Digital Transformation in the Automotive Industry

Digitale Transformation in der Automobilindustrie

Dr. Michael Nolting

Lecture 7

Tutorials

- Homework will be reading the book from Gene Kim „The Phoenix Project“

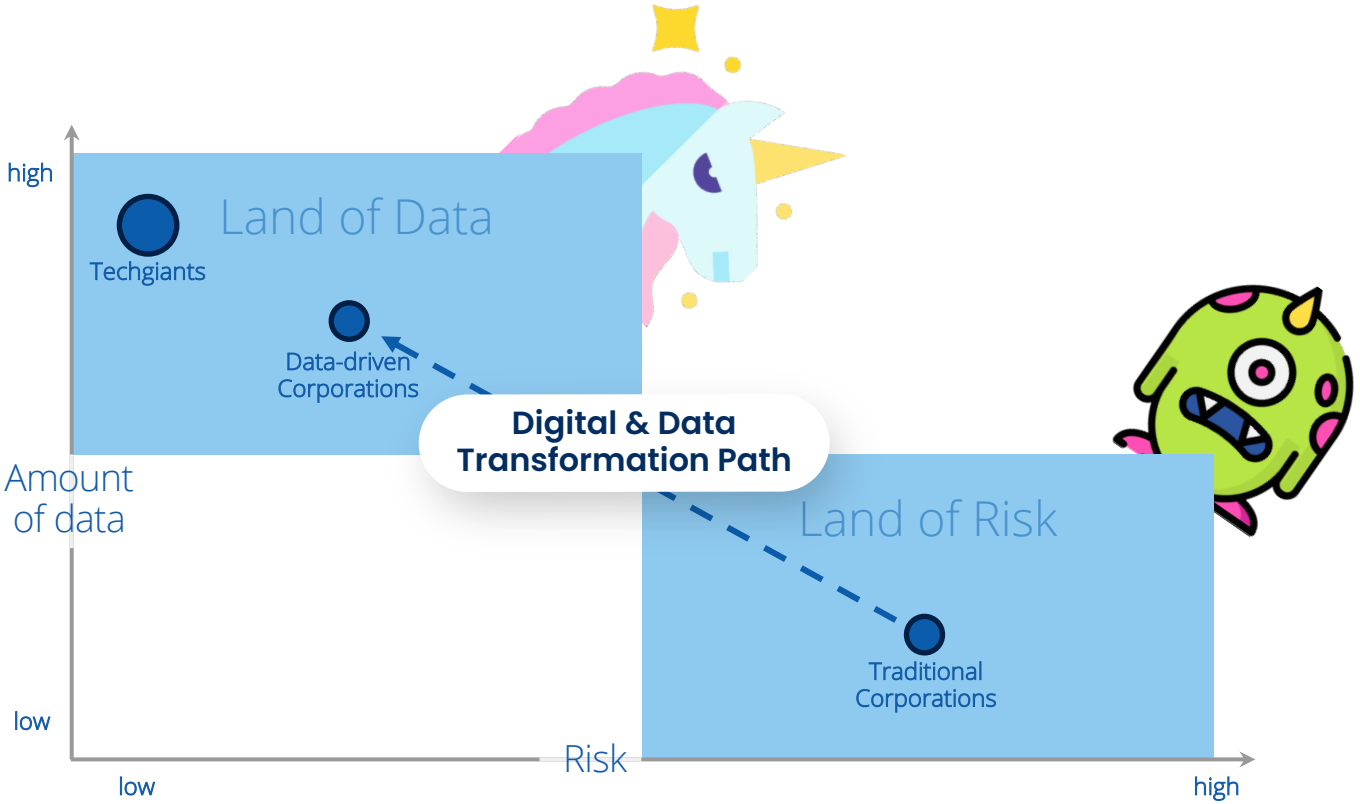
The homeworks are optional and not relevant for the exam

Lecture Overview

1. Introduction: Why Digital & Data Transformation	**7. Culture & Organization**
Homework 1: Reading 60 mins the Phoenix Project	Homework 7: Reading 60 mins the Phoenix Project
2. The World is Changing: ACES & VUCA	**8. Examples of Digitalization Projects I**
Homework 2: Reading 60 mins the Phoenix Project	Homework 8: Reading 60 mins the Phoenix Project
3. The Technological Disruption	**9. Examples of Digitalization Projects II**
Homework 3: Reading 60 mins the Phoenix Project	Homework 9: Reading 60 mins the Phoenix Project
4. Challenges for the Transformation - Innovation	**10. TESLA as THE Digital Player**
Homework 4: Reading 60 mins the Phoenix Project	Homework 10: Reading 60 mins the Phoenix Project
5. Challenges for the Transformation - Legacy	**11. Q & A – Exam**
Homework 5: Reading 60 mins the Phoenix Project	
6. How to Transform Into a Techgiant	
Homework 6: Reading 60 mins the Phoenix Project	

Digital & Data Transformation

We need the best people (the unicorns) to master this way as fast as possible



Agenda

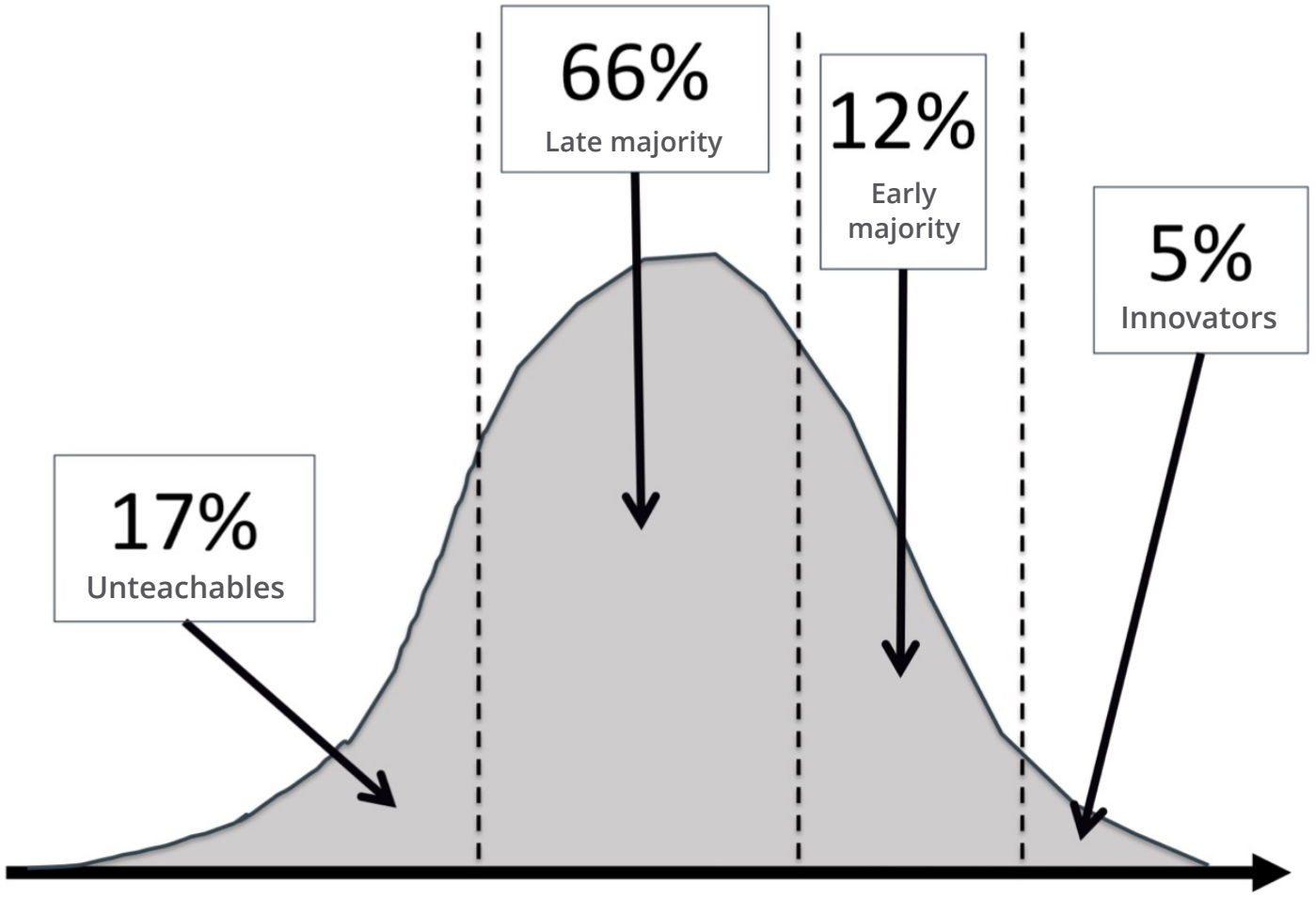
01 Mindset & People

02 Agility & Scaling

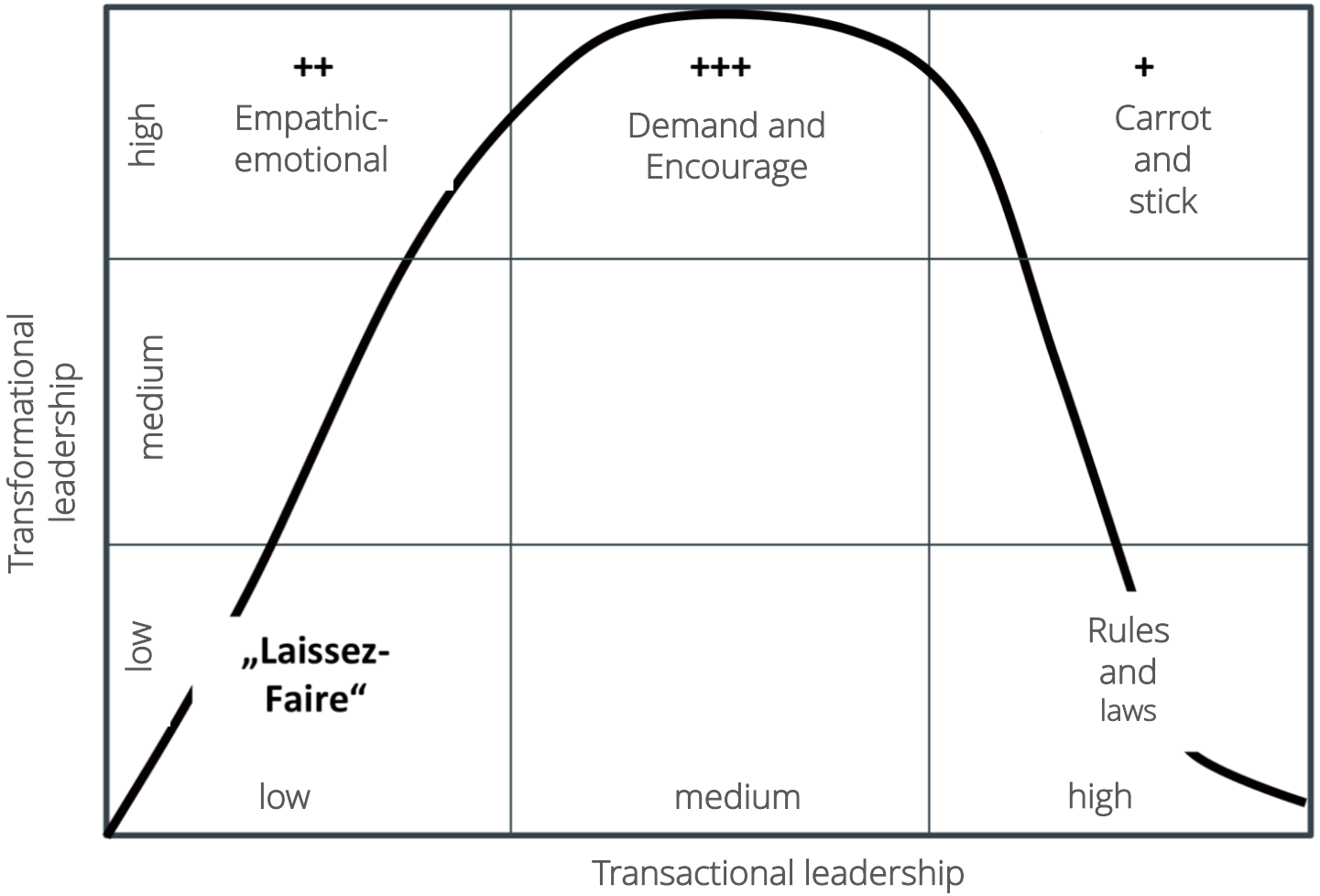
03 Responsibility

04 Summary

Acceptance Curve

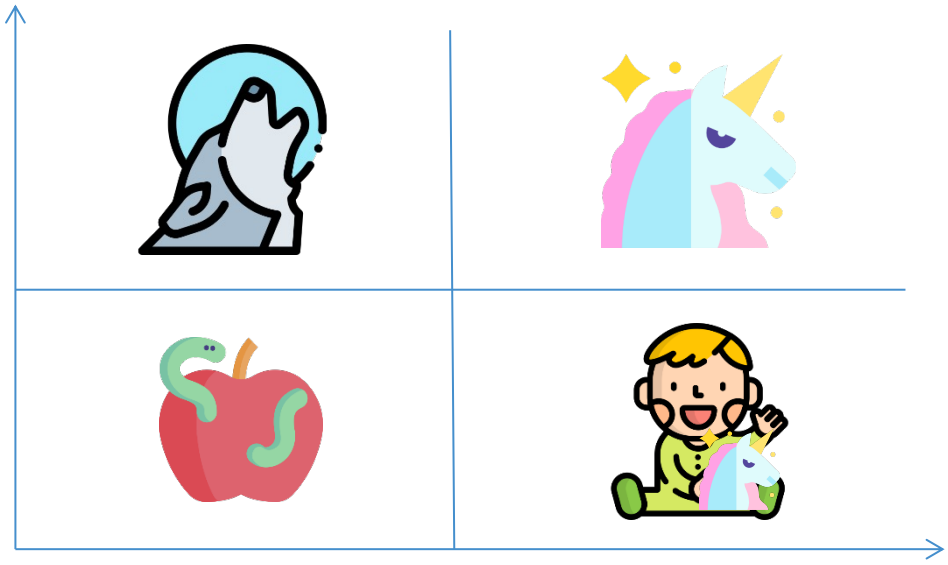


Transactional vs. Transformational Leadership



How to find the unicorns? Value-based hiring

Results



Values

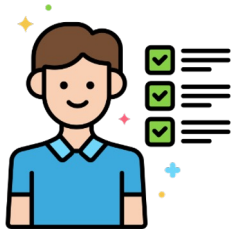
What is the most important value? Trust!

Trust

=

Credibility

+



Reliability

+

Intimacy

The biggest challenge in corporations.

Self-Orientation

Old world: Employer-centricity

Quelle: Neumann (2013)

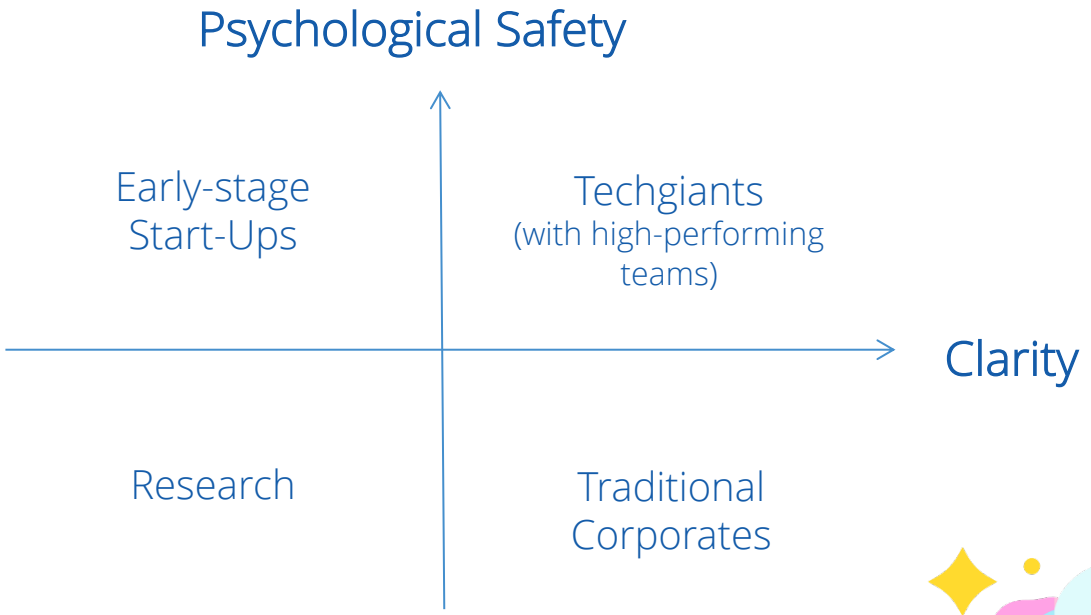
Why is it not working any more?
What has changed?

The unicorns of today need a different environment, which is driven by clarity and psychological safety.

You have to understand them as your customers.

The Secret #2 of the techgiants – or – how to attract unicorns

- High-performing teams need psychological safety and clarity



Psychological Safety vs. Psychological Danger

https://www.plays-in-business.com/psychological-safety-why-employees-need-it/

How to make a UNICORN (for us a DEVELOPER) happy?

Top priorities for the new generation of software developers:

1. Clarity & psychological safety
2. A mature environment for SW development
3. Career paths & growth
4. Money

1 Clarity & psychological safety

2 A mature environment for software development

CI/CD, DevOps, ...

3 Career paths & Continous growth

Money 4

You have to see the employee as your customer

Clarity: 1 (digital) SOP per Day

Own your code! Own your data!

The books from Gene Kim:
The Phoenix Project &
The Unicorn Project

	1	2	3	4	5
KEY ACTIONS	Agile Development	Dynamic stages	Test Automation	Feature Toggles	Democratization of data and ML
WHAT	Own your Code! Growing A-level teams Working cross-functional with the business units – without silos	Pre-Submit-infrastructures to spin up releases of feature branches in the cloud. Like GOOGLE.	Tests run on each and every code change. Optimal test coverage. Like UBER.	A/B tests. Respecting our long-running legacy processes. Like FACEBOOK.	Centralized the data on a single platform. We know our data sets and our business departments leverage their potential. Like AMAZON.
WHY PAIN POINTS	Efficient requirements engineering	Reduction of integration cost	Reduction of test cost	Eliminating process-driven bottlenecks	Data-driven decisions; Reducing risk

Digital Transformation

Data Transformation

MOD
PROJEKTHAUS

Psychological Safety: On Eye-Level with the Business

We develop customer-centric, scalable and robust mobile online services for thousands of commercial customers; and transform VWCV into a data-driven company.

I. **Design & Development**

We work very close with the business departments and are co-located. We develop digital services for B2C- and B2B-markets.

Cross-functional

We work in cross-functional teams and leverage modern agile frameworks such as SCRUM, SAFe, LEAN etc.

However, what really matters is output.

III. **Think like a Start-Up, act like a Grown-Up**

We act fast. If we make failures, we learn from them. We are not afraid of making failures. We share and do not hide failures. We comply to the corporate processes.

IV. **Continuous improvement & customer obsession**

Continuous improvement is in our DNA. We are customer obsessed and are a core-cell of the ongoing transformation of Volkswagen.

And finally.. generating meaningful results (each and every day)

Leveraging agile methods like SAFEScrum and OKR

Soft factors: SWAG, hackathons and the best hardware

Mission stickers

Best hardware; IDE of your choice: Your tools, your rules

Google Fridays

Hoodie

Bi-annual hackathons

A lead magnet : A recruiting video covering all these aspects

Eventually, recruiting is a sales process.

How to accelerate?
The self-accelerating, data-driven recruiting fly-wheel

A lead magnet

GETTING HIGH-QUALITY LEADS
Landing page: An awesome & very specific role description

e.g. Python Data Engineer, Angular Frontend Developer, GO Backend Developer, AWS Cloud Engineer, Flutter App developer

QUALIFYING THESE LEADS
Better-matching employees

BEST DEAL

Sales happens here!

KEEPING OUR LEADS HAPPY
Great results of employees & personal growth

MOIA

Great marketing

More data / improving the role description

GETTING EVEN BETTER LEADS
Even more leads

Job description: Automotive Software Developer - Backend

Generated hundreds of high-quality leads over time.

Purpose? Vision? Can I grow?

Aufgabenbeschreibung:

To revolutionize the future of transportation, one has to go one step further and set standards today. Smart technologies which are strongly taking over driving tasks and providing improved safety and comfort are already in place in our commercial vehicles today. But our vision is that "We make transportation smart". So, we need visionary thinkers and creative developers as well as experts in the fields of machine learning, artificial intelligence, sensor technology and software development to achieve this vision. If you want to leave an impact on the world, join us!

An exciting position has become available within Volkswagen Commercial Vehicles as an Automotive Software Developer - Backend (f/m). You will be part of an experienced international team, working with agile software development processes and standard design methods in the heart of Hanover. You will gain significant experience throughout the complete stack: The entire way from our ConnectFleet service backend down to the online connectivity units in the car. On top of that we are further developing and operating Volkswagen commercial vehicles' Connect Fleet platform - based on a cutting-edge cloud services architecture - which provides premium personalized, contextual and seamless fleet management experiences across all touchpoints.

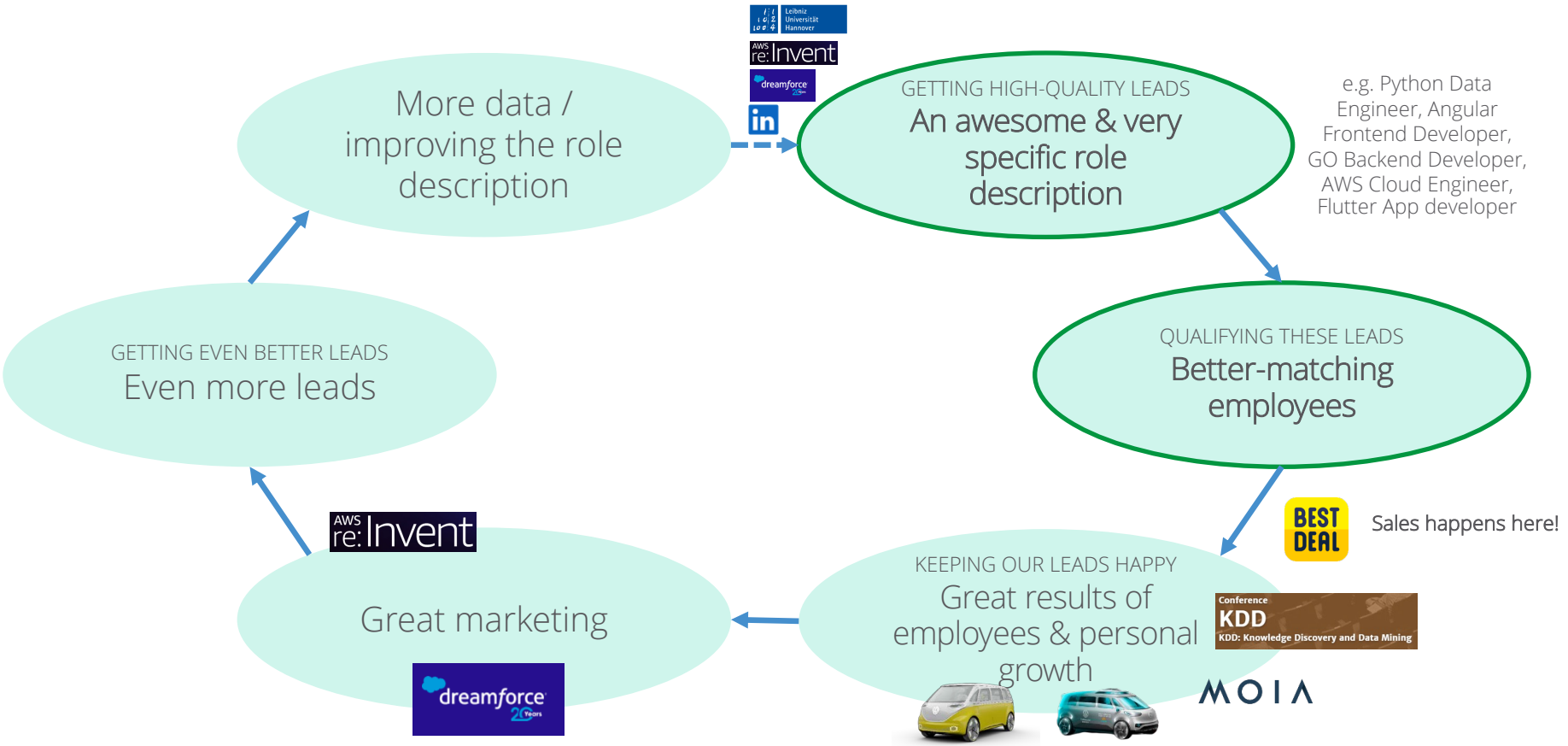
- Be a coder and develop features for our backend applications
- Help us materialize our long term vision by turning it to tactical goals and requirements
- Facilitate collaboration with other engineers, product owners, and designers to solve interesting and challenging problems across our fleet platform.
- Be a valued member of an autonomous, cross-functional agile team.
- Architect, design, develop, deploy and operate code that serve thousands of drivers in a DevOps manner.
- Hack on what you want during regular hack days and bi-annual hack weeks.
- Work from our awesome office in Hannover City where we solve problems in mobility solutions for the future

Is it a mature environment, where great things could happen?

Berufserfahrung, besondere Kenntnisse:

- Basic understanding of writing clean, standards based code for backend applications
- Fundamental web technologies (e.g. HTTP, REST, and JSON) are second nature
- Familiarity with open source development workflow using GIT, Fork, Pull Requests, and branches
- Solid understanding of the software development lifecycle, and familiarity with modern Agile development methodologies; familiarity with test automation
- Good knowledge in Python, Go, PHP or Java
- You know and leverage the power of container systems like Docker as much as possible

How to accelerate?
Our secret: The self-accelerating, data-driven recruiting fly-wheel



Qualifying leads: One interview with 4 people is sufficient
Follow a clear recruiting script with a process to allow learning

To avoid the re-production of the leaders and enable diversity follow the 8-eyes principle. Trust your gut feeling!

1. Introduction (CV/department)

2. Team test
   Does the applicant fit into your team? And how?

3. Coding test
   "Please order the number 1..100."

4. Self-reflection question
   "Please tell us at least THREE weaknesses."

5. The airport question
   "Imagine you meet a friend at the airport in 5 years..."

Lieber Bewerber,

willkommen beim Team-Test, der vor unserem Bewerbungsgespräch läuft. Dieser ist dafür da, um ein wenig mehr über Dich erfahren – speziell Dein Teamprofil. Jedes Ergebnis, das rauskommt ist gut. Es gibt kein schlechtes und kein gutes Teamprofil. Diversität macht erst gute Teams aus.

Der Test funktioniert wie folgt:

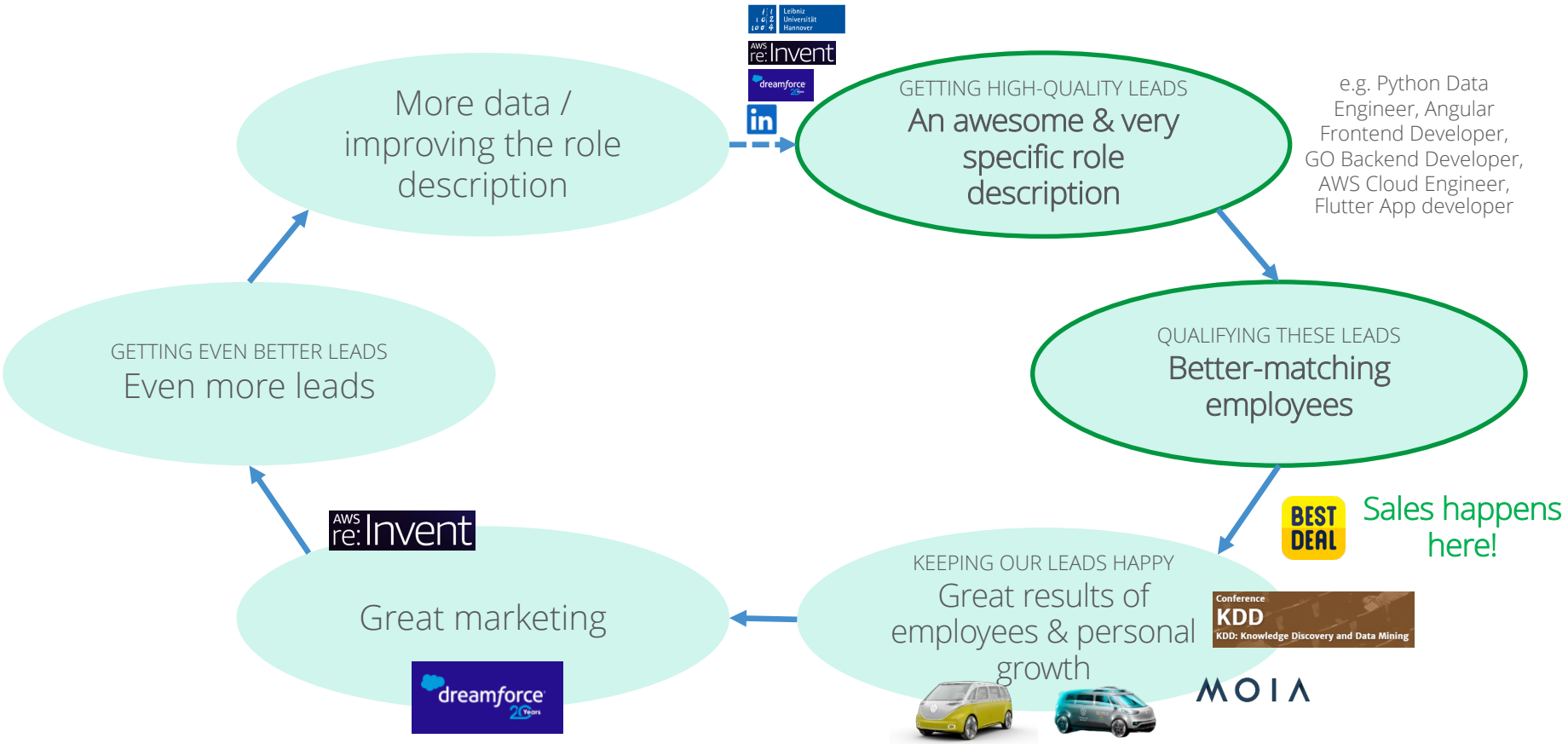
Bitte wähle für jeden der vier Bereiche aus, ob eher das linke oder das rechte auf Dich zutrifft!

Extrovertiert	Introvertiert
Entwickeln gern Gedanken, im Gespräch mit anderen	Denken lieber erst gründlich nach, ehe sie sprechen
Treffen gern andere und suchen viel Kontakt	Arbeiten gern und gut allein
Widmen sich gern gleichzeitig verschiedenen Aufgaben und Aktivitäten	Konzentrieren sich gern auf eine Aufgabe
Mögen unvorgesehene Unterbrechungen	Lieben es nicht, unvorgesehen unterbrochen zu werden
Können aus dem Stegreif reden und sprechen viel	Bereiten sich auf Reden gut vor und halten sich ans Konzept
Können impulsiv sein	Eher zurückhaltend und besonnen

Hier musst Du z.B. auswählen, ob Du eher „Extrovertiert" oder „Introvertiert" bist.

4 questions-based team test

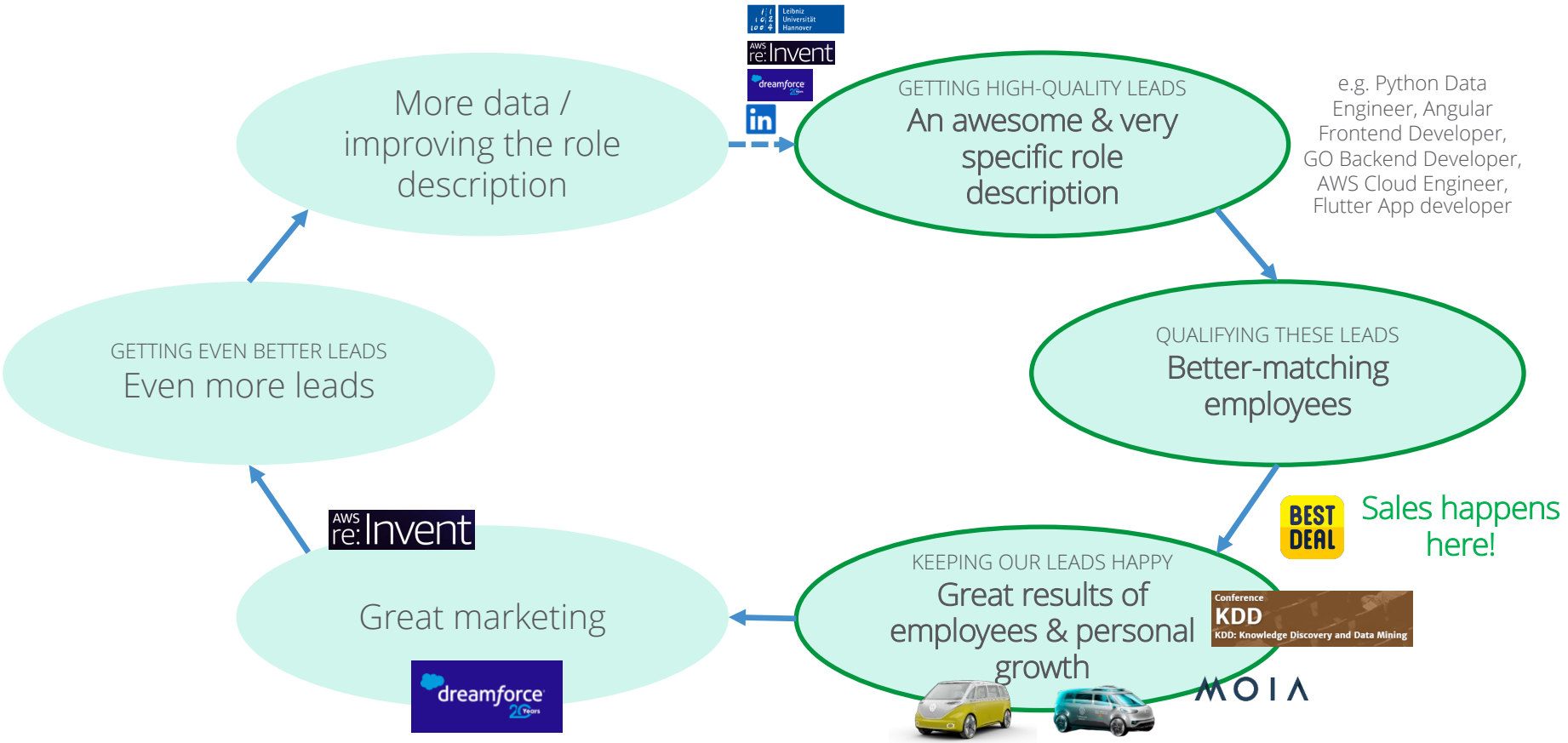
How to accelerate?
Our secret: The self-accelerating, data-driven recruiting fly-wheel



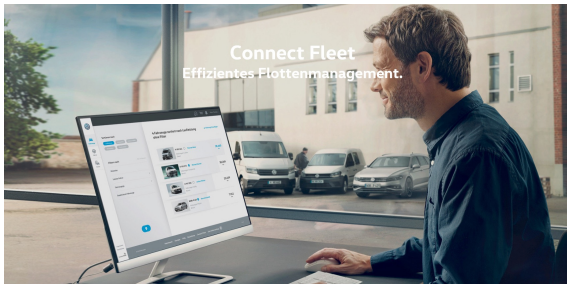
Final interview with our CIO and HR. And then... the offer!

BEST DEAL

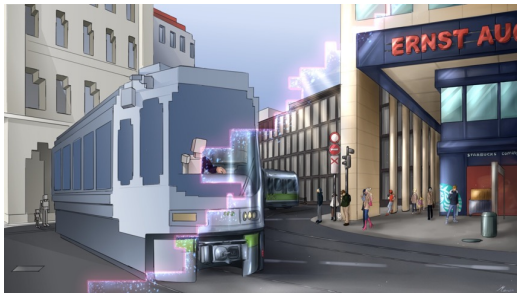
How to accelerate?
Our secret: The self-accelerating, data-driven recruiting fly-wheel



Great results of employees



Group solution for all brands for fleet management



Three research projects (7 PhD students) in important future areas like data, 5G etc.

Supporting DPP, autonomous driving etc.

300 Terabytes of real mobility data from MOIA / democratized to all brands

Transforming production workers into click workers

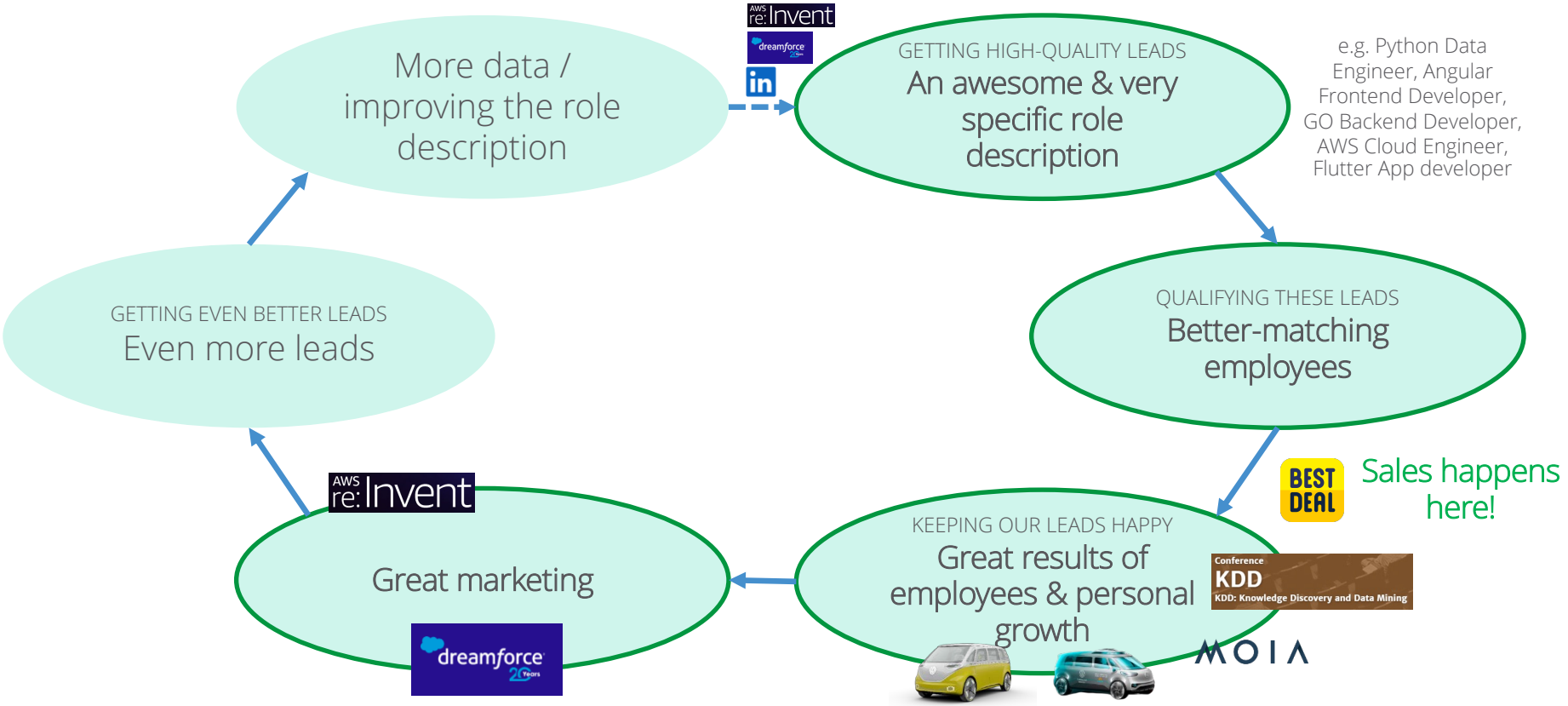
Paper at A-star conferences like KDD

Keeping your developers happy

Career paths of developers enabling personal growth

Level	Technical	Collaboration	Recruiting	Scope	Mentorship
Junior	* Crafts straightforward pull requests that are easy to review * Owns the implementation of their projects once the scope is clearly defined * Operational with VW core technologies * Familiar with VW's testing practices and leverages them effectively. * Participates in On-call rotation	* Operational with all core working methodologies * Gives precise technical updates on tasks they have completed and bugs that have been raised * Voices concerns or need for clarification to their crew lead * Accepts feedback graciously and learns from experience		* Owns well-defined tasks with limited ambiguity * Give input to the definition of the scope and technical approach to their projects * Are able to quickly align with more senior engineers on scope and technical approach	* Reviews simple pull requests for bugs and correctness
Mid-Level	* Is trusted to give quality PR reviews on small to medium level changes * Is comfortable working on any part of the stack, after a short ramp up	* Operates as an Engineering referent on the framing of well defined problems with some ambiguity		* Owns tasks that contain some ambiguity * Is able to find creative solutions to implement them elegantly and effectively, with some guidance from more senior engineers * Is able to work side by side with a more senior engineer on a large ambiguous project	* Good pair programming partner, comfortable as either driver or navigator * Share their technical knowledge internally * Contributes positively to Engineering team
Senior	* Reasons about the design of our existing systems, and understands what changes would match or contradict a given design * Solves self-contained problems that are not completely well defined and contain some ambiguity * Has proficiency or will quickly learn all core technologies * Identifies opportunities for better testing, either in coverage and in methodology	* Owns the discussion of how to solve self-contained technical problems in a ticket/issue * Provides relevant and actionable feedback to other engineering team members on technical matters and personal growth * Is role buddy for other members of the Engineering team		* Identifies problems to solve * Owns tasks that contain some ambiguity * Is able to find creative solutions to implement them elegantly and effectively * Engineers at this level don't need guidance from more senior engineers for the organization to be productive on scoped initiatives * Contributes in a significant volume to the Engineering team, in a way that has impact and help the team move forward with defined next steps	* Gives relevant feedback on design specs * Can drive a minor learning process in engineering (e.g. eng lunches, retros, demos) * Is able to serve as role buddy and help ramp up new engineers to productivity
Lead	* Designs new solutions from scratch for difficult problems that are often ill-defined and ambiguous * Has good judgement of technical risk and is able to find solutions that minimize or avoid it completely * Speaks out regarding code maintainability and testability, identifies and executes on significant refactors * Speaks up to ensure quality of team's work and design decisions	* Communicates near-term plans and progress for a single team * Owns the discussion of important technical decisions * For each solution provided, can lay out the alternative solutions considered, the design tradeoffs made, and identify the technical risks involved and gather info from other teams to get alignment * Helps other engineers grow * Is a coach for a team member	* Is able to lead pre full day technical interviews	* Owns long-term projects for a team of several engineers * Breaks down high-level projects down to individual tasks * Can be productive and work toward business goals without any external guidance or support	* Can coach and grow engineers senior and lower * Identifies problems, opens issues and drive solutions to improve processes in the engineering community
Senior Lead	* Designs new systems that create new technical capabilities and solve difficult problems that require many engineers to build * Develops effective engineering practices around on-call, dev tools, and planning to maximize the effectiveness of the team	* Communicates 6-month engineering roadmap for multiple teams * Communicates plans to a broad audience to explain/justify their reasoning for major tech choices * Owns the decision process for major technology choices * Demonstrates leadership within the team about priorities and planning	* Leads each type of technical interview * Gives high-quality interview feedback that lead to solid and confident hiring decisions * Contributes to onboarding, and/or training newer members of the team	* Owns long-term projects that involve many engineers that are cross-cutting and impact multiple crews * Owns major technology choices such as databases, programming languages, CI tools, vendor services	* Mentorship impact improves engineers across multiple crews * Can coach and grow engineers principal and lower * Drives the creation (and removal) of processes to tackle wider problems in the Engineering team and beyond
Principal	* Is able to guide the development of long-term high-impact high-risk projects that involve many engineers * Reasons about how all of the engineering systems are designed and how they interact with each other, and has relevant feedback on which systems get built	* Engages with other functions within the company and creates plans that align Engineering goals with goals of other communities * Owns Engineering OKRs for a broad segment of the company * Contributes to the strategy of the whole company	* Can manage the recruiting process for an individual candidate * Can manage the process for the hiring decision for individual technical candidates	* Owns engineering work for a broad segment of the company	* Mentorship impact improves engineers and other roles across the entire company * Can mentor and grow engineers distinguish and lower * Drives the creation (and removal) of processes with a large impact across the company

How to accelerate?
Our secret: The self-accelerating, data-driven recruiting fly-wheel



Great marketing: Presenting at top-tech conferences of the world

Starting over with even better leads and more data ☺

Presentation of our data platform SNOWPARK at AWS Re:Invent 2022 in Las Vegas

Presentation of our data products at Dreamforce 2022 in San Francisco

Agenda

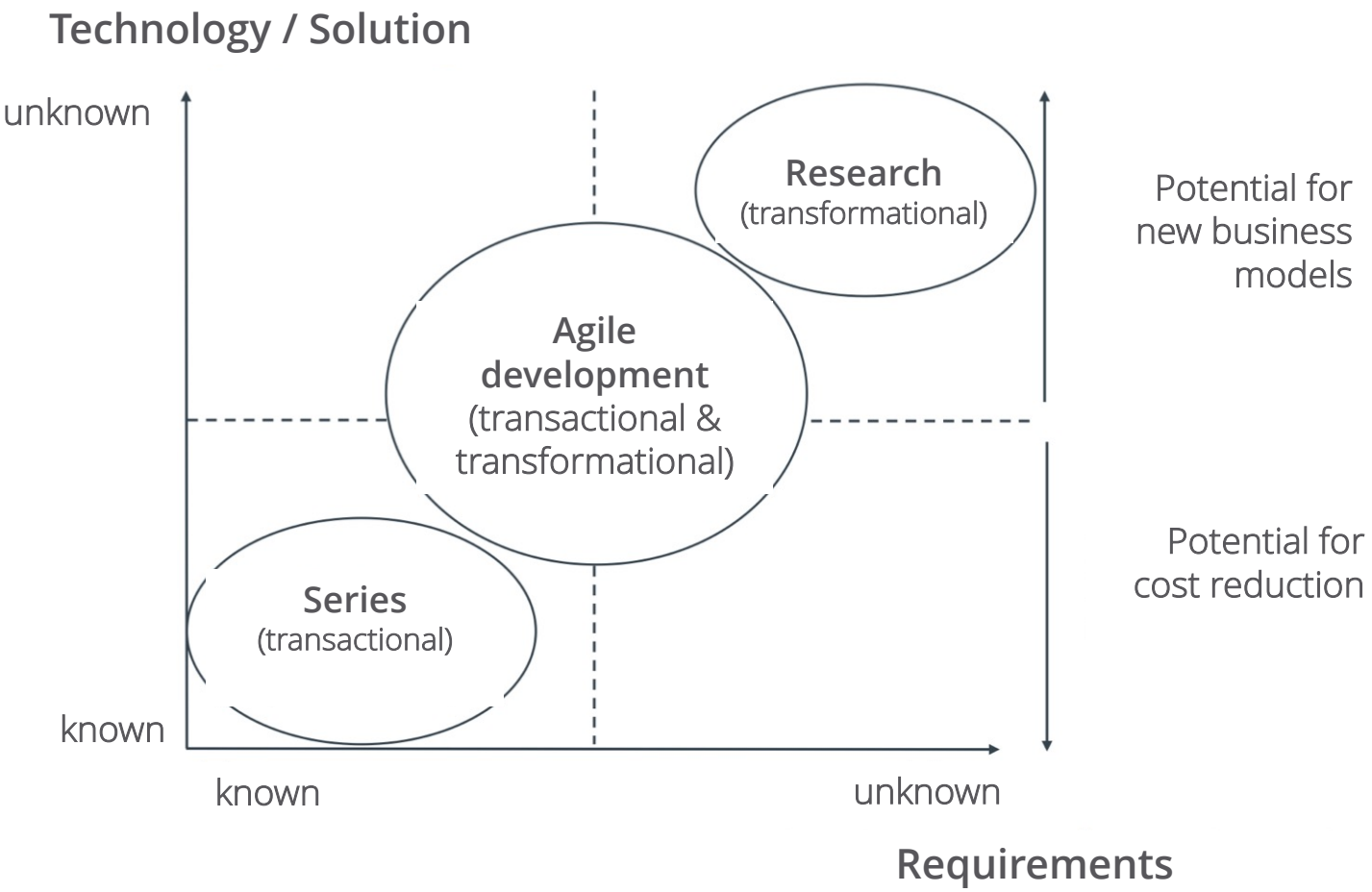
01 Mindset & People

02 Agility & Scaling

03 Responsibility

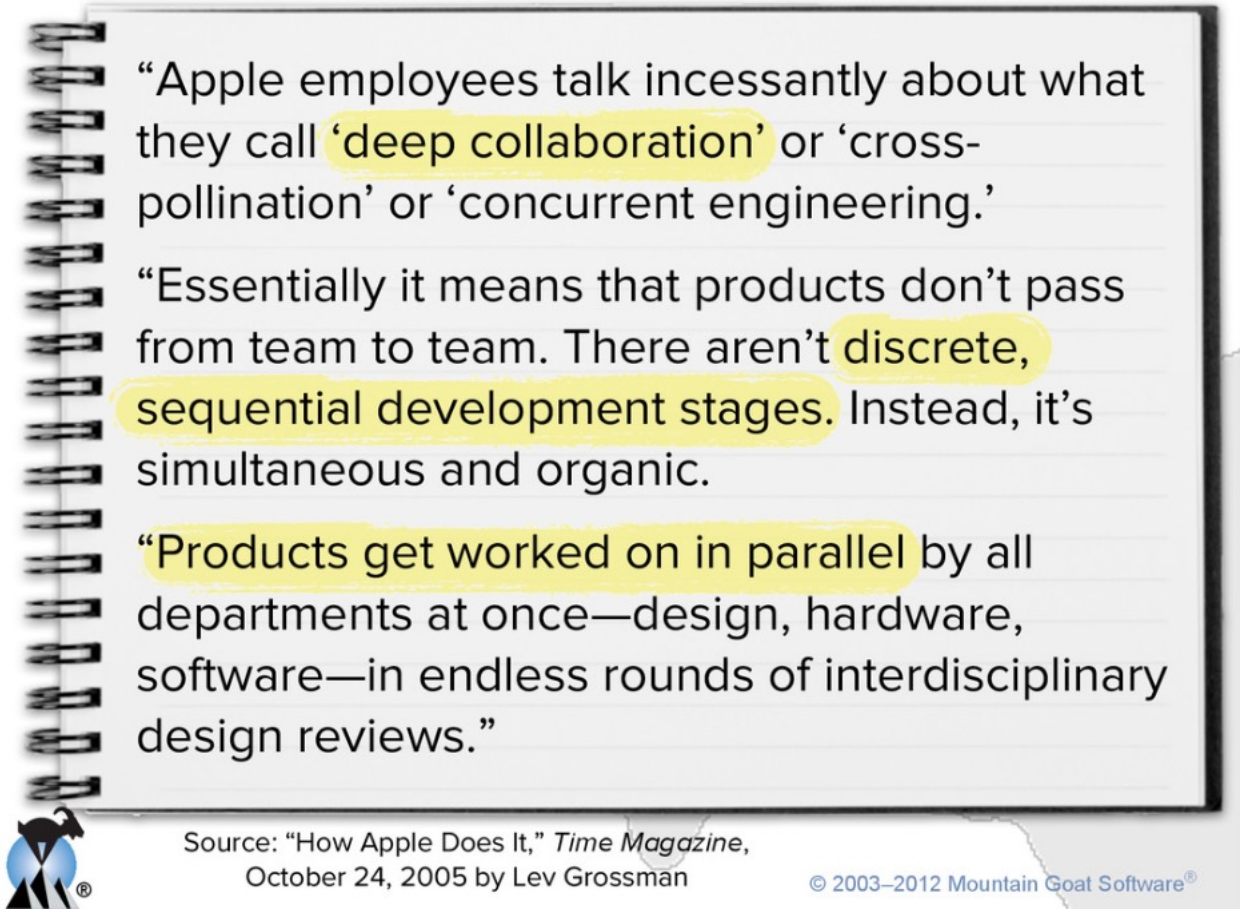
04 Summary

Stacey-Matrix



We‘re losing the Relay Race

https://www.mountaingoatsoftware.com/uploads/presentations/Getting-Agile-With-Scrum-Norwegian-Developers-Conference-2014.pdf


“Apple employees talk incessantly about what they call ‘deep collaboration’ or ‘cross-pollination’ or ‘concurrent engineering.’
“Essentially it means that products don’t pass from team to team. There aren’t discrete, sequential development stages. Instead, it’s simultaneous and organic.
“Products get worked on in parallel by all departments at once—design, hardware, software—in endless rounds of interdisciplinary design reviews.”
Source: “How Apple Does It,” *Time Magazine*, October 24, 2005 by Lev Grossman
© 2003–2012 Mountain Goat Software®

Scrum has been used by/for:

- Microsoft
- Yahoo
- Google
- Electronic Arts
- IBM
- Lockheed Martin
- Philips
- Siemens
- Nokia
- Capital One
- BBC

- Apple
- Nielsen Media
- First American Corelogic
- Qualcomm
- Texas Instruments
- Salesforce.com
- John Deere
- Lexis Nexis
- Sabre
- Salesforce.com
- Time Warner

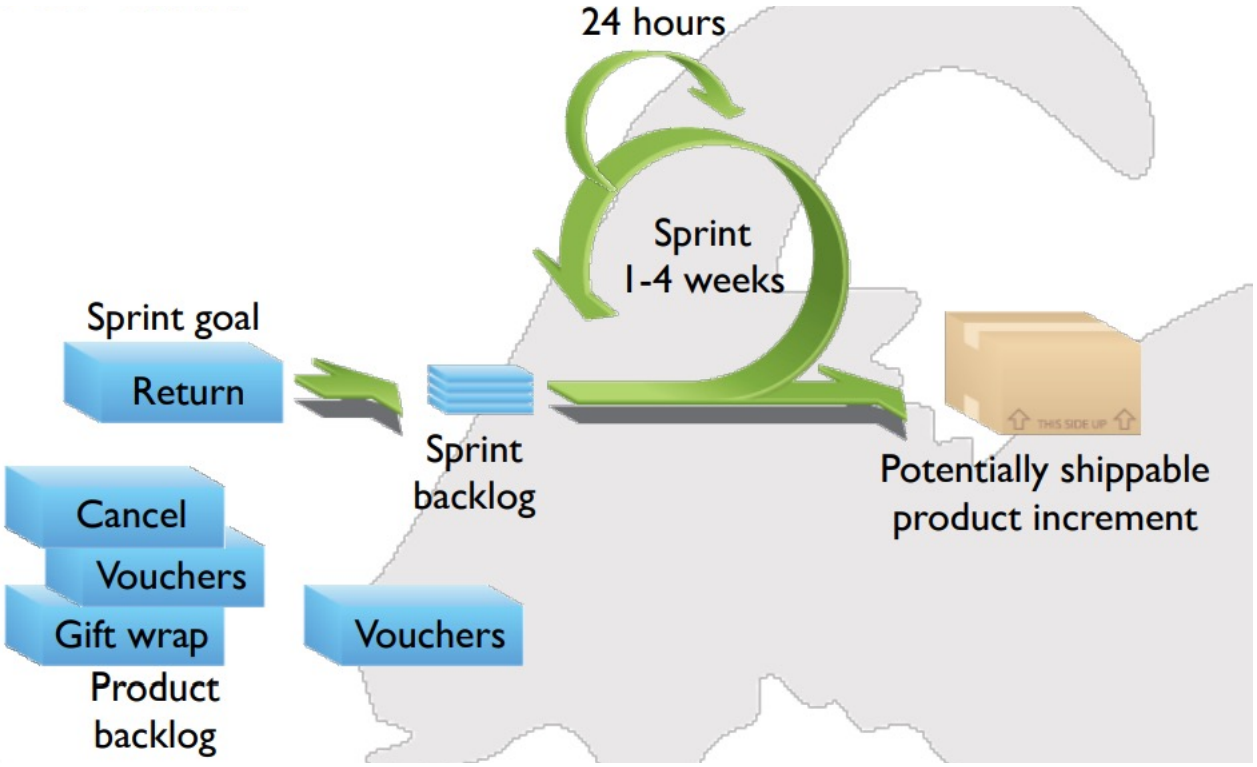
- Commercial software
- In-house development
- Contract development
- Fixed-price projects
- Financial applications
- ISO 9001-certified applications
- Embedded systems
- 24x7 systems with 99.999% uptime requirements
- the Joint Strike Fighter

- Video game development
- FDA-approved, life-critical systems
- Satellite-control software
- Websites
- Handheld software
- Mobile phones
- Network switching applications
- ISV applications
- Some of the largest applications in use

Characteristics

- Self-organizing teams
- Product progresses in a series of month-long “sprints”
- Requirements are captured as items in a list of “product backlog”
- No specific engineering practices prescribed
- Use generative rules to create an agile environment for delivering projects
- One of the “agile processes”

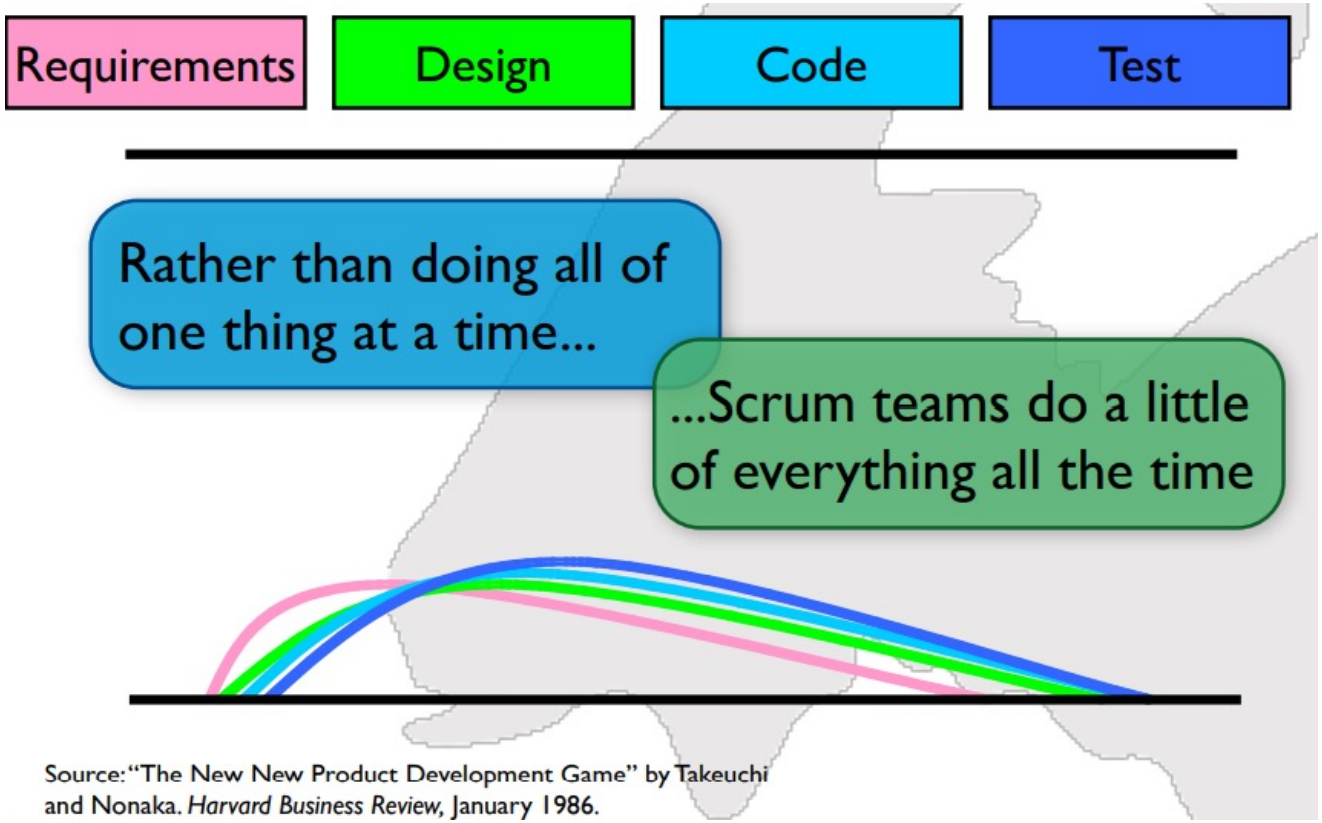
Scrum


24 hours
Sprint
1-4 weeks
Sprint goal
Return
Cancel
Vouchers
Gift wrap
Product
backlog
Sprint
backlog
Vouchers
Potentially shippable
product increment

Sprints

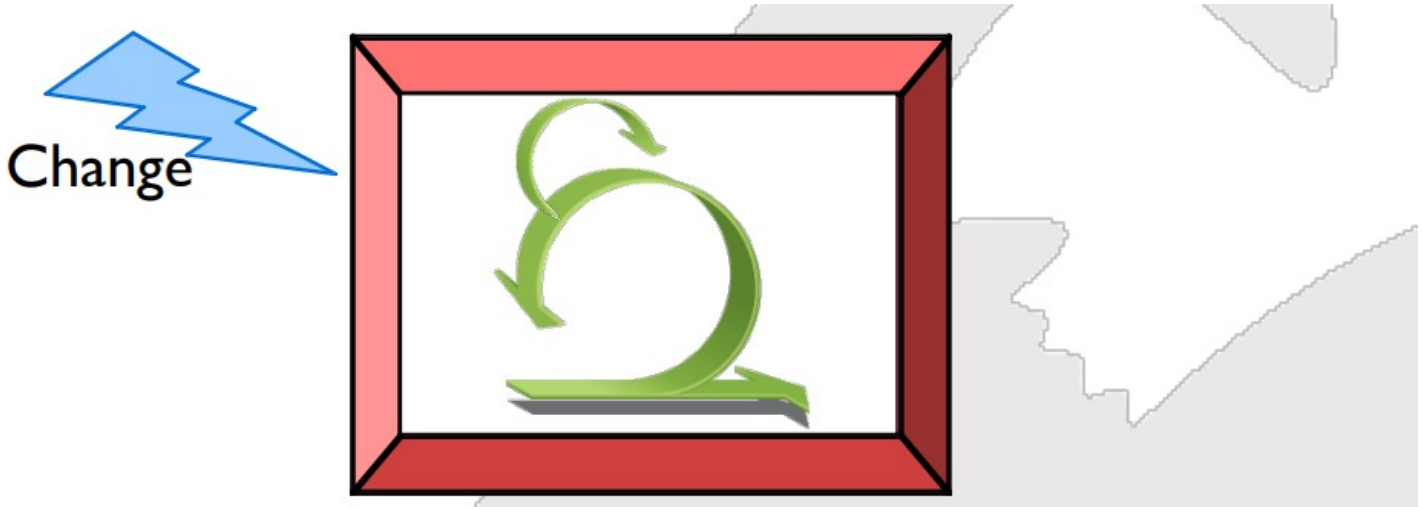
- Scrum projects make progress in a series of “sprints”
- Typical duration is 2 – 4 weeks or a calendar month at most
- A constant duration lead to a better rhythm
- Product is designed, coded, and tested during the sprint

Sequential vs. Overlapping Development



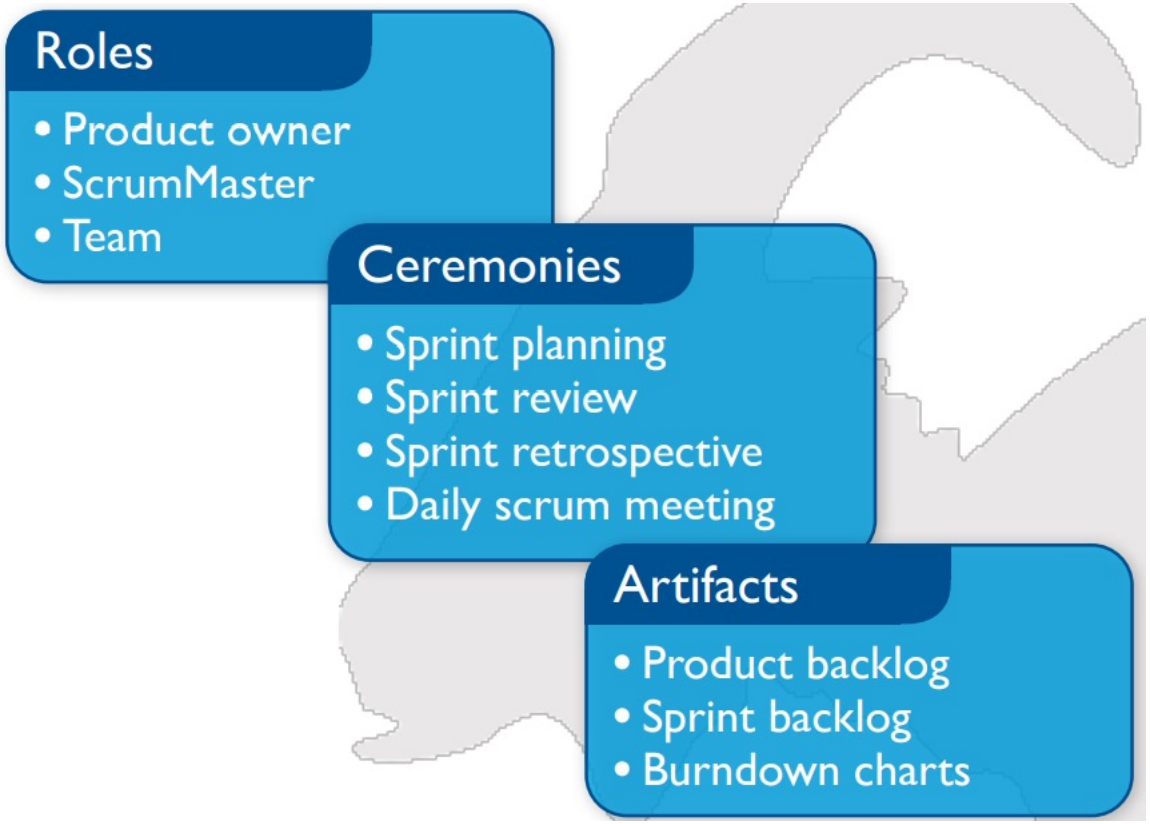
Source: "The New New Product Development Game" by Takeuchi and Nonaka. *Harvard Business Review*, January 1986.

No Changes during a Sprint



- Plan sprint durations around how long you can commit to keeping change out of the sprint

Scrum Framework



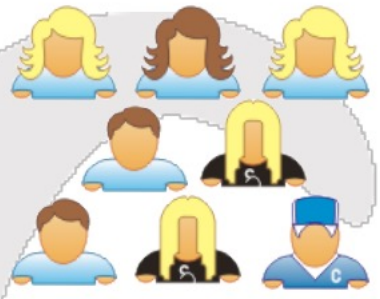
Product owner

- Define the features of the product
- Makes scope vs. schedule decisions
- Responsible for achieving financial goals of the project
- Prioritize the product backlog
- Adjust features and priority every sprint, as needed
- Accept or reject work results

The ScrumMaster

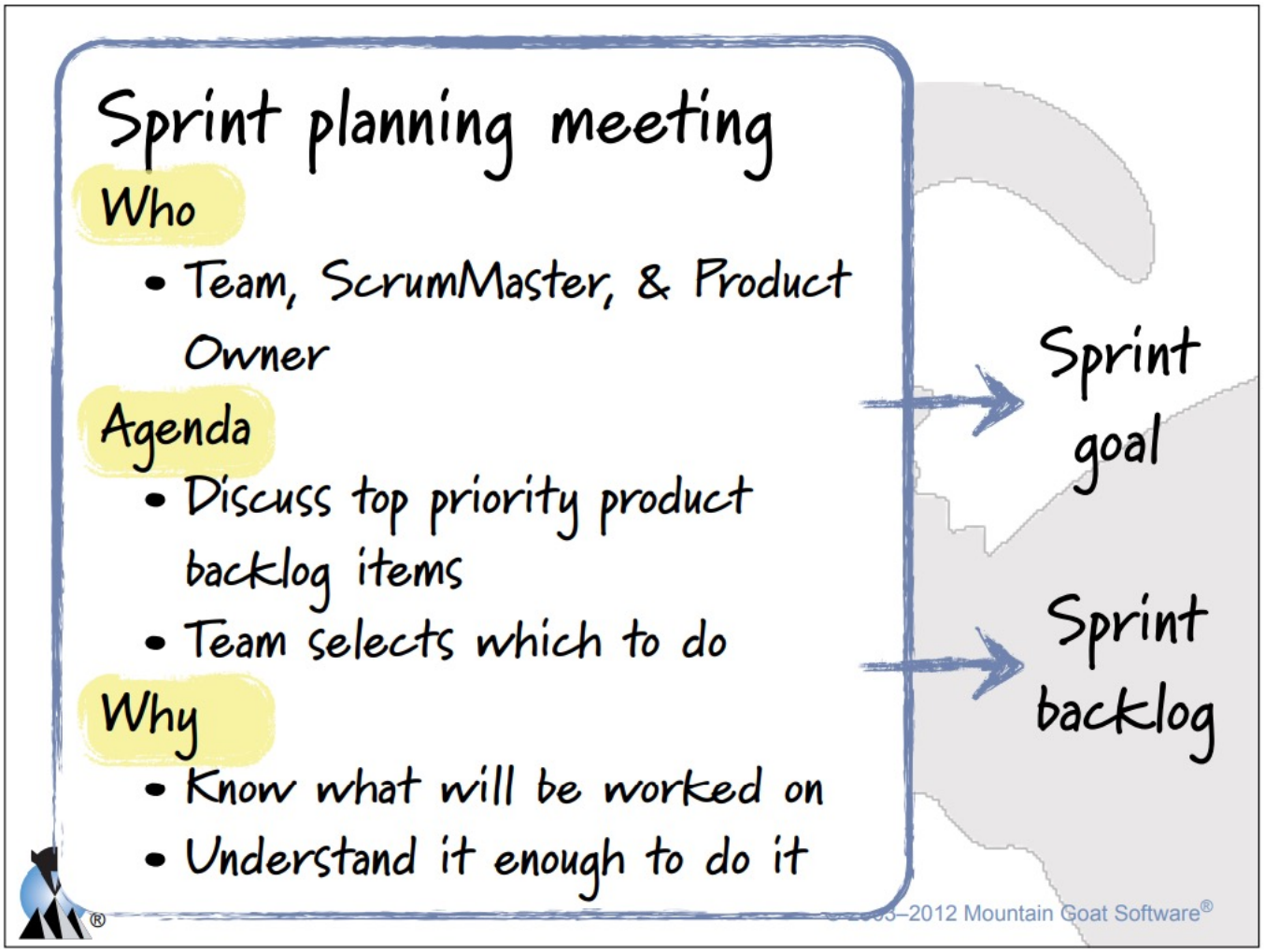
- Responsible for enacting Scrum values and practices
- Removes impediments
- Coaches the team to their best possible performance
  - Helps improve team productivity in any way possible
- Enable close cooperation across all roles and functions
- Shield the team from external interference

The team



- Typically 5-9 people
- Cross-functional:
  - Programmers, testers, user experience designers, etc.
- Members should be full-time
  - May be exceptions (e.g., database administrator)
- Teams are self-organizing
  - Ideally, no titles but rarely a possibility
- Membership should change only between sprints

Sprint Planning Meeting



Sprint planning

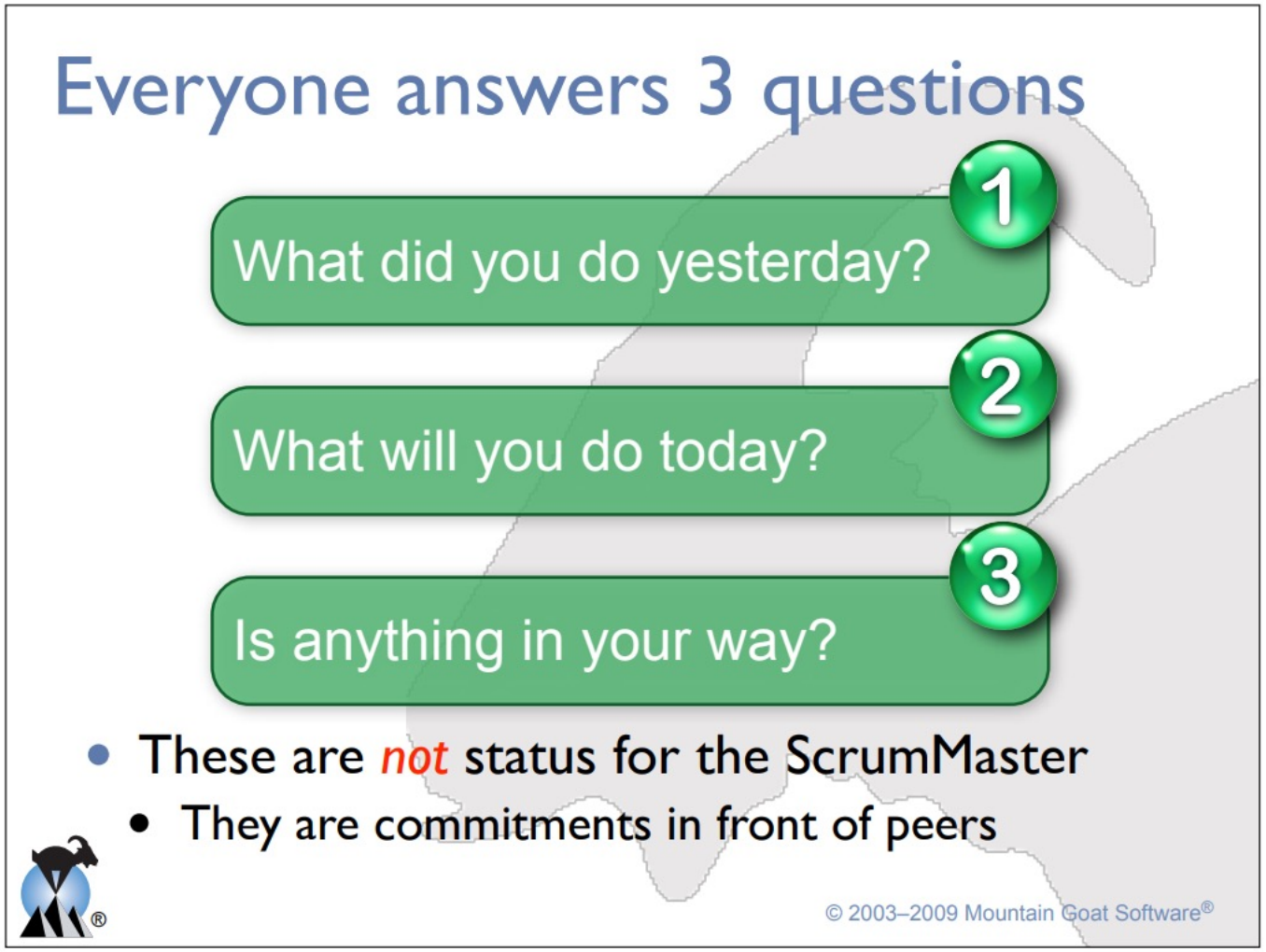
- Team selects items from the product backlog they can commit to completing
- Sprint backlog is created
  - Tasks are identified and each is estimated (1-16 hours)
  - Collaboratively, not done alone by the ScrumMaster
- High-level design is considered

As a vacation planner, I want to see photos of the hotels.

Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)

The daily scrum

- Parameters
  - Daily
  - 15-minutes
  - Stand-up
- Not for problem solving
  - Whole world is invited
  - Only team members, ScrumMaster, product owner, can talk
- Helps avoid other unnecessary meetings

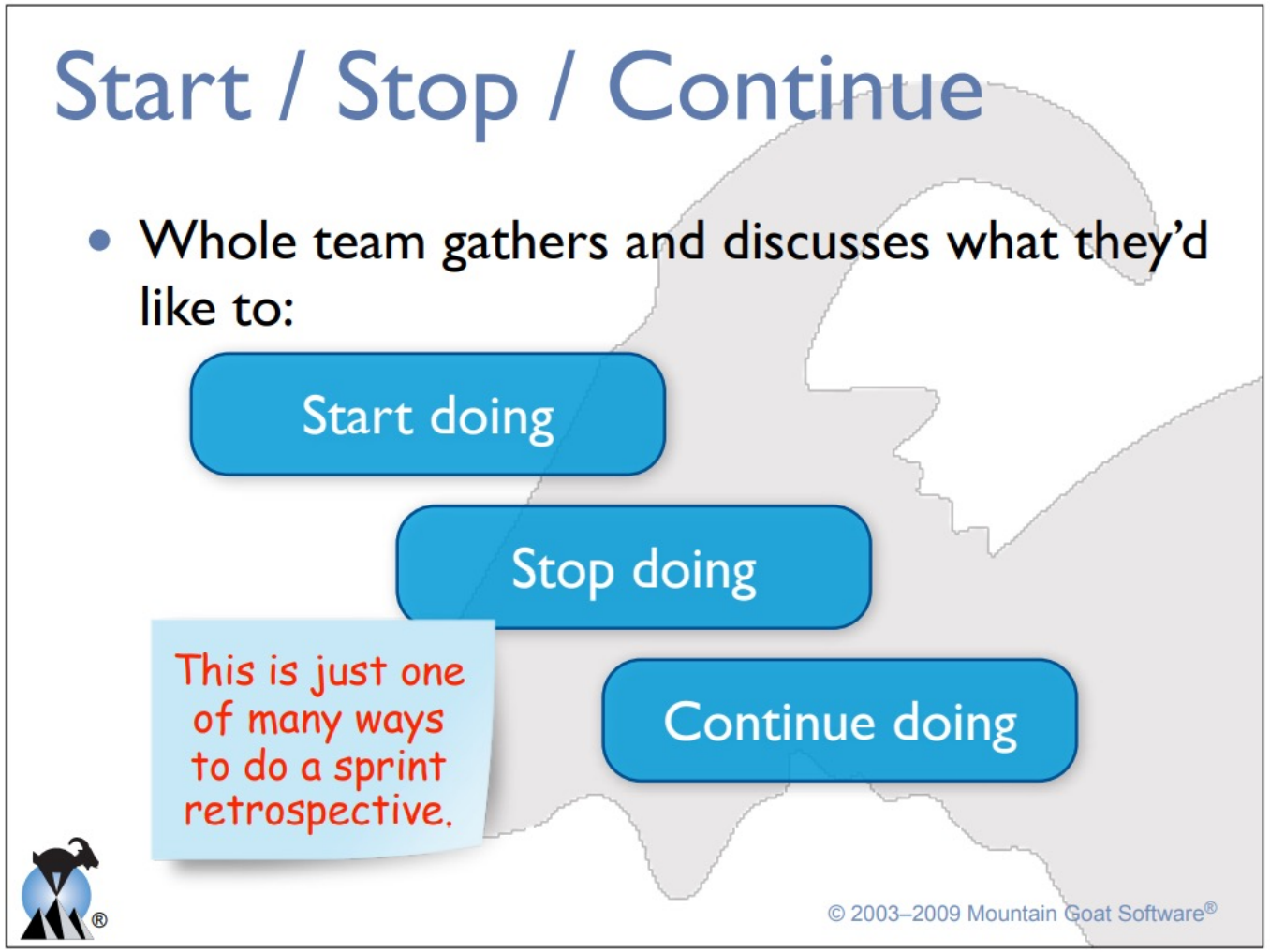

Everyone answers 3 questions
1
What did you do yesterday?
2
What will you do today?
3
Is anything in your way?
These are not status for the ScrumMaster
They are commitments in front of peers
© 2003–2009 Mountain Goat Software®

The sprint review

- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
  - 2-hour prep time rule
  - No slides
- Whole team participates
- Invite the world

Sprint retrospective

- Periodically take a look at what is and is not working
- Typically around 30 minutes
- Done after every sprint
- Whole team participates
  - ScrumMaster
  - Product owner
  - Team
  - Possibly customers and others


Start / Stop / Continue
Whole team gathers and discusses what they'd like to:
Start doing
Stop doing
Continue doing
This is just one of many ways to do a sprint retrospective.
© 2003–2009 Mountain Goat Software®

Product backlog

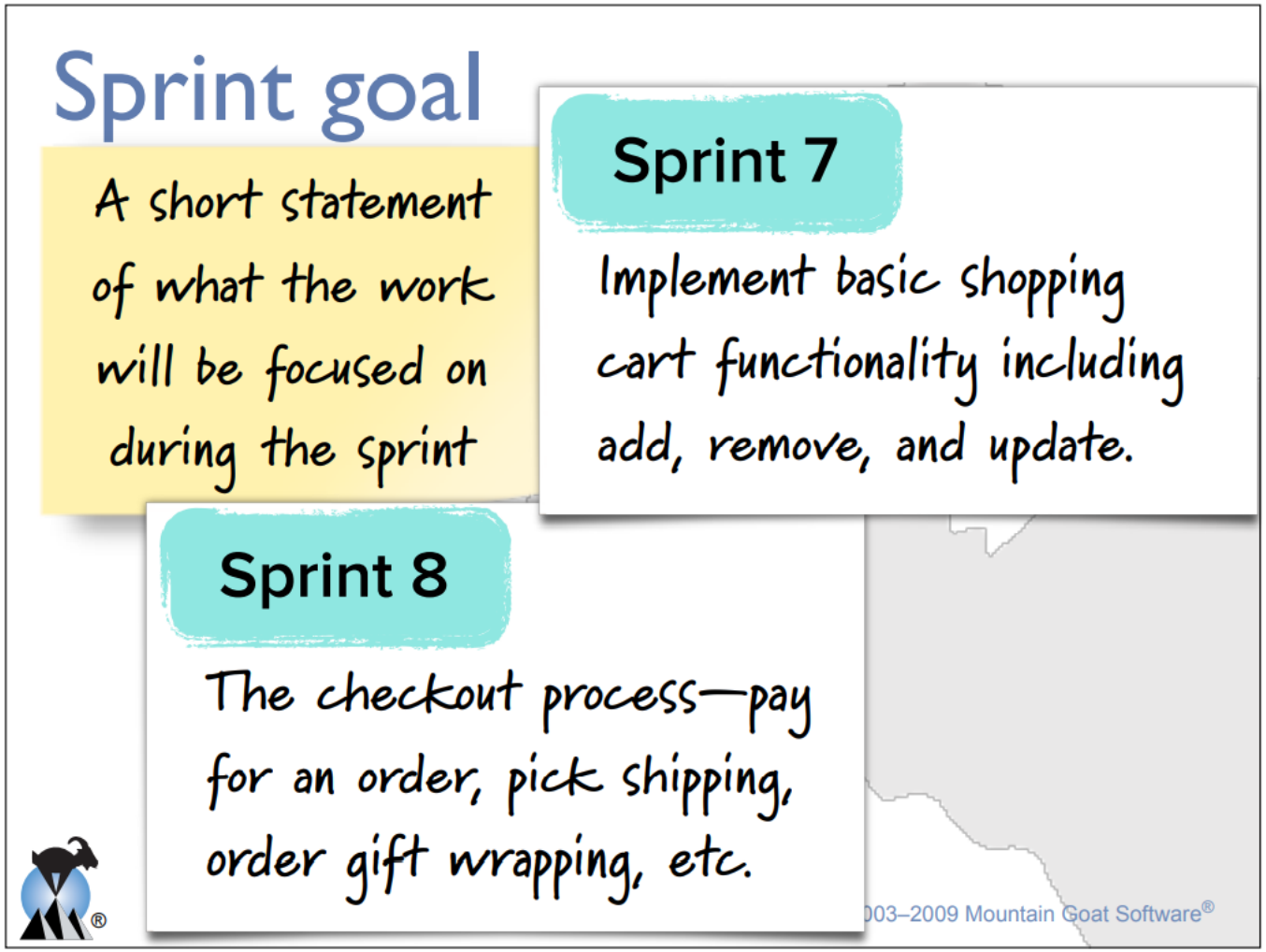
- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Prioritized by the product owner
- Reprioritized at the start of each sprint

This is the product backlog

A sample product backlog

Backlog item	Estimate
Allow a guest to make a reservation	3
As a guest, I want to cancel a reservation.	5
As a guest, I want to change the dates of a reservation.	3
As a hotel employee, I can run RevPAR reports (revenue-per-available-room)	8
Improve exception handling	8
...	30
...	50

Sprint Goal


Sprint goal
A short statement of what the work will be focused on during the sprint
Sprint 7
Implement basic shopping cart functionality including add, remove, and update.
Sprint 8
The checkout process—pay for an order, pick shipping, order gift wrapping, etc.
003–2009 Mountain Goat Software®

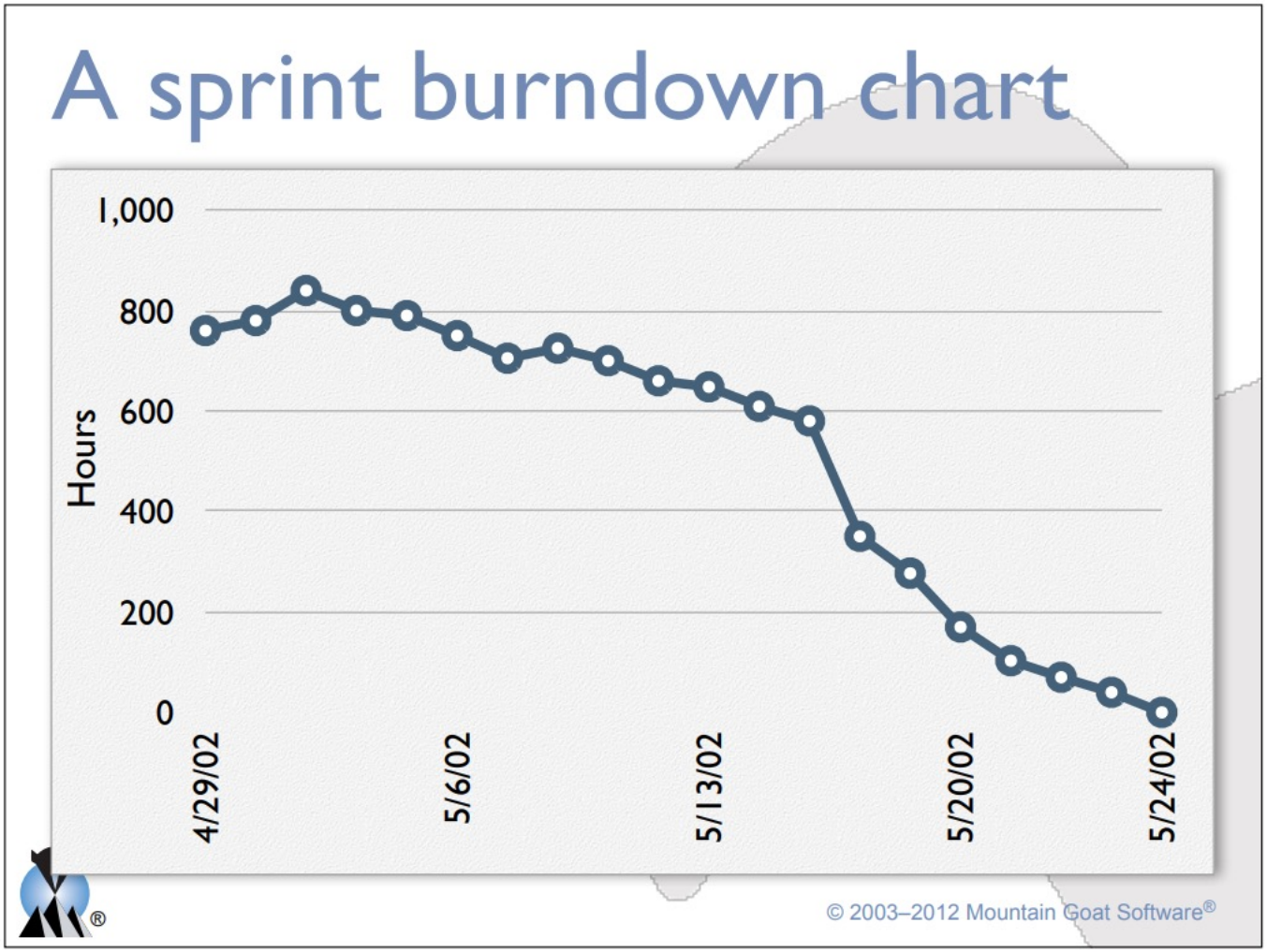
Managing the sprint backlog

- Individuals sign up for work of their own choosing
  - Work is never assigned
- Estimated work remaining is updated daily
- Any team member can add, delete or change the sprint backlog
- Work for the sprint emerges
- If work is unclear, define a sprint backlog item with a larger amount of time and break it down later
- Update work remaining as more becomes known

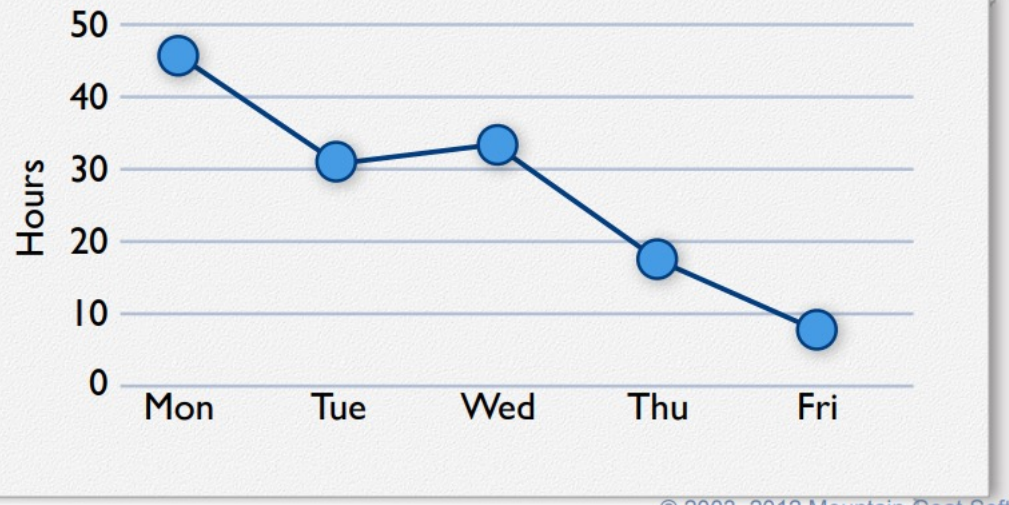
A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

A Sprint Burndown Chart



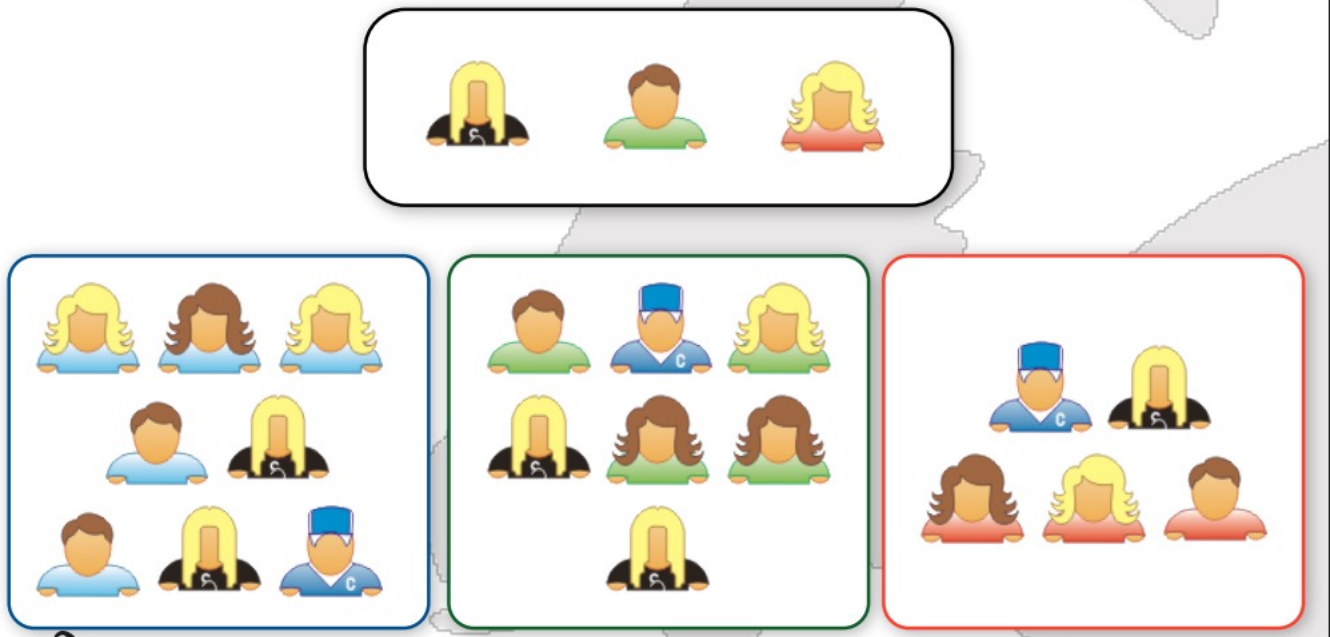
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				

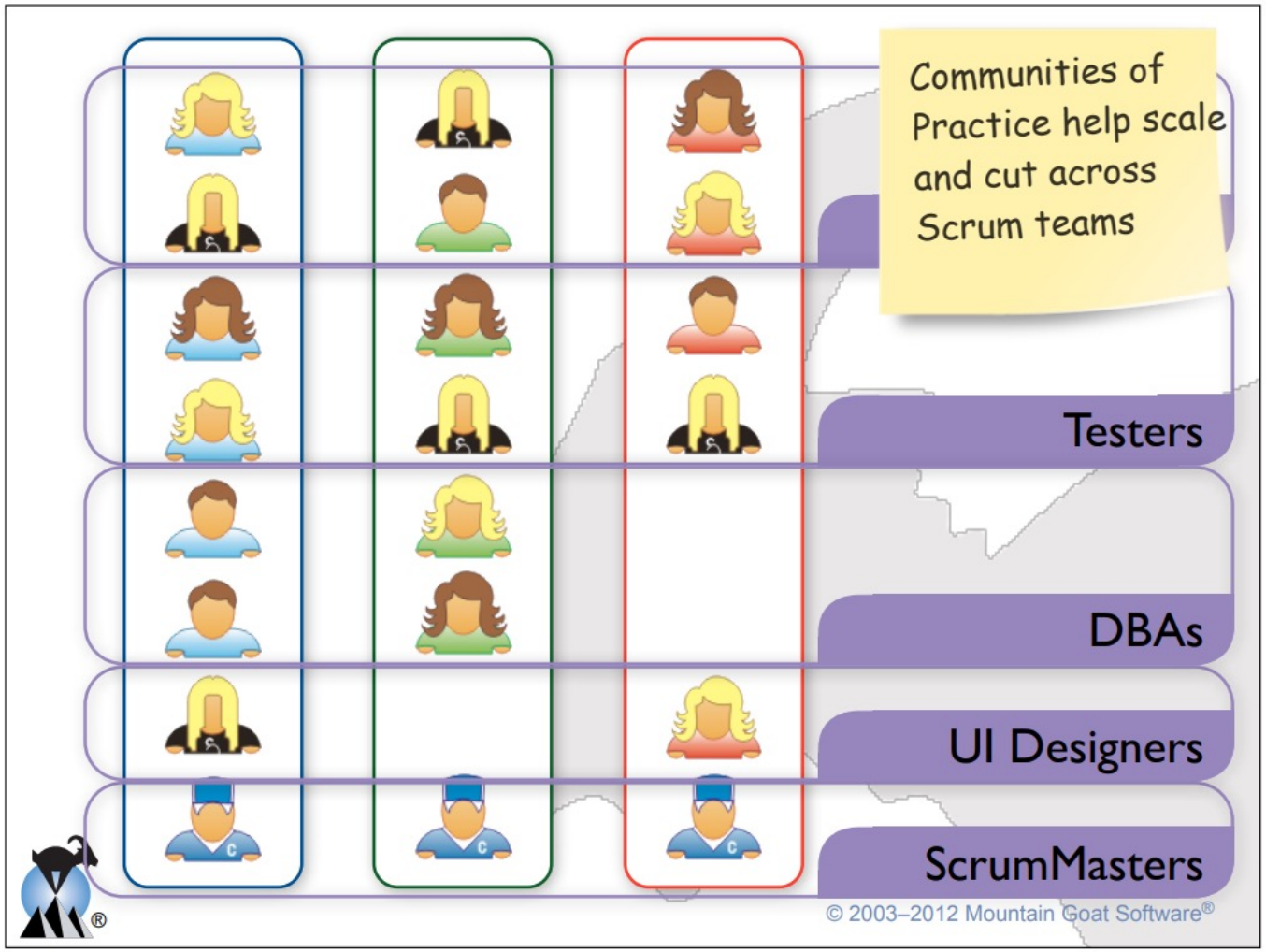


Scalability

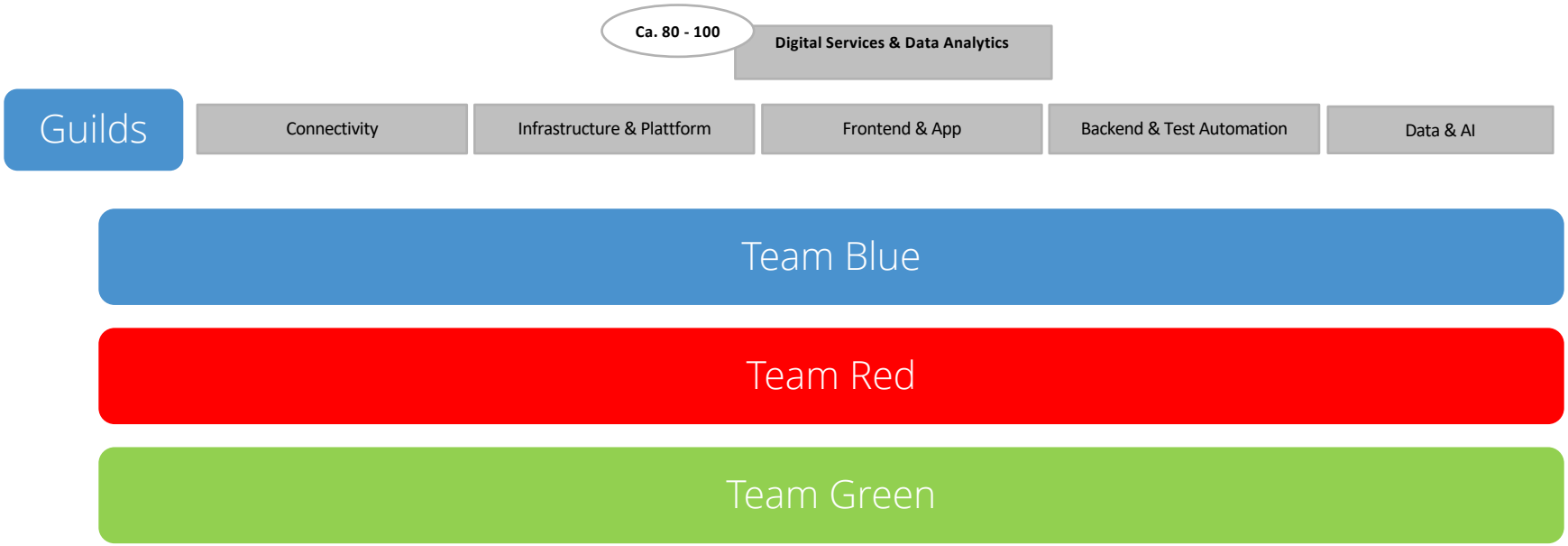
- Typical individual team is 7 ± 2 people
  - Scalability comes from teams of teams
- Factors in scaling
  - Type of application
  - Team size
  - Team dispersion
  - Project duration
- Scrum has been used on projects of over 1,000 people

Scaling through the Scrum of scrums

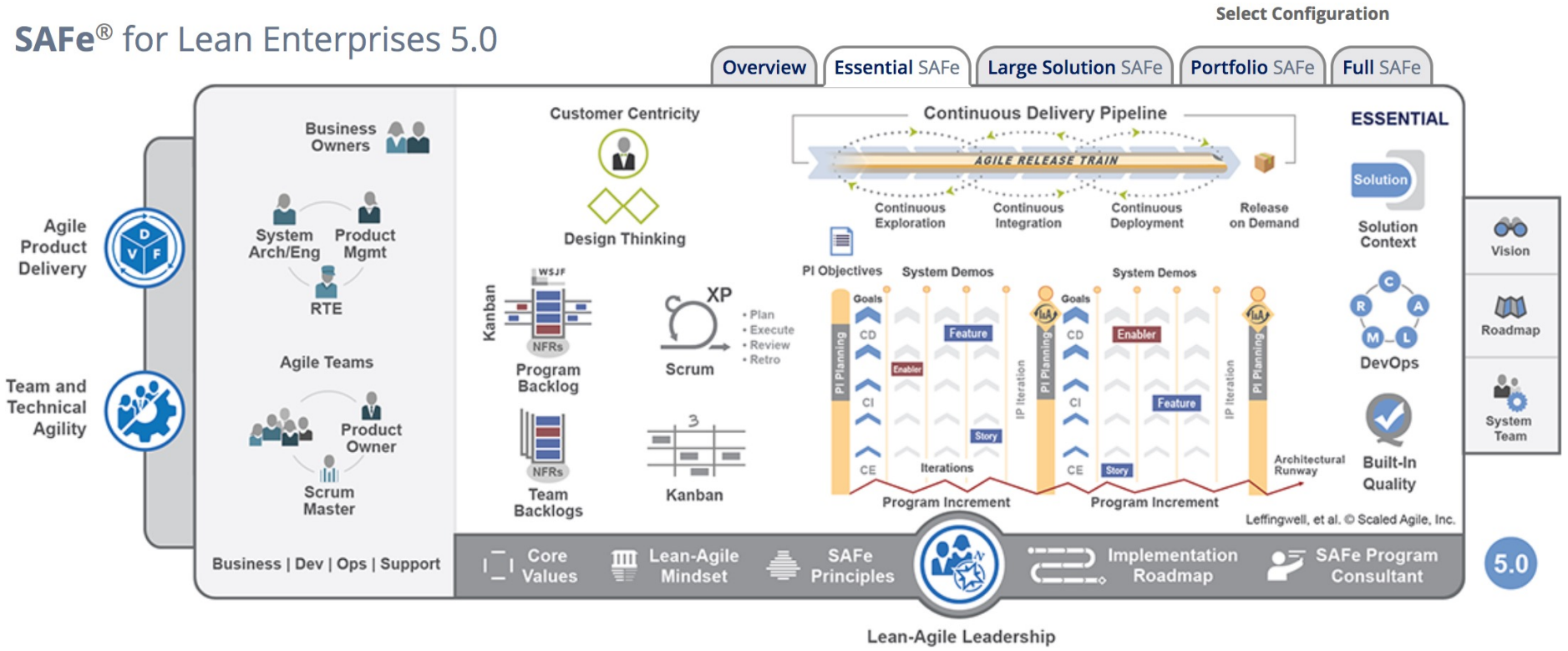



Communities of Practice help scale and cut across Scrum teams
Testers
DBAs
UI Designers
ScrumMasters
© 2003–2012 Mountain Goat Software®

Cross-functional Teams

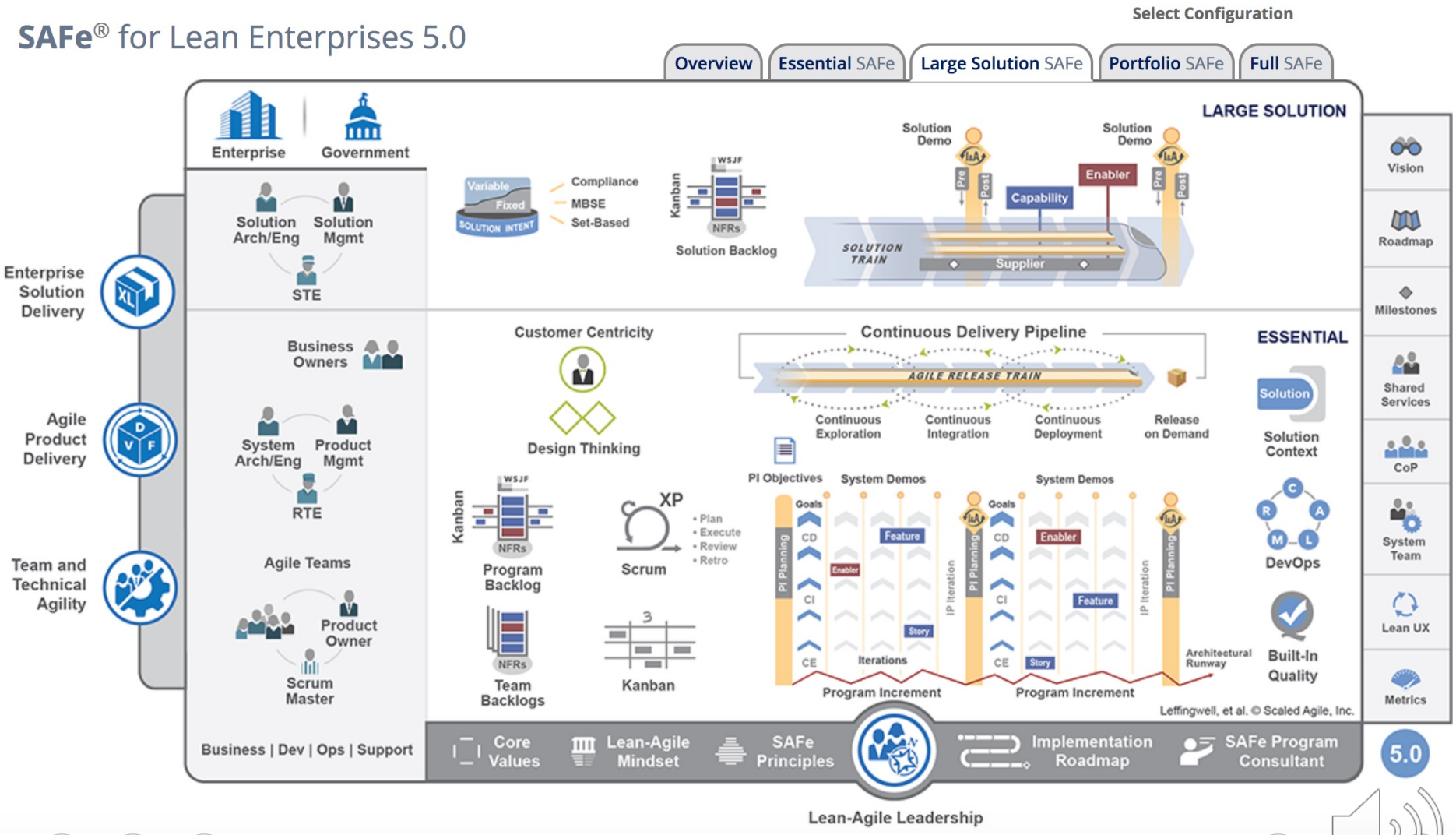


SAFeScrum – Essential SAFe

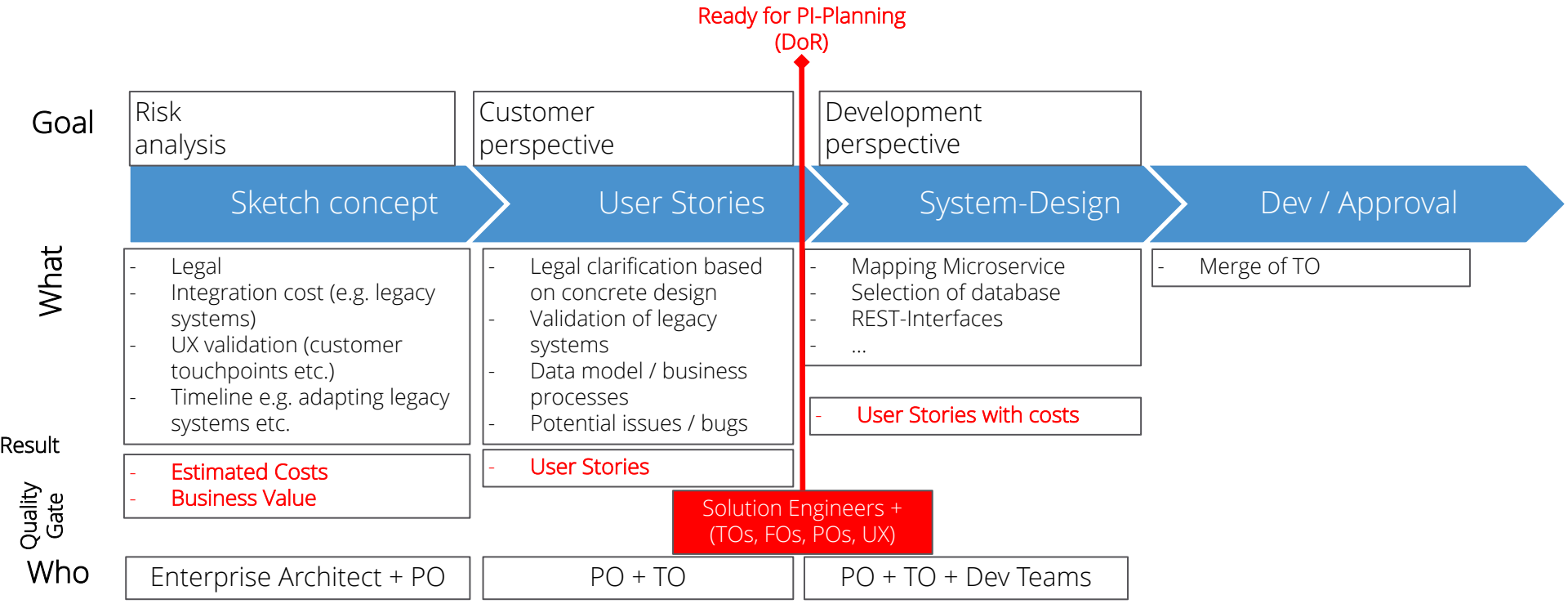


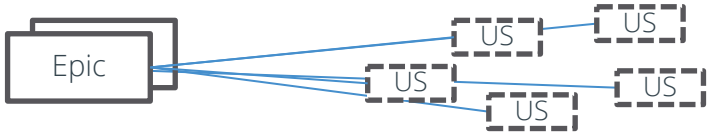
https://www.scaledagileframework.com/?_ga=2.201000878.1829074188.1581599312-457139379.1561986684#

SAFeScrum – Large Solution



Concrete Implementation of SAFEScrum






```
function foo {
}
```


SPRINT 4
SPRINT 3
We make
Risiko
Green
Red
Yellow
Grey
Magenta

Agenda

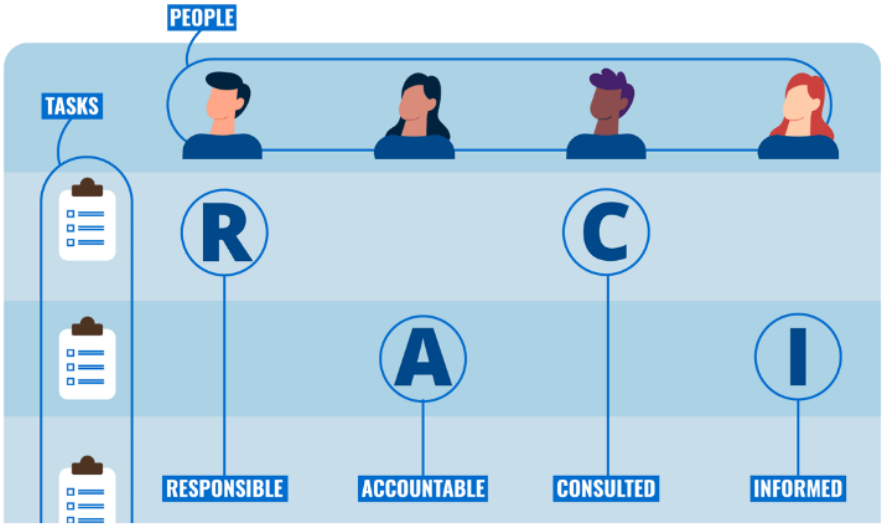
01 Mindset & People

02 Agility & Scaling

03 **Responsibility**

04 Summary

RACI



Responsible

Responsible in the RACI chart implies anyone who must complete a task or make a decision. In other words, anyone responsible for getting something done is assigned with an R. Using this roles and responsibilities matrix, several people who might be responsible for various tasks and decisions can be defined in the form of a chart.

Accountable

People who are deemed accountable in the RACI matrix include individuals who must sign off or approve tasks, decisions, and objectives. It is worth mentioning here that only one person can be accountable for a task and deliverable. This is important to remember to correctly use the RACI matrix.

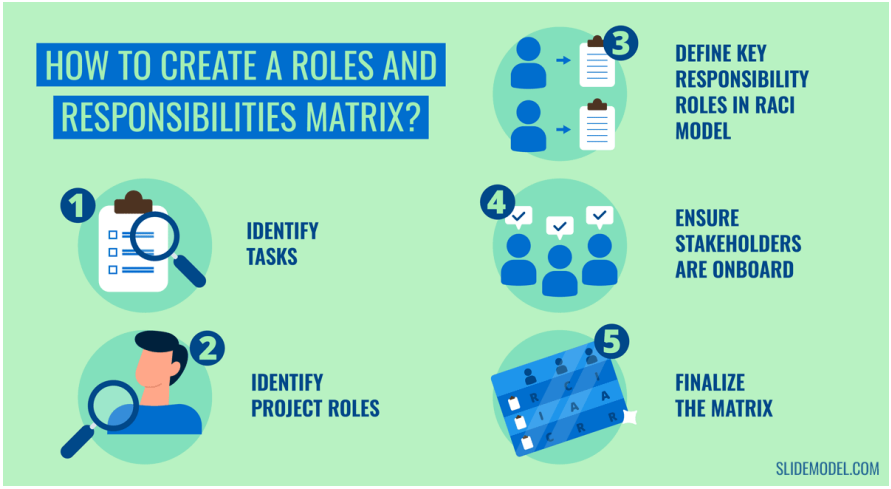
Consulted

Consulted includes individuals who are subject matter experts or whose opinion is considered for input before the work is signed off. These active participants are kept in loop so that they can provide timely input.

Informed

People who might be required to receive updates on the project or progress of executable tasks fall in the informed section on the RACI Chart. These people don't directly contribute to the task or decisions.

How to Create a RACI Matrix?



Identify Tasks

The first step in creating the RACI matrix is to identify all the tasks involved in a project and to list them. These tasks need to be listed on the left of the RACI chart in proper order of execution.

Identify Project Roles

In the next step, identify all the stakeholders and list them at the top of the RACI Chart. This will be the list of people who will have different roles in the execution of the project.

Define Key Responsibility Roles in RACI Model

Once the tasks and stakeholders have been listed, it's time to define which individual will be responsible for what role in the matrix. To do this, you will require filling in the remaining cells in the RACI chart with who is responsible, accountable, consulted, and informed.

Ensure Stakeholders are Onboard

While it might be simple enough to create a RACI matrix, ensuring that everyone is on board is equally important.

If people are given a responsibility, are held accountable, need to be consulted or informed, it is also important that they know and agree with their roles in the RACI matrix. If people aren't on board or lack commitment towards their defined role, this can lead to a major problem in ensuring project success or the proper execution of a business process. This is also the step where you should look to resolve ambiguities and possible conflicts to ensure everyone is on board.

Finalize the Matrix

It is quite possible that parts of the RACI Chart might need to be reorganized after consultation with stakeholders. Regardless of whether the matrix remains the same or is altered, you can finalize it once the stakeholders have been consulted, and any possible conflicts or ambiguities have been resolved.

Ensure There Aren't Too Many or Too Few Rs in the Chart

There are employees that are really good at their job and then there are individuals who might be less efficient or might not have a great reputation in getting things done. This also means that the people deemed most reliable can end up with more than fair share of work. However, efficient individuals should not have to make up for the inefficiencies of others. Instead, it's important to ensure that the team is robust enough to get things done. This is why it is important that there aren't too many Rs (responsibilities) assigned to a few people in the RACI Chart.

Ensure Every Person Accountable has one A

Everyone who is accountable can only be assigned a single A for a task. As mentioned earlier, it is important to have only one accountable person per task and deliverable.

Avoid Too Many Cs

Are there too many people being consulted? Could this slow down the execution of tasks? These are important considerations when constructing or finalizing the RACI matrix. You can look at the Rs and Cs to see if some of them need to be swapped for the smooth execution of tasks.

Ensure There are No Empty Cells

It is important to ensure there are no empty cells in the RACI Chart and all required sections are properly covered. It is also essential to ensure that cells aren't hastily filled in just to complete the matrix, but are carefully filled out to make the matrix effective.

Ensure Stakeholders are Properly Included in the RACI Chart

When writing down the stakeholder at the top of the roles and responsibilities matrix, make sure everyone who needs to be in the matrix is covered. Afterthoughts of including people who were not deemed stakeholders aren't feasible.

Pros

RACI Matrix can Provide Structure and Clarity for Projects

RACI matrix can be useful for providing a proper structure for projects. It can enable mapping all the tasks, stakeholders, and provide clarity regarding the assigned roles.

Helps Identify Roles and Responsibilities

RACI Chart can help remove ambiguities in the roles, responsibilities, accountability, and execution of projects.

Can Help Improve Accountability

When roles aren't properly defined, there can be ambiguities and question marks regarding the commitment of the people involved. Placing responsibilities in a well-defined box and ensuring that someone is accountable for deliverables can help make things more transparent and enable placing the blame in case of failure of timely deliverables.

The Right People can be Consulted According to Need

A RACI matrix can help identify if there are too many Rs or Cs to ensure there aren't too many people consulted or given extra responsibilities. By balancing responsibilities and the people who are consulted, the right people can be consulted according to need and unnecessary consultation steps which might result in the slow down of work can be removed.

Useful for Cross-functional or Departmental Projects and Processes

RACI Chart is useful for cross-functional or departmental projects and processes. Its linear structure helps easily chalk out the people responsible and accountable for tasks and deliverables while ensuring amalgamation with cross-functional and departmental structures.

Cons

RACI Does not Reflect Specific Obligations and Responsibilities

A RACI matrix can be more of a broad overview of the way roles are defined for a project and can lack specific information associated with the obligations and responsibilities that each individual must perform. The matrix might not be effective to properly cover the roles of specific employees or contractors.

It Might be Hard to Balance Rs, As, and Cs

Balancing the responsibilities, accountability, and who is consulted can be tricky. Sometimes, there might be a need to overlap tasks that the matrix does not allow. For example, small businesses might need to hold one person accountable for multiple tasks due to a lack of resources and the vast skill set of the individual. Balancing the Rs, As, and Cs can therefore become hard, especially when an organization is pressed for resources.

Suitable Mainly for Large Projects

A RACI matrix is suitable for large projects and can be hard to use for small projects or organizations where multitasking and changing hats is inevitable due to the small workforce and lack of funds available to run operations.

Might not be Suitable for All Organizational Structures

Since RACI Chart is a linear matrix, it can be hard to adapt it for all organizational structures. Furthermore, some organizations might prefer fewer people to make decisions and micromanagement might even be a necessity due to the lack of an experienced workforce. Startups often suffer from this dilemma, where decision-making might rest with the founder or a handful of people, with careful management of employees to ensure the organization can survive on a shoestring budget in the short term.

A RACI matrix might also be hard to implement for certain types of organizational structures such as a network organizational structure.

RACI Matrix Can Slow Down a Project or Process

Despite using RACI Chart, ambiguities, workload inefficiencies, additional expenses, and a slowdown in the execution of projects or processes can occur. Before using RACI Chart, you need to ensure if it fits with your organization's structure and requirements.

Agenda

01 Mindset & People

02 Agility & Scaling

03 Responsibility

04 Summary

Summary

1. People have to recruited in a customer-centric way; corporations have to see them as their customer and have to sell the open position
2. Employees want to work in an environment of clarity and psychological safety
3. Agile methods are cross-functional and perform an over-planning in sprints to manage the risk
4. Clarity often lacks in terms of responsibility which might become a problem