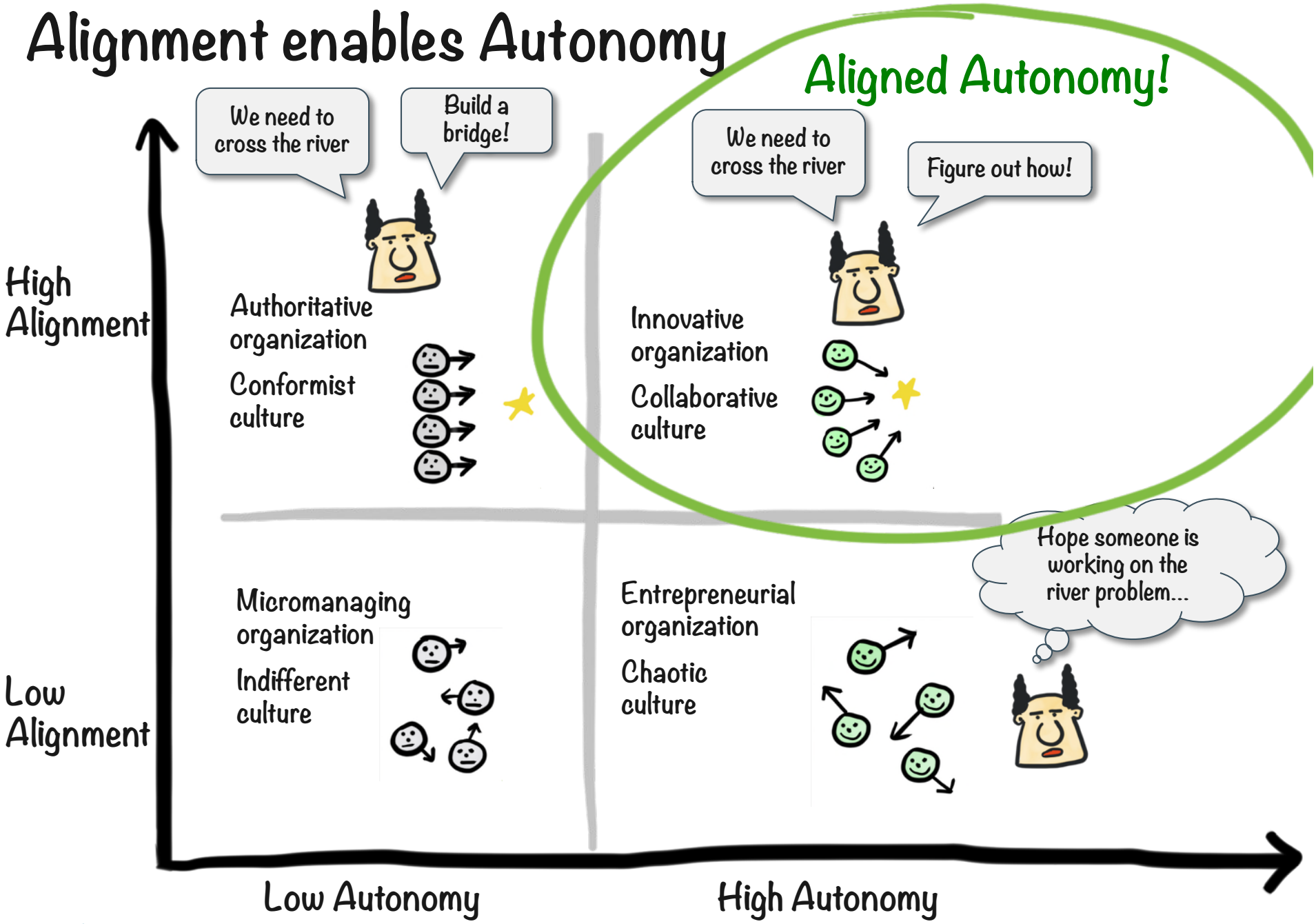
Common reaction

Alignment & Autonomy



Henrik Kniberg

Alignment enables Autonomy



Henrik Kniberg

Leader's job:
Explain what problem needs to be solved.
And Why.

Henrik Kniberg

Agile outside IT

Henrik Kniberg

JAS 39E Saab Gripen

Agile practices implemented at every level and in every discipline: software, hardware and fuselage design.

World's most cost-effective military aircraft
($4700 Cost per Flight Hour)

Compared to F35 joint strike fighter, Gripen 39E has:

- 50x lower development cost!
- 10x lower unit cost!

Pilots on the same site as development teams. Direct feedback provided every sprint.

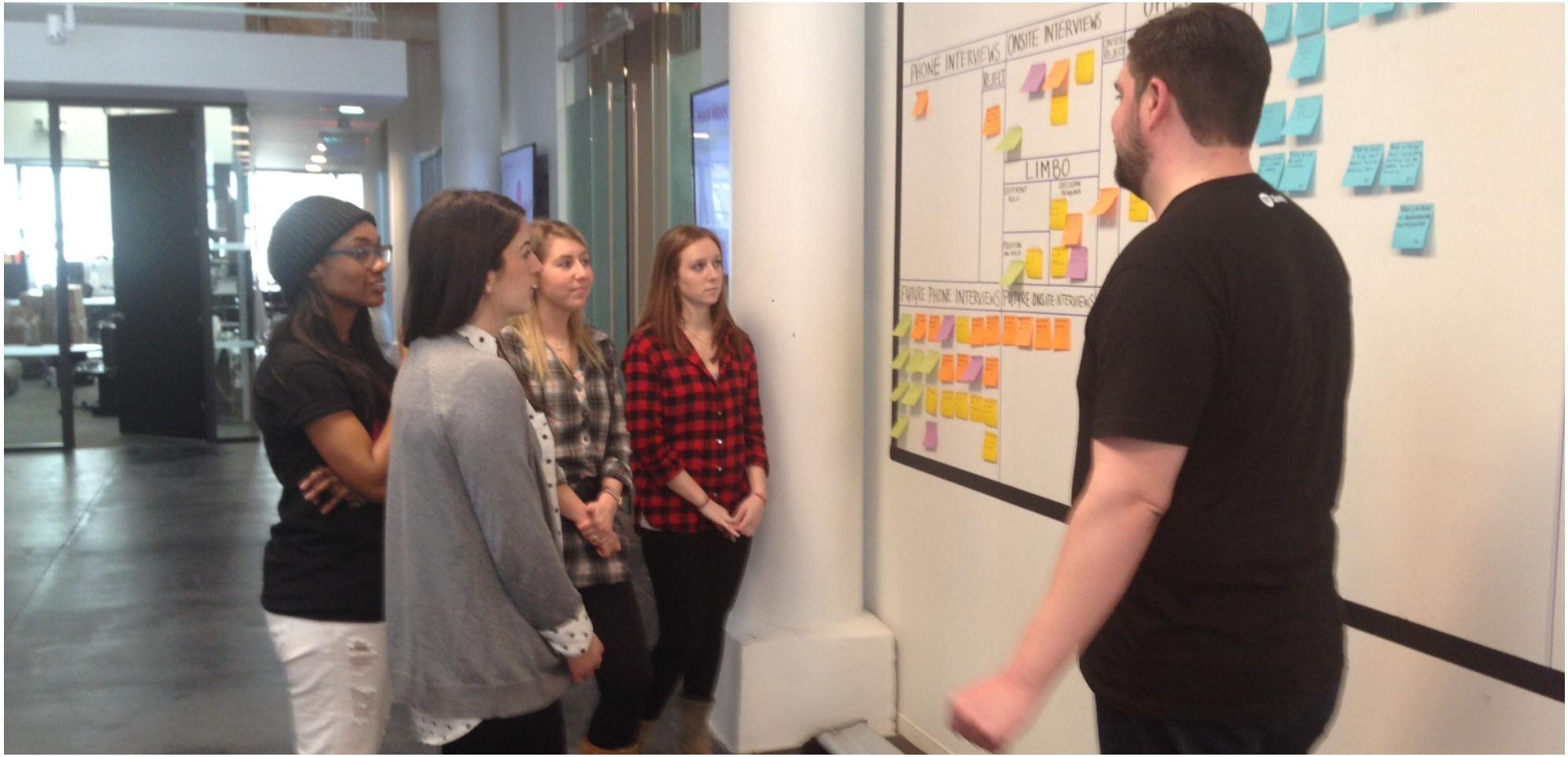
1500 people, all co-located in Linköping, Sweden.

Sources:

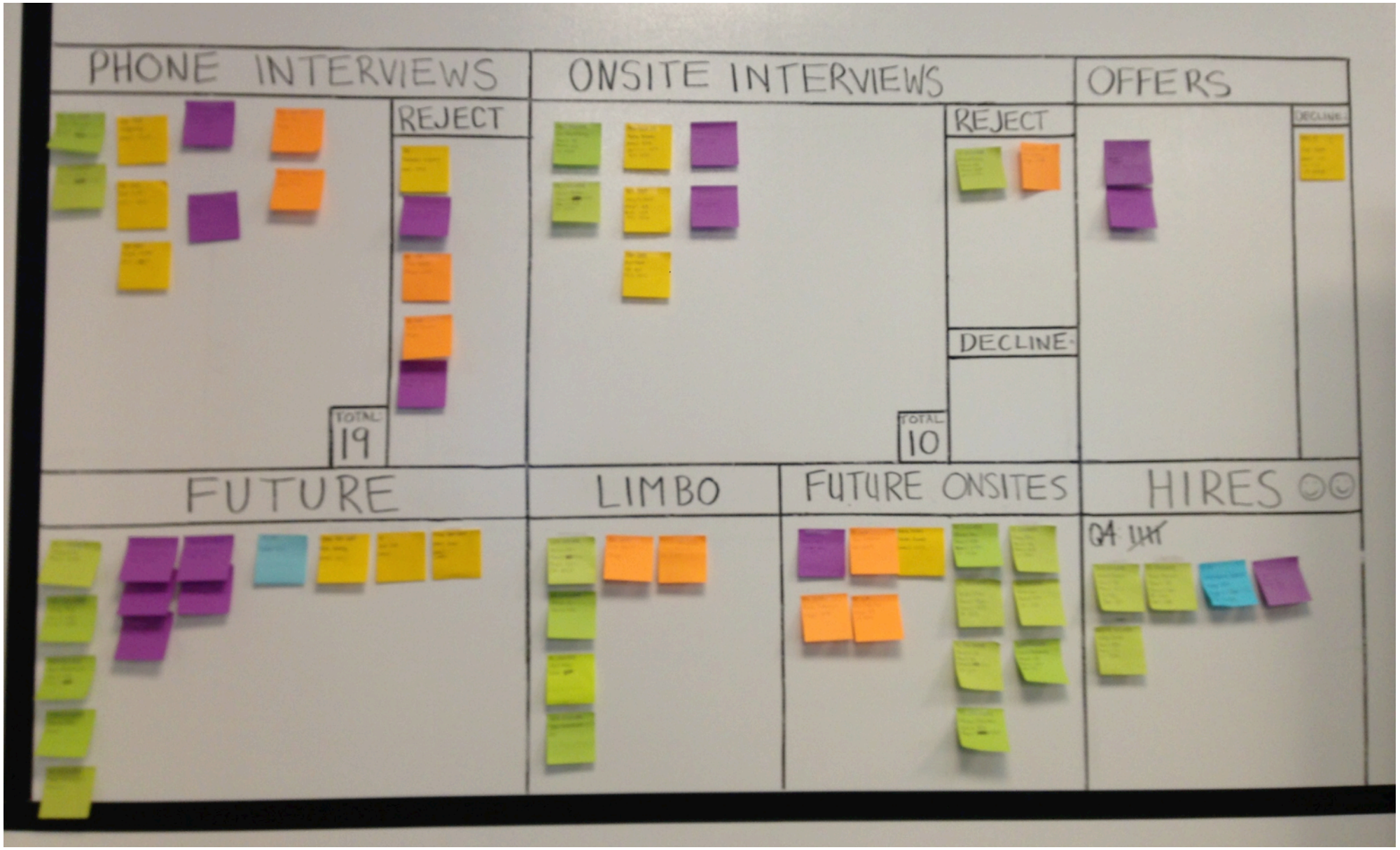
- http://www.stratpost.com/gripen-operational-cost-lowest-of-all-western-fighters-janes
- Personal visit to SAAB Linköping
- Research paper "Owning the Sky with Agile"

Henrik Kniberg

Recruitment team



Recruitment team

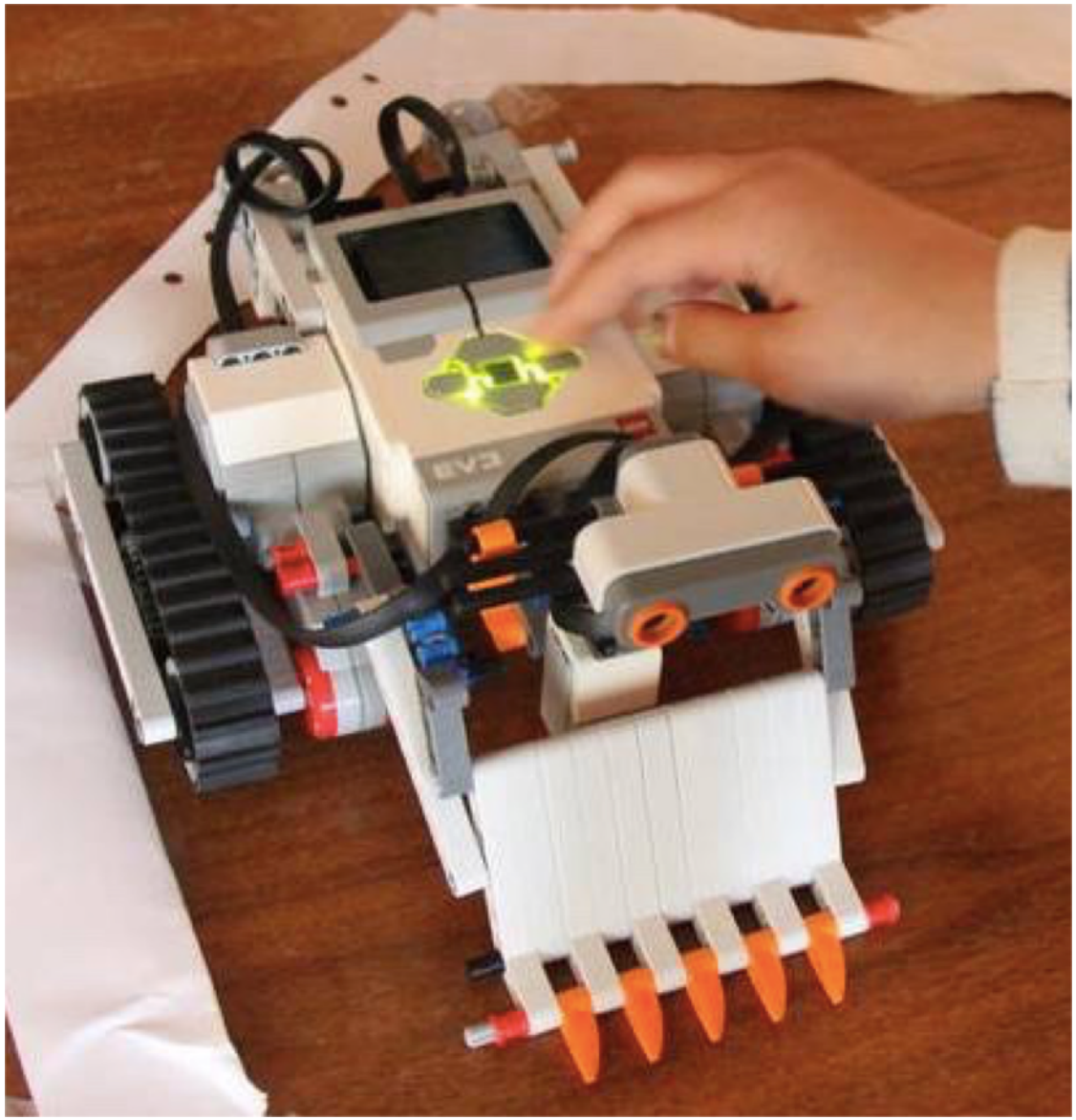



BBQ Board
To do
Going on
Done!
Drink table
Dip snacks
Light the BBQ
Orange Juice
Veggie sticks
Sallad
Food serving table
BBQ
Ice cream
Grapes + cheese + crackers

The story of Robbit

Henrik Kniberg

Robit



Henrik Kniberg

Henrik Kniberg

COPENHAGEN
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2015

goto; conference

Follow us @gotocph

Oct 7-8, 2015

Program Schedule Contact

2 kids & rookies with very little robot experience...

... vs ten teams of adult geeks and programmers

LEGO® MINDSTORMS® COMPETITION

Do you have what it takes to fight and win the competition of the future? - Then sign up and enter the LEGO® MINDSTORMS® Robotic competition at GOTO Copenhagen 2015 and win fabulous prizes on top of the fame and glory!

How to enter the game?

1. Form a team of 2-5 members (NB: Only conference attendees can join the competition)
2. Build your own intelligent, autonomous robot before the conference (use your own LEGO® MINDSTORMS® Robotic Toolkit or borrow one for free when registering to the competition)
3. Pitch it against the robots from other teams at the GOTO Conference Dinner, Monday October 5, 19:30-22:30
4. There will be prizes for the winning team

GOTO Cph 2016

GOTO Copenhagen 2016 will take place in Bella Center. Mark the days already: **October 3-6, 2016**

Said about GOTO

We have collected quotes from blogposts and articles etc. about GOTO Copenhagen 2015 on a single page

GOTO Community

Join the worldwide GOTO Community:

Platinum sponsor

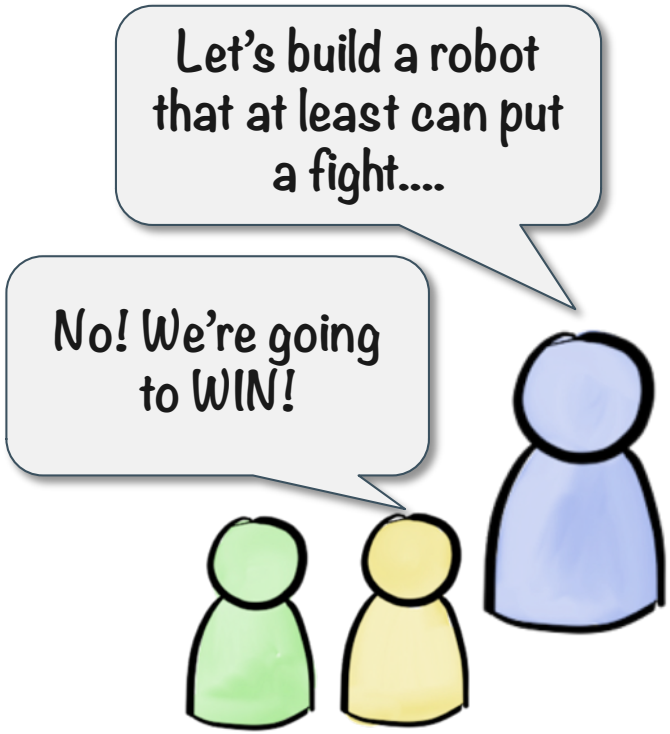
Nordea

I ♥ GOTO

"GOTO is definitely the best place to get a feeling for the newest trends. If there was just one conference I would attend to keep up with what is

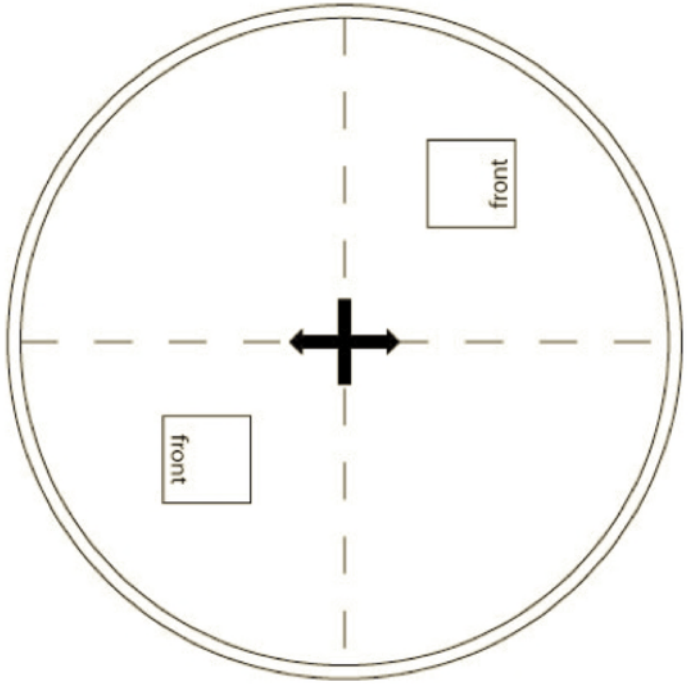
Henri

Step 1: Set a clear goal (define "success")



The Rules

1. The two sumo robots are placed as shown in the picture below with the front pointing away from each other.
2. On the judge signal the sumo robot's program is started. The robot have to wait 3 seconds before it starts being active.
3. A match lasts at most 2 minutes.
4. A sumo robot wins, if the other sumo robot is knocked over or pushed outside the ring. A sumo robot is outside the ring, if it touches the surface that supports the ring. If a sumo robot drives outside the ring by itself the sumo robot has lost.
5. If none of the sumo robots have left the ring or has been knocked over within the 2 minutes the match ends with a tie. If both sumo robots leaves the ring at the same time the match also ends with a tie.
6. The winner of a match receives 2 points, while both teams receives 1 point if the match ends in a tie, and the loser of a match receives 0 points.
7. A sumo tournament can be run with groups, sessions, semifinals, multiple rounds per match, etc, depending on the number of teams participating.



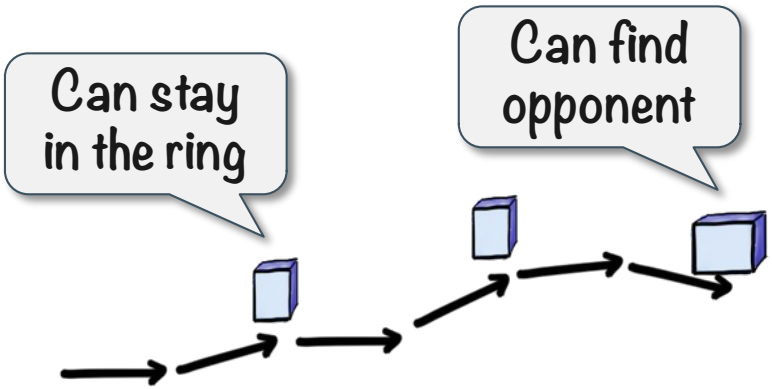
Henrik Kniberg

Henrik Kniberg

Step 2: Build a Minimum Viable Robot (Earliest Testable Robot)

Henrik Kniberg

Aim for the clouds,
but deliver and test in small steps

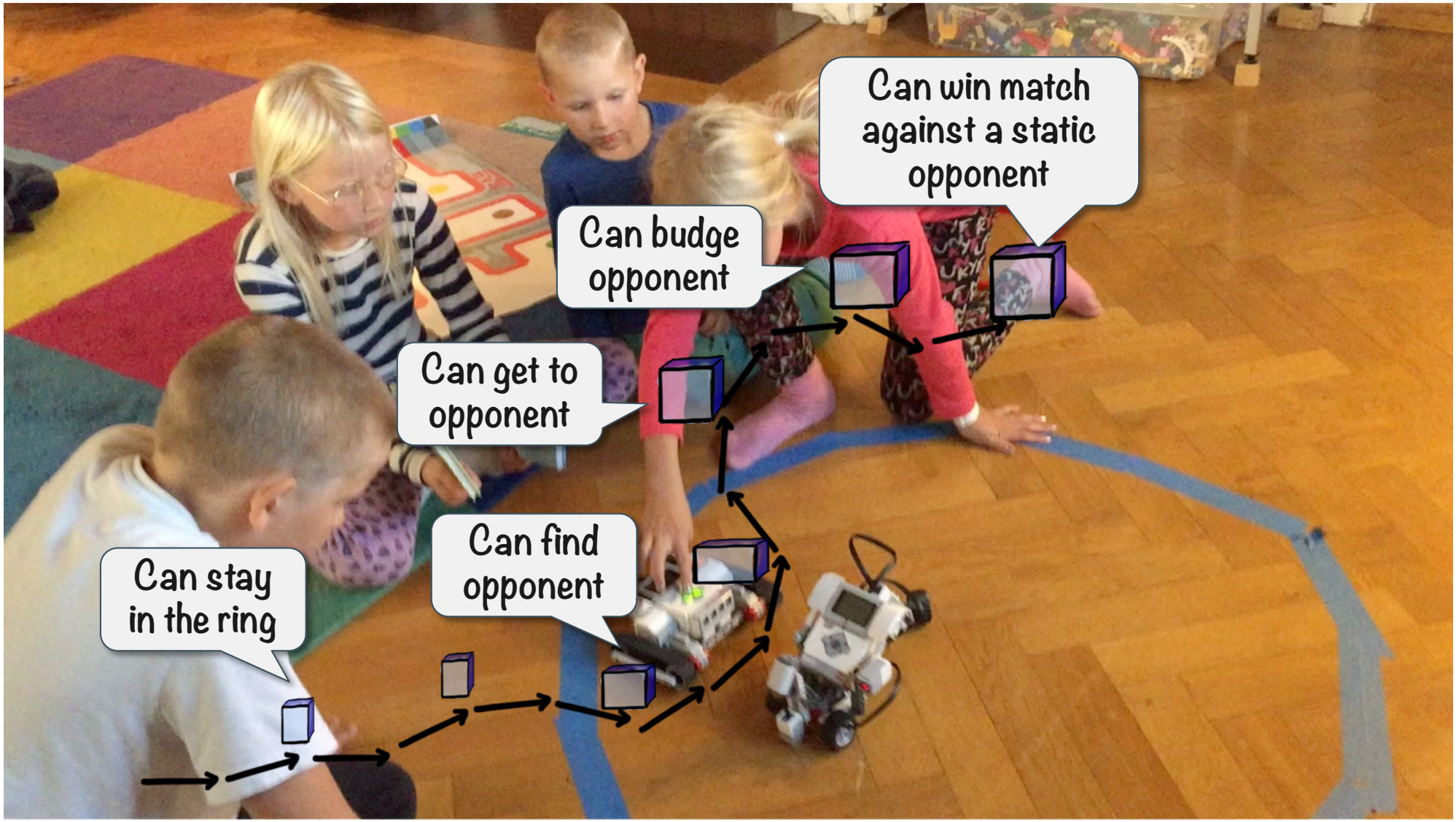


Henrik Kniberg

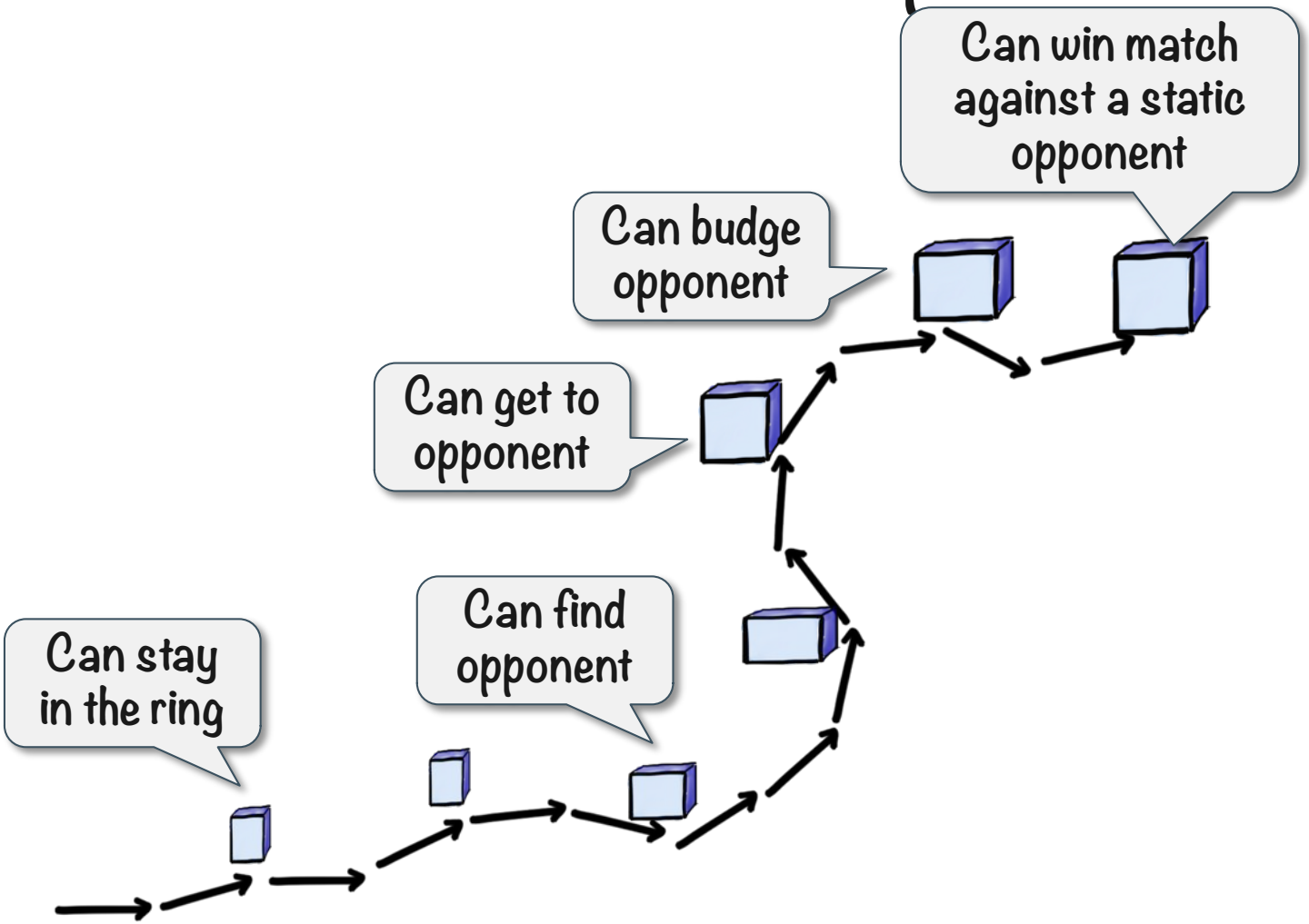
Step 3: Build an opponent to practice against

Henrik Kniberg

Field test, Field test, Field test



Aim for the clouds, but deliver and test in small steps



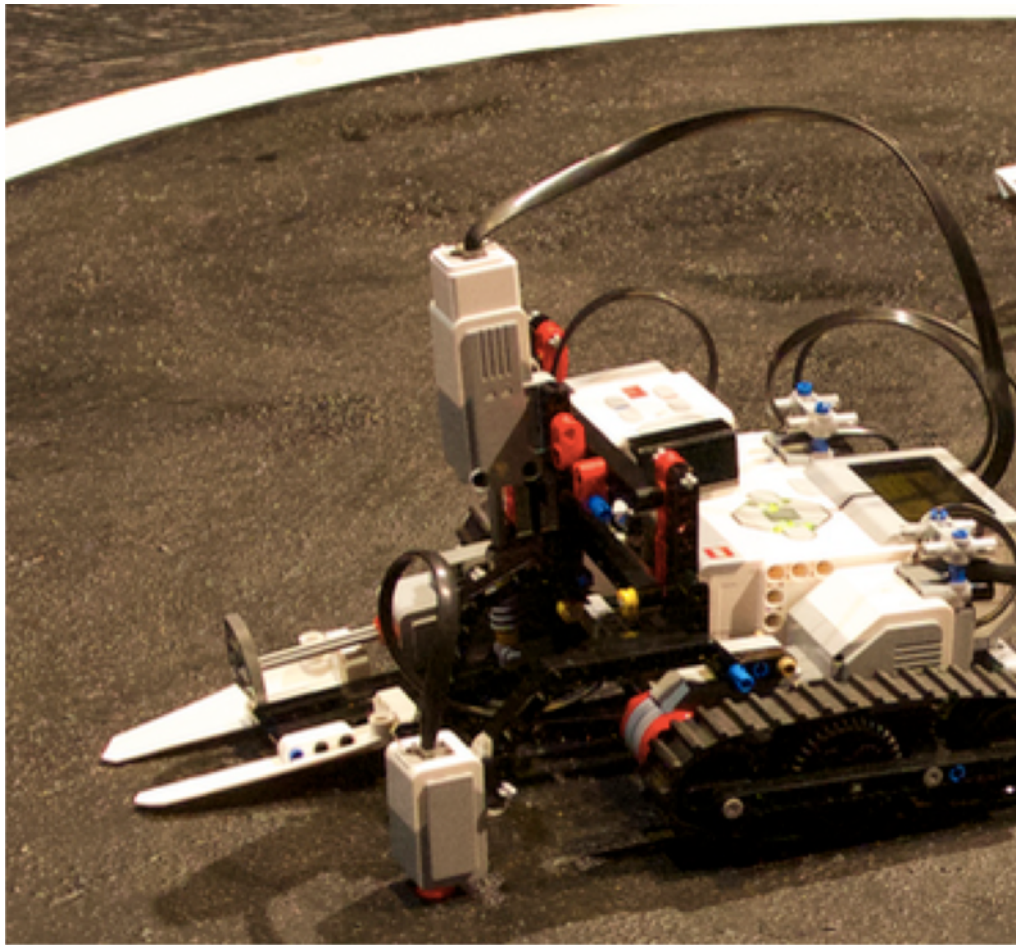
Henrik Kniberg

Lifter? Or no lifter?

Hypothesis:
- Mechanical Lifter can help us win

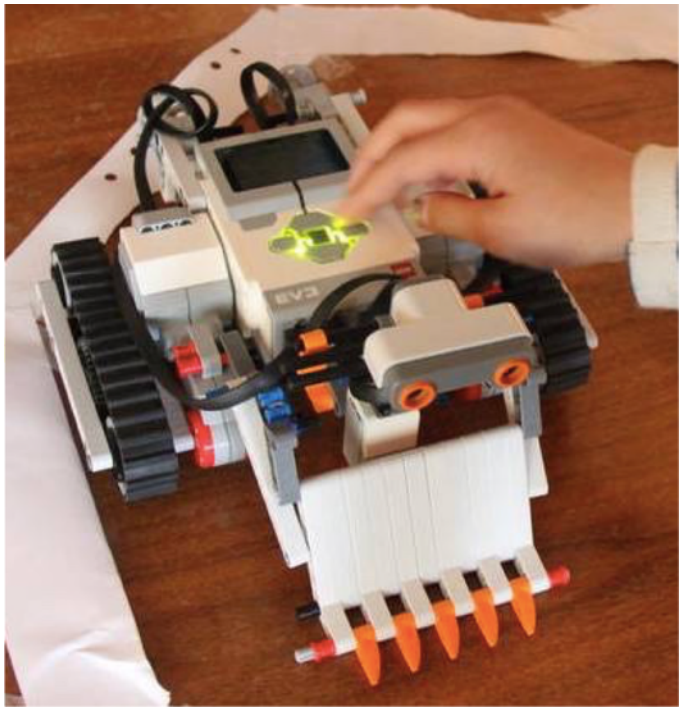
Experiment:
- Build a simple lifter and try

Learning:
- Works as designed...
- But too weak to lift opponent
- ... so it doesn't help us win!

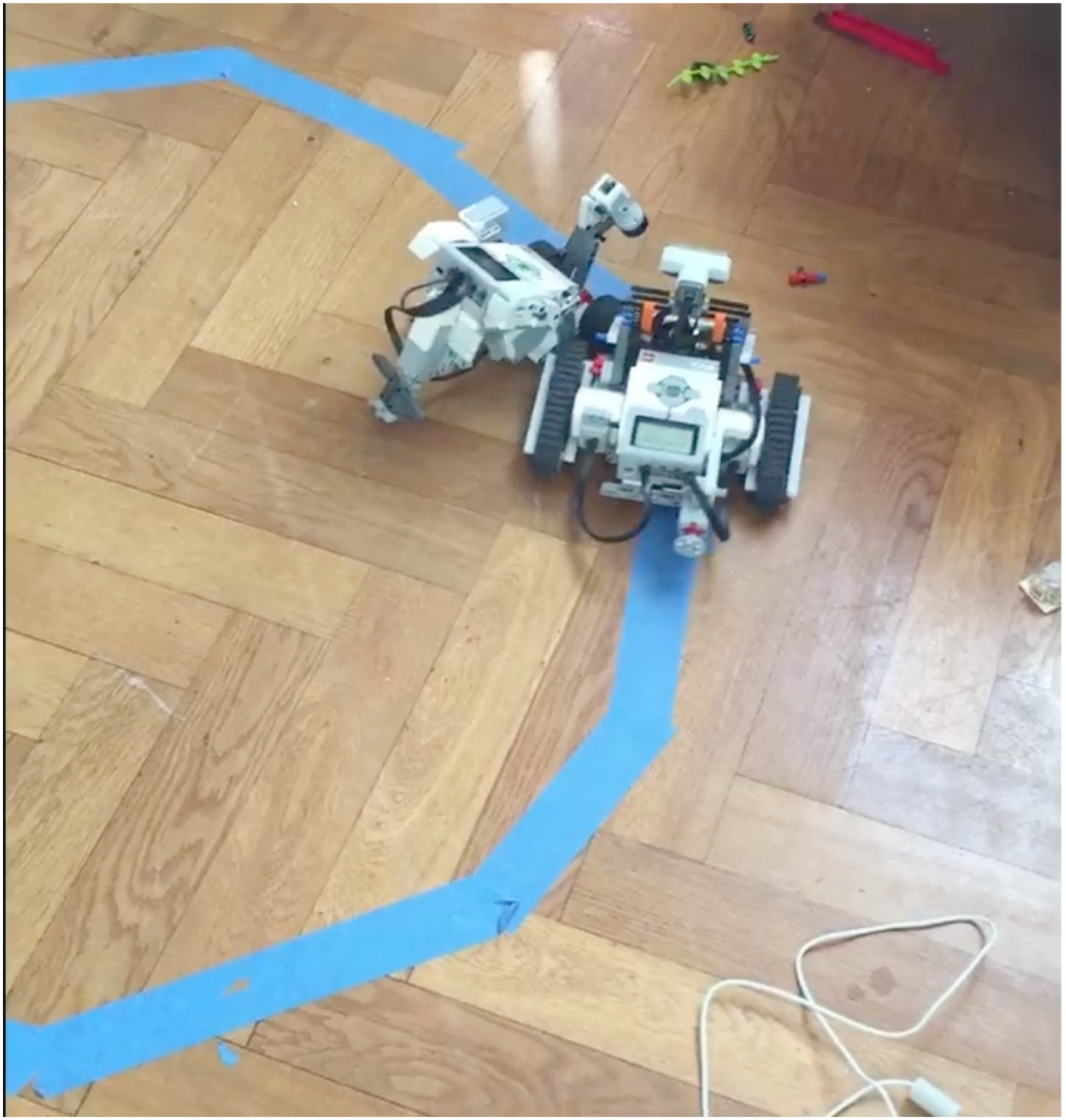


Options:
- Keep it cuz it's cool (who needs to win anyway)
- Improve it
- Remove it, try a different approach

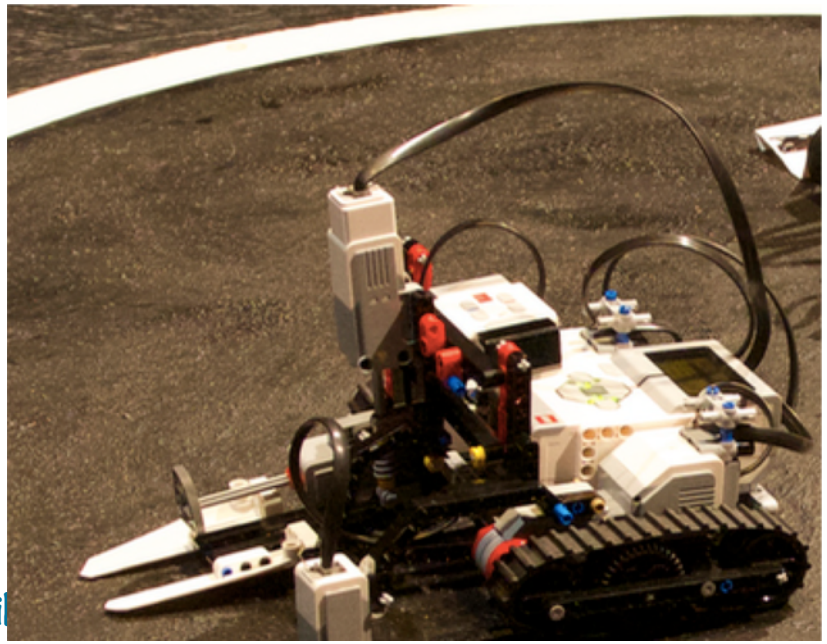
Simpler was better

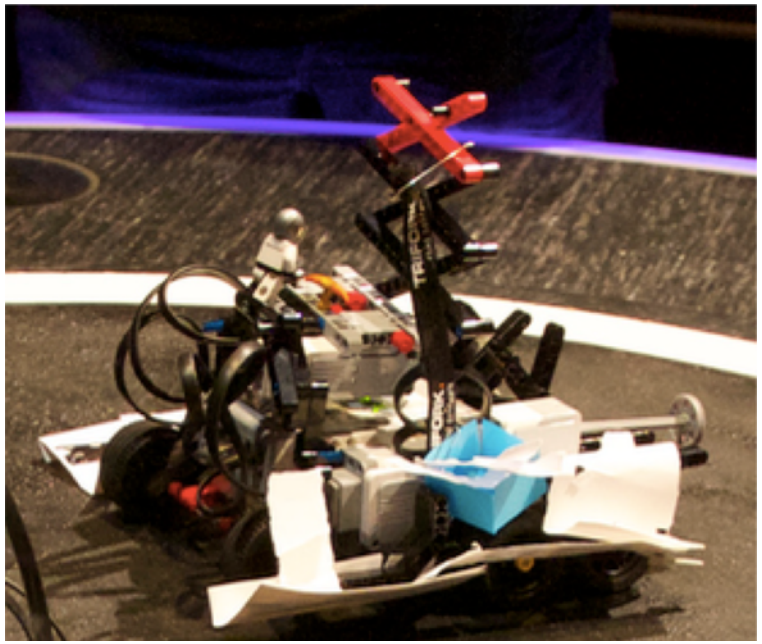


Field testing = Success by 100 failures



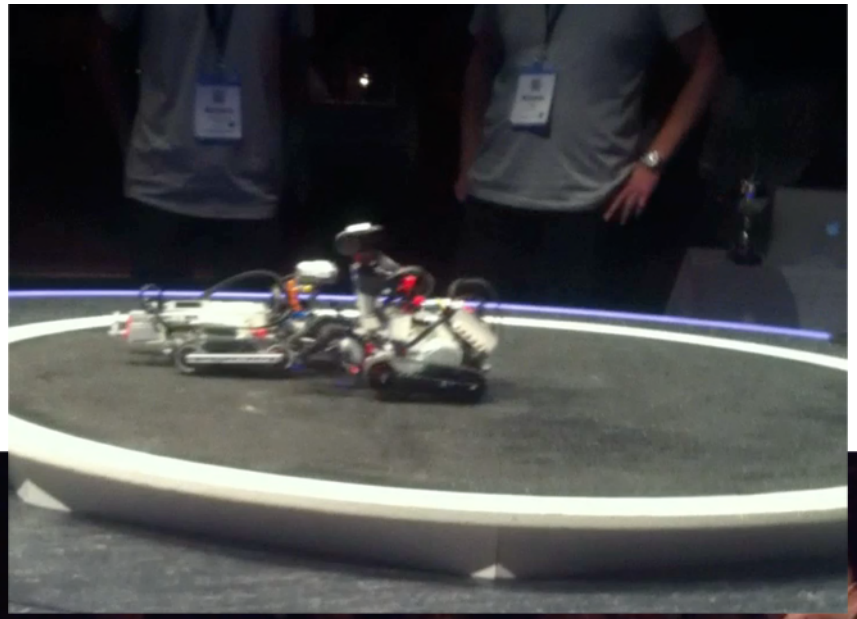
Henrik Kniberg





Henri

Henrik Kniberg



Henrik Kniberg

Henrik Kniberg

How could they win?

Building skill? No.
Programming skills? No.
Luck? Partly, but not entirely.

1) Clear goal
2) Low self-confidence
3) Emergent design
4) LOTS of field testing!

Henrik Kniberg

Henrik Kniberg

The biggest problem in the world

The Biggest Problem In The World!

Radical innovation needed

Energy production & storage

Carbon capture & storage

Transportation

Agriculture

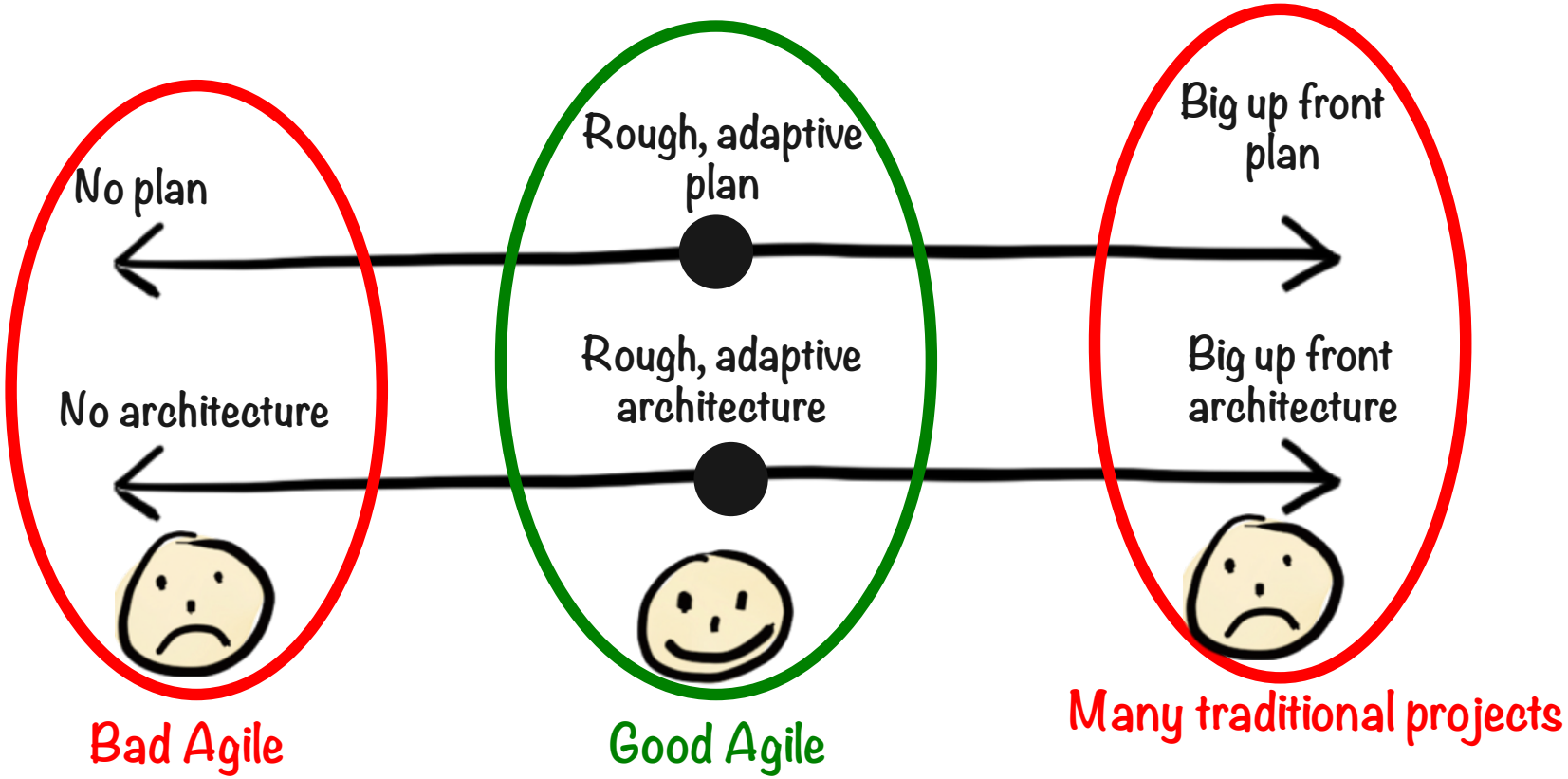
Henrik Kniberg

Wrapup

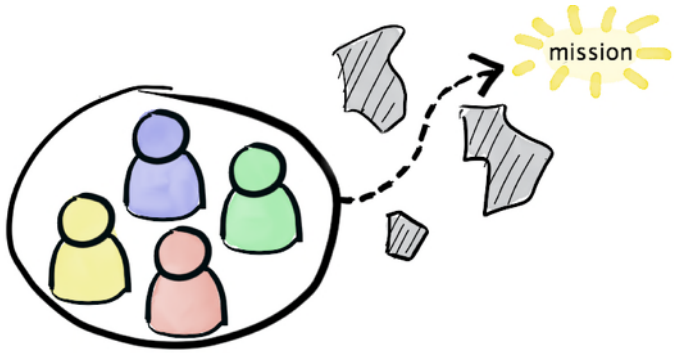
15,000 person-years of experience

Henrik Kniberg

Don't go overboard with Agile!



Find (or create) agile companies!



How to recognize real agility:

- Work in small, cross-functional, self-organizing teams
- Release often & get real user feedback
- Focus on Value rather than Output/Cost
- Experiment a lot with product & process

Beware empty buzzwords