

# Miho Itakura



**I am Miho Itakura. I run an agile training company in Japan. We have trained 4,000 people in agile training. My goal is to transform Japan to Agile.**

**Today, I will introduce "What is Agile? Why Agile is important" with examples from Japan.**

**My English is not good, so I will be assisted by Joe Justice and Tomoko, who I usually train with in class.**

私は板倉美帆と申します。日本で、アジャイルトレーニングの会社を運営しています。これまで4000人をアジャイルトレーニングしています。  
私の目標は日本をアジャイルへ変革することです。

今日は「アジャイルとは何か?アジャイルが重要な理由」について日本の事例を交えながら紹介していきます。

私は英語が得意ではありませんので、いつも一緒にクラスでトレーニングしているJoe Justice氏とTomoko氏のサポートを受けながら進めていきます。

よろしくお願いいたします。



## AGILE BUSINESS INSTITUTE

Drives Enterprise Speed and Innovation



# Agenda Kanban board

We will proceed with the contents of the Kanban board below.  
Pink cards are for the lecture. Yellow cards are for you to work as a group.

下記カンバンボードの内容で進めていきます。  
ピンクのカードは講義です。黄色のカードはみなさんにもワークをしていただきます。

## To DO

# No Estimate  
2

Why Agile is  
Important  
3

#No Budgets  
2

Agile Teams  
and Mob  
10

Lean Coffee  
5

Kaizen  
5

## Doing

What is Agile?  
3

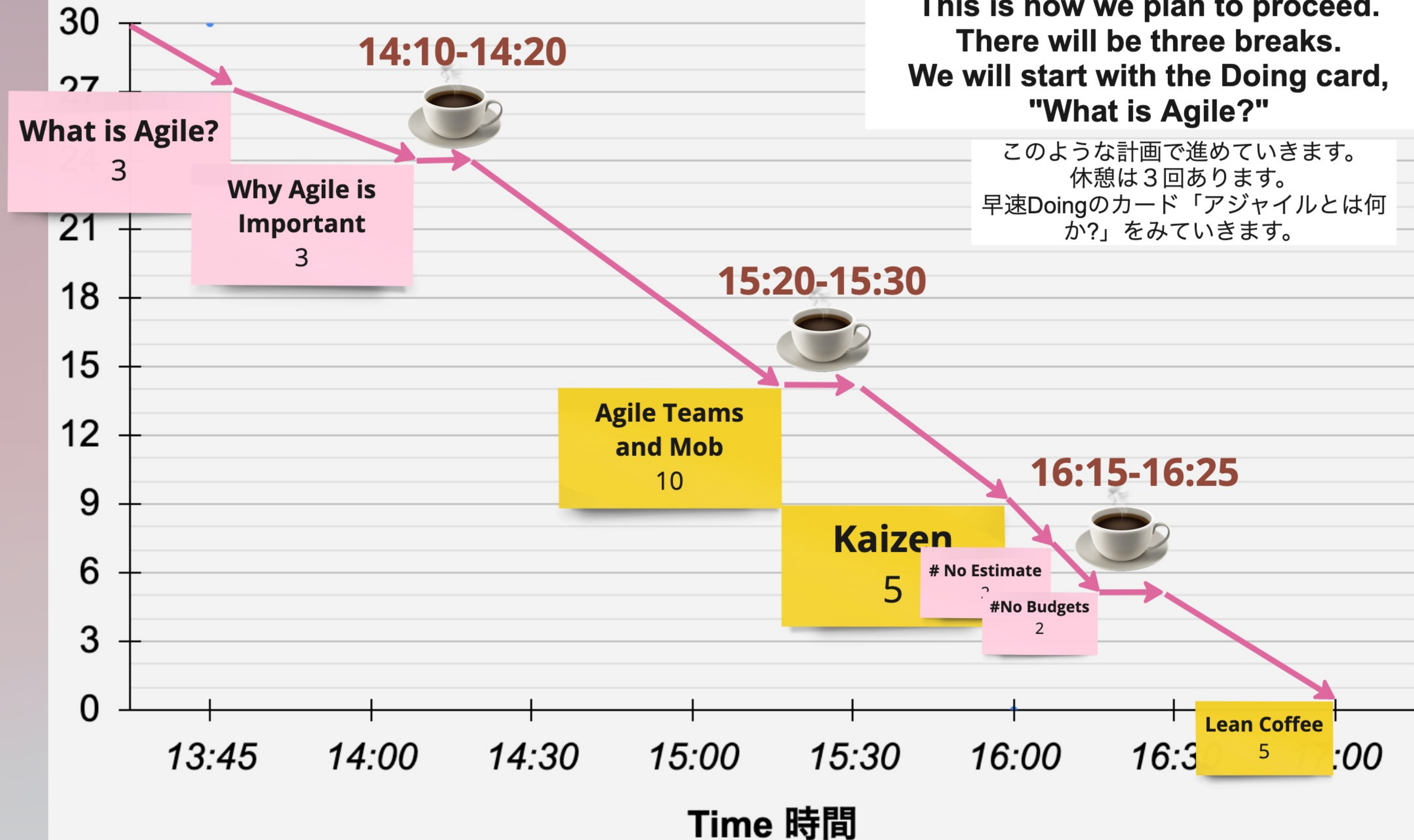
## Done



# Plan

This is how we plan to proceed.  
There will be three breaks.  
We will start with the Doing card,  
"What is Agile?"

このような計画で進めていきます。  
休憩は3回あります。  
早速Doingのカード「アジャイルとは何か？」をみていきます。

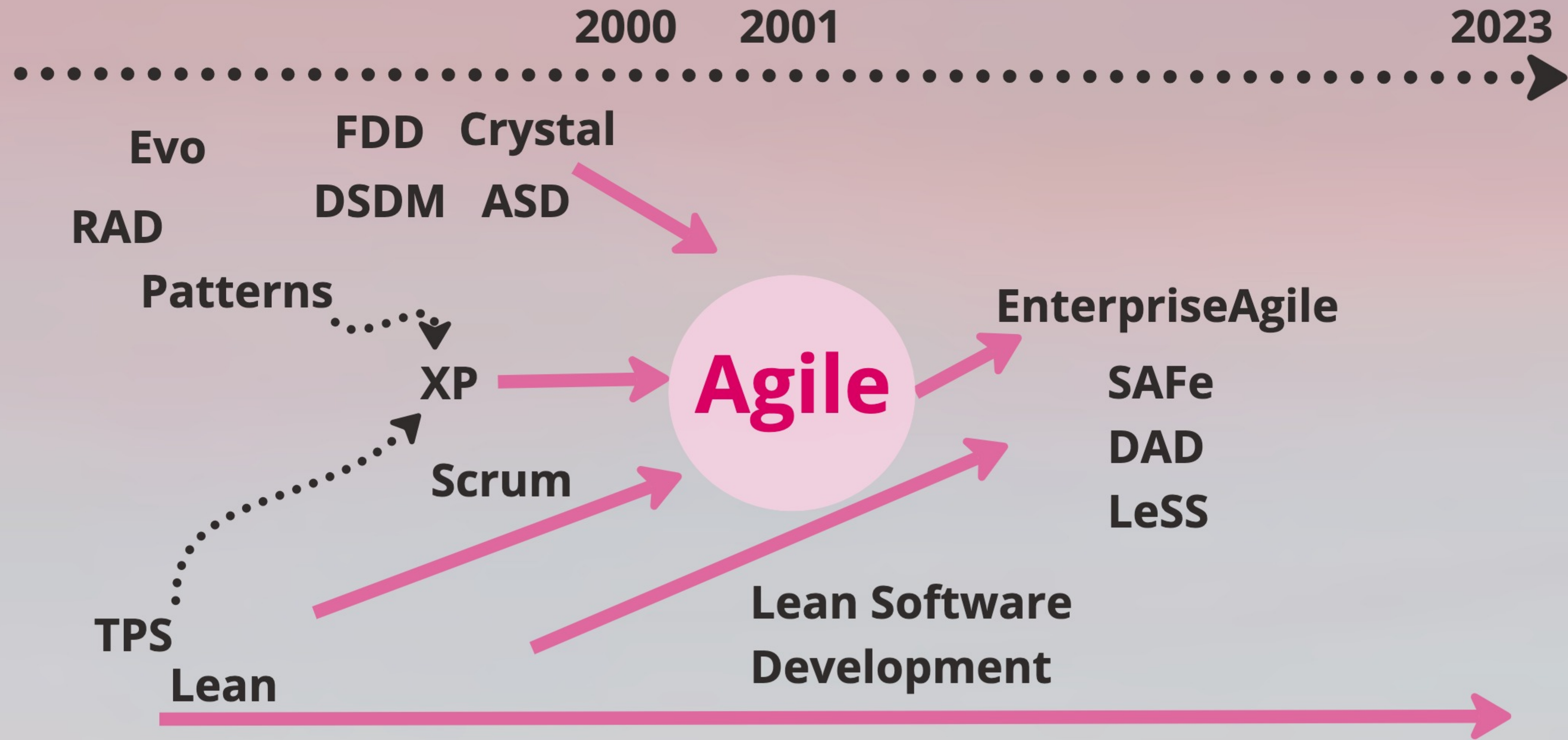




# What is Agile?

Agile was first announced in 2001 as the Agile Development Manifesto. It is a compilation of commonalities and principles that were brought together by the advocates of each of the previous software development methodologies and became "Agile". In other words, Agile is a generic term for multiple methodologies and frameworks, concepts, values, and mindsets.

アジャイルは、2001年に「アジャイル開発宣言」として発表されました。これは、それまでの各ソフトウェア開発手法を提唱した人々たちによって、共通点や原則がまとめられ「アジャイル」となりました。つまり、アジャイルは複数の手法やフレームワークの総称であり、概念や、価値観、マインドセットです。

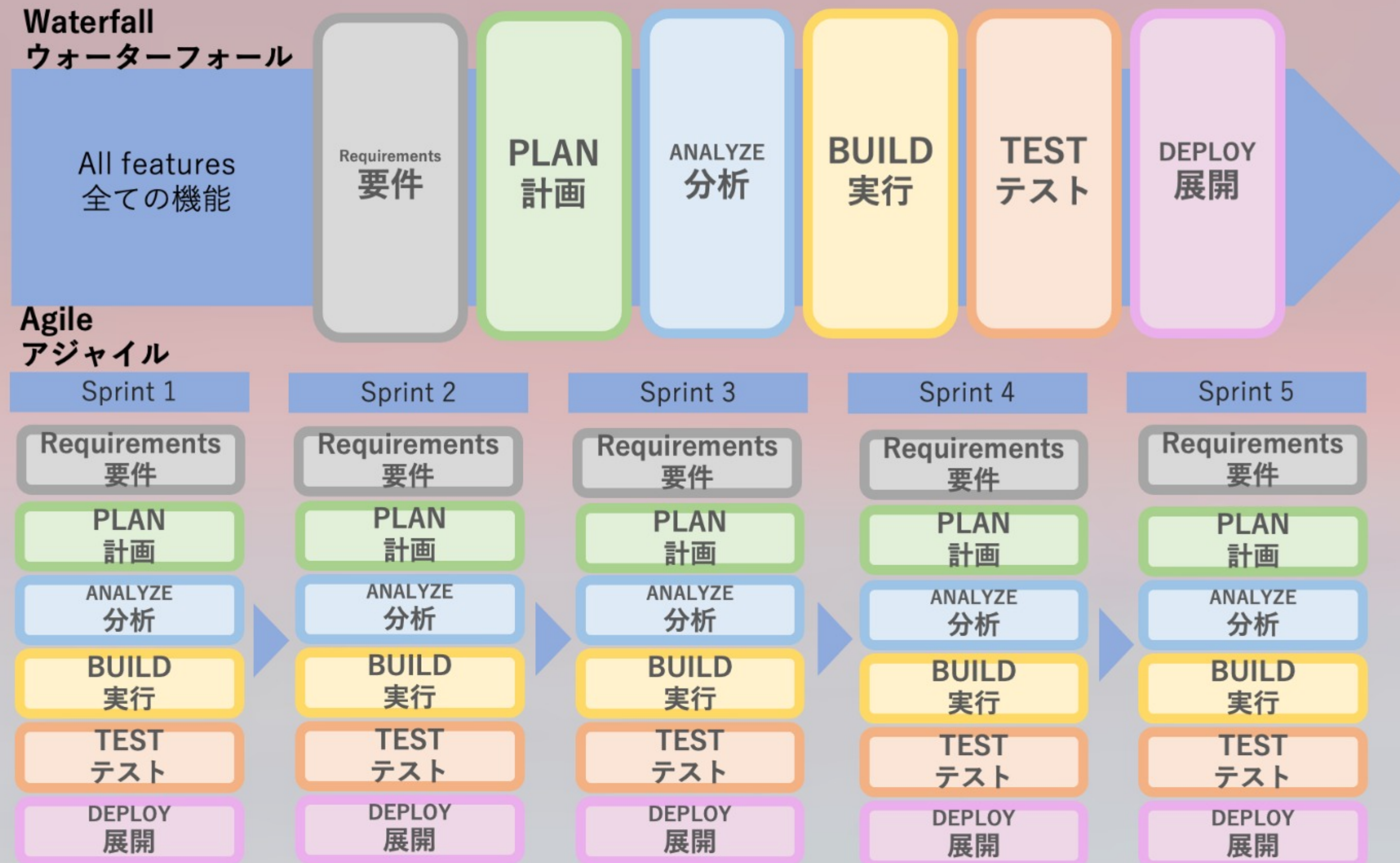




# Difference between Agile and Waterfall

Anticipate the future, plan carefully, and proceed based on needs that are considered highly necessary.

将来を予測し、綿密な計画をたて、必要性が高いと考えられるニーズに基づき進める



Respond flexibly to business changes by repeatedly making improvements in small units based on real-time needs and feedback.

リアルタイムなニーズやフィードバックを受けて、小さな単位で改善を繰り返し、ビジネスの変化に柔軟に対応する

Waterfall has a fixed schedule and budget, and requires careful planning in advance to anticipate the future. A large product is then completed in a single cycle.

Agile, on the other hand, divides a project into several functions, services, or phases of development.

Therefore, improvements are made repeatedly in small units in response to the real-time needs and feedback of customers, and the project flexibly responds to changes in the business. In other words, a large product is divided into smaller units and completed in multiple cycles in an agile manner.

ウォーターフォールは、スケジュールと予算が固定されており、将来を予測し、事前に綿密な計画を立てる必要があります。そして大きなプロダクトをひとつのサイクルで完成させます。

一方アジャイルでは、プロジェクトをいくつかの機能やサービス、または段階に分けて開発を進めていきます。そのため、顧客のリアルタイムなニーズやフィードバックを受けて、小さな単位で改善を繰り返し実行し、ビジネスの変化に柔軟に対応していきます。つまり、大きなプロダクトを、小さく分割し、複数のサイクルで俊敏(アジャイル)に完成させていきます。



# History of Agile and Waterfall

The waterfall development model originated in the manufacturing and construction industries. In highly structured physical environments, after-the-fact changes were considered to be very costly, even if not impossible. Then, a paper by Herbert D. Bennington in 1956 and Winston W. Reuss in 1970 described the use of phases for software development. This is often said to be the first software waterfall development model, but it is unclear who actually defined the term "waterfall" and when.

※The reason why people are moving away from waterfall development in recent years

- Once a step is completed, it is not possible to go back and make changes to the previous step.
- Waterfall methodology is highly dependent on initial requirements.
- The entire product is tested only at the end.
- The evolving needs of the client are not taken into account in the planning. If you want to make new changes created during the course of the project, the project will be delayed and the budget will be affected

As a solution to these shortcomings of the waterfall development model, the agile development model was born. The Agile development model is flexible enough to accommodate business changes by completing the product through a phased approach rather than a sequential design process.

ウォーターフォール開発モデルは、製造業や建設業で生まれました。高度に構造化された物理的環境では、事後の変更は不可能ではないにしても、非常に大きなコストがかかると考えられていました。そして、1956年のハーバートD.ベニンソンと1970年のウィンストンW.ロイスの論文によって、ソフトウェア開発のためのフェーズ使用が説明されました。これがソフトウェアのウォーターフォール開発モデルの最初と言われることも多いですが、実際には誰がいつ「ウォーターフォール」と定義をしたのかは不明とされています。

※近年ウォーターフォール開発離れが進む理由

- ステップが完了すると、前の段階に戻って変更を加えることができない
- ウォーターフォール手法は、初期要件に大きく依存する
- プロダクト全体は最後にのみテストされる
- 計画では、クライアントの進化するニーズは考慮されない。プロジェクトの過程で生まれた新たな変更を加えたい場合は、プロジェクトは遅れ予算に影響を与える

このようなウォーターフォール開発モデルの欠点に対する解決策として、アジャイル開発モデルは2001年に生まれました。アジャイル開発モデルは、順次設計プロセスではなく、段階的なアプローチによってプロダクトを完成させることにより、ビジネスの変更に柔軟に対応します。



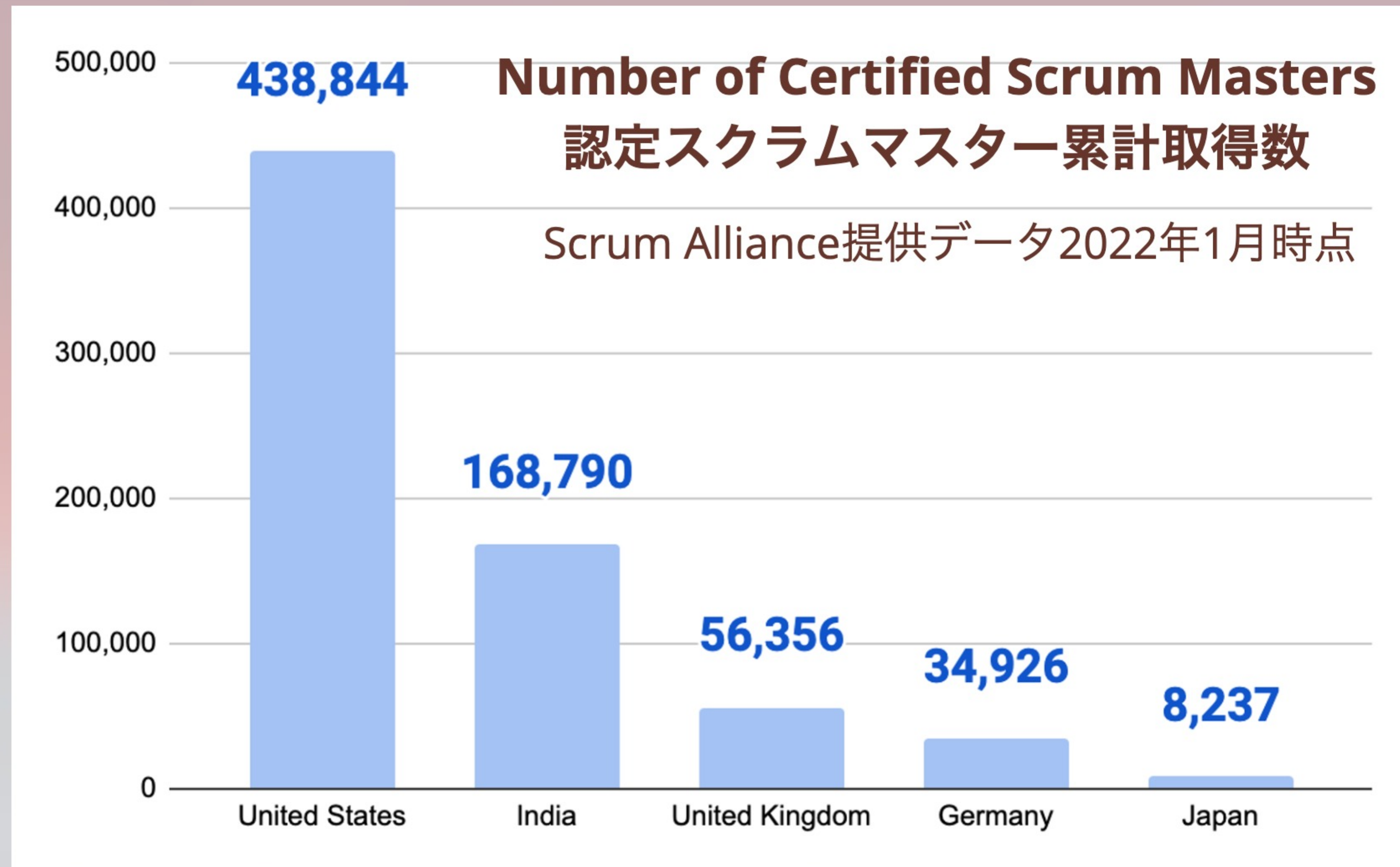


# Agile Expansion

This is data on the number of Scrum Master certifications by country.

Currently, the U.S. has the highest number of certified ScrumMasters with over 430,000. Germany has 34,000. Japan is still small compared to the U.S. and Germany.

これはスクラムマスターの資格取得者数の国別データです。現在アメリカが最も取得者数を多く43万人以上が取得しています。ドイツは3.4万人。日本は、アメリカやドイツと比較するとまだ少ないです。



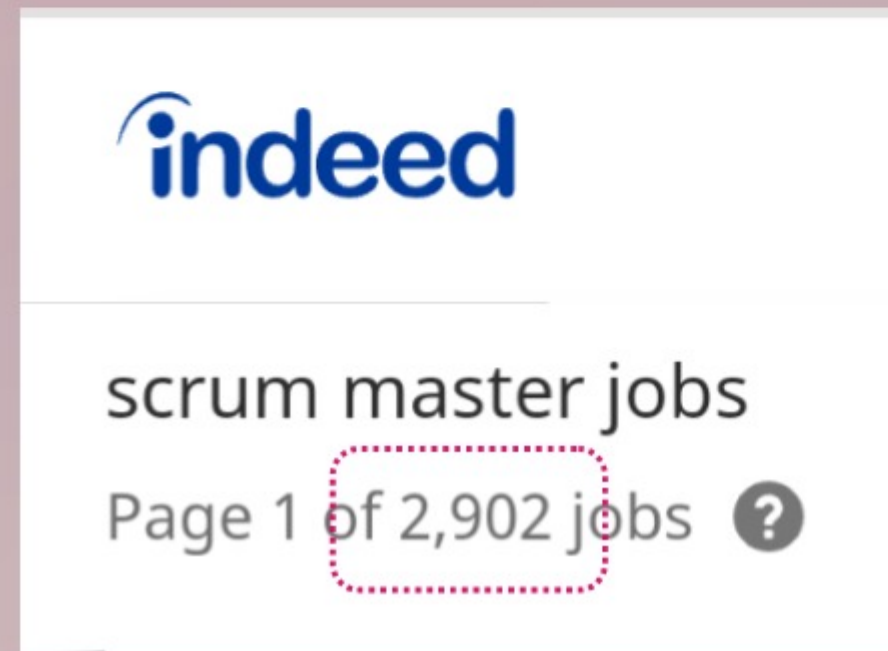


# Agile Expansion

## America

As of November 2021

**2,902 Jobs**



## Japan

As of November 2021

**493 Jobs**



Here is a look at the number of job openings.

The number of job openings for ScrumMasters in both the U.S. and Japan is growing.

In Japan, the number of jobs has grown more than 5 times in about a year and a half.

We also found 2,203 jobs in Germany.

こちらは求人数を調べたものです。

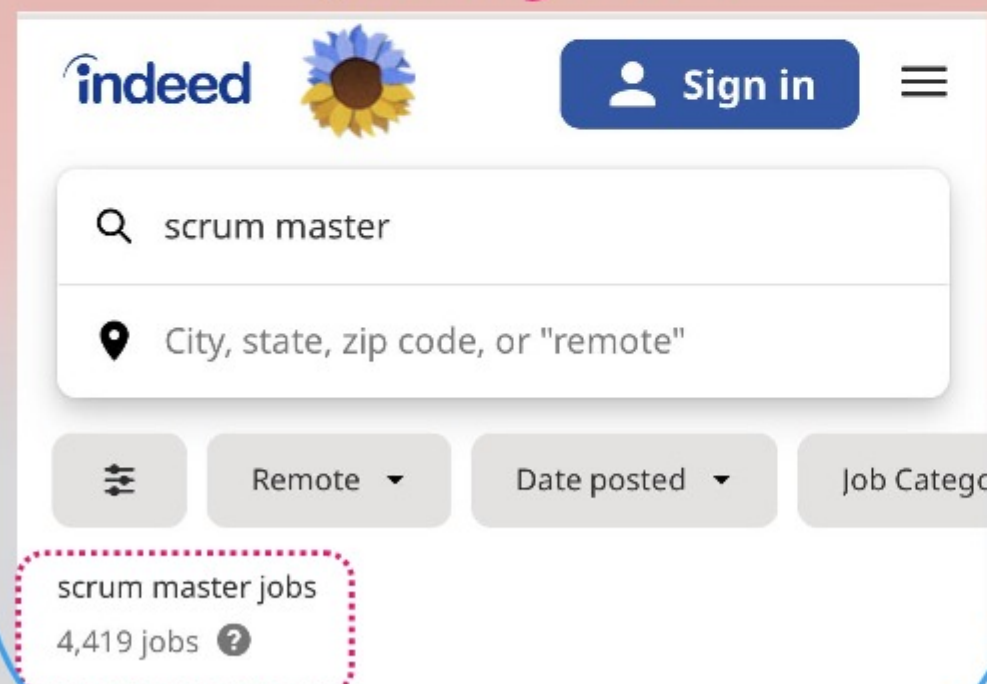
アメリカ、日本ともにスクラムマスターの求人数は伸張傾向にあります。日本は1年半程度で、5倍以上に求人数が伸長しています。

ドイツでも2,203件の求人が見つかりました。

## America

As of March 2023

**4,491 Jobs**



## Japan

As of March 2023

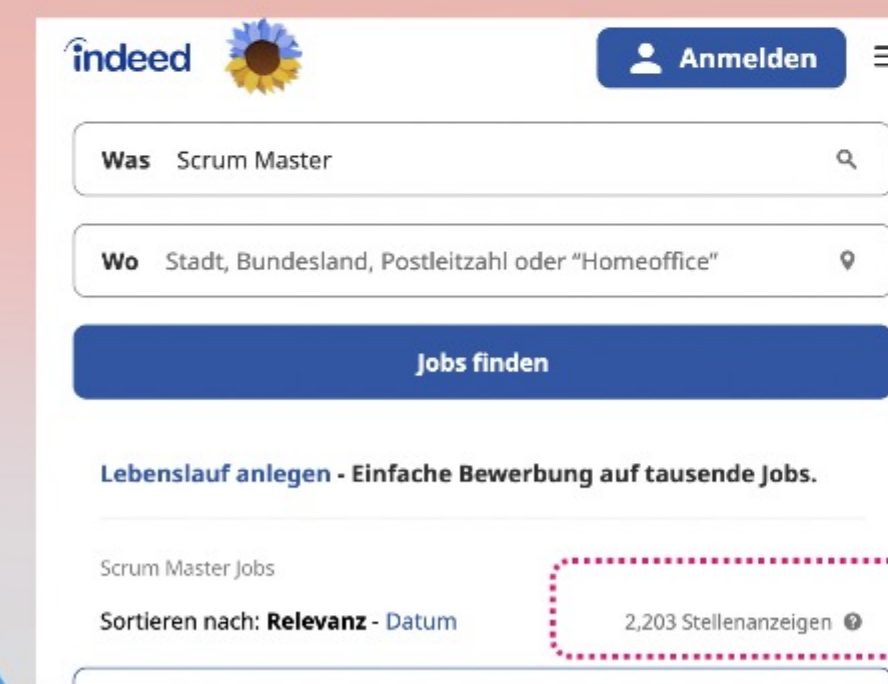
**2,731 Jobs**



## Germany

As of March 2023

**2,203 Jobs**





# Who are our clients

We have trained many companies in Japan to help them implement Agile.  
This year we are also preparing to provide training to some groups in the Japanese government.

私たちは日本で多くの企業のアジャイル導入支援のトレーニングを行なっています。  
今年は、日本政府の一部グループへのトレーニングも準備しています。

**Toyota Motor Corporation**  
**Honda Motor Co.**  
**NTT Group**  
**Softbank Corporation**  
**Googel**  
**Yahoo!**  
**LINE**  
**Rakuten Group**  
**Panasonic Group**  
**Toshiba Group**  
**Sony Group**  
**IBM Japan, etc.**

**トヨタ自動車**  
**本田技研工業**  
**NTTグループ**  
**ソフトバンク**  
**Googel**  
**Yahoo! Japan**  
**LINE**  
**楽天グループ**  
**パナソニックグループ**  
**東芝グループ**  
**ソニーグループ**  
**日本IBM など**



# What do they want from Agile?

## Gaining Speed and Flexibility

Companies must continue to deliver value in the face of growing uncertainty. The solution is to increase business speed. The agile approach is one way to achieve this.

## スピードの獲得と柔軟性

不確実性が拡大する状況でも、企業は価値を提供し続ける必要があります。その対策としてビジネススピードを上げること。その手段としてアジャイルアプローチがあります。

### Increased uncertainty

Unpredictable business environment and fluidization of competitive principles

### Situation 状況

### 不確実性の増大

予測不可能なビジネス環境と競争原理の流動化

### Gain overwhelming business speed

Visualization at high speed → Judgment at high speed  
→ Act at high speed

### Countermeasure 対策

### 圧倒的なビジネススピードの獲得

高速に見える化 → 高速に判断 → 高速に行動

### Agile approach

Small, fast, verify and improve repeatedly to gain speed

### Means 手段

### アジャイルアプローチ

小さく、早く、検証して改善を繰り返しスピードを獲得する

### Significance of existence of a company

What value do you provide to society and customers?

### Objective 目的

### 企業の存在意義

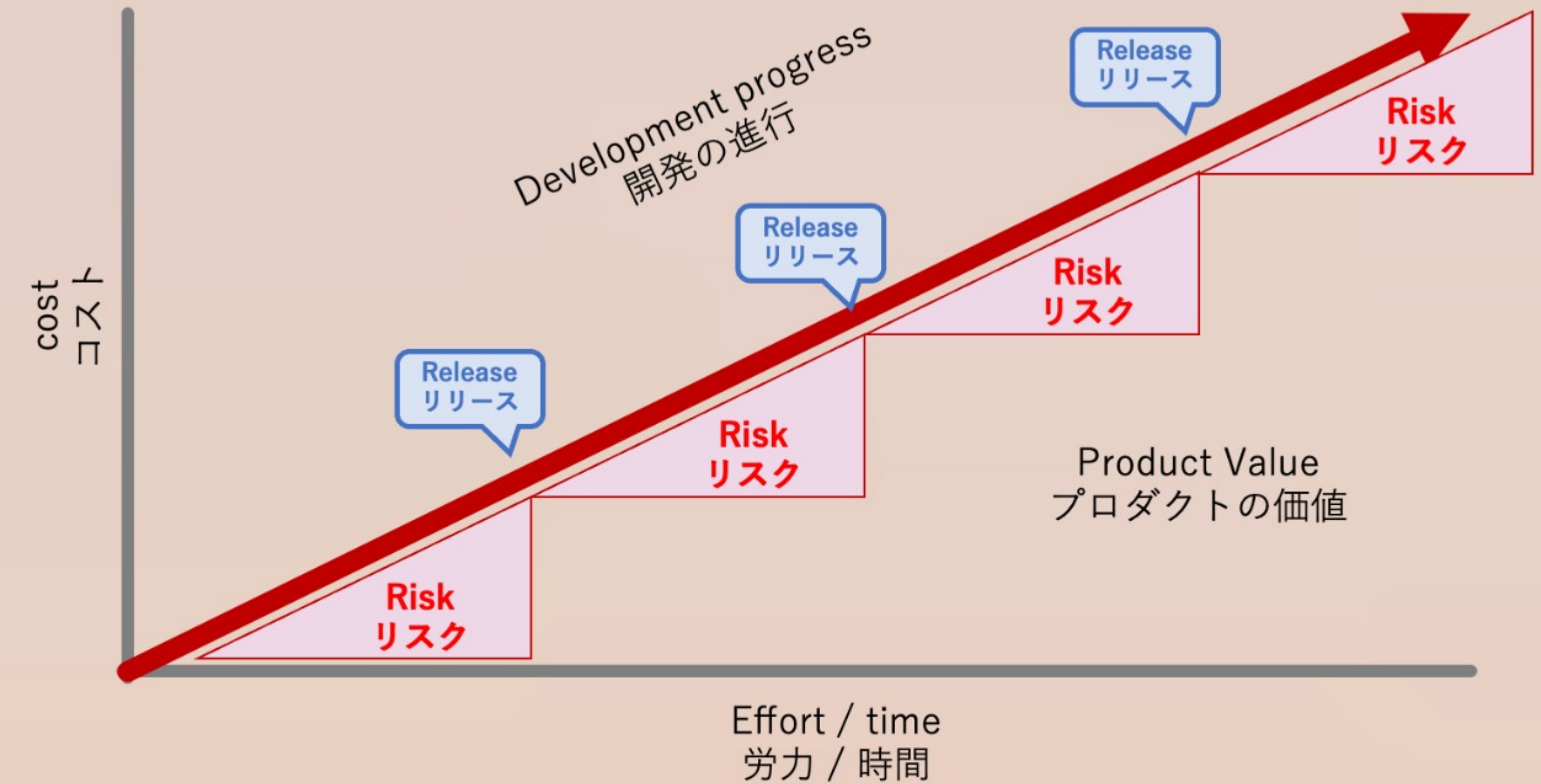
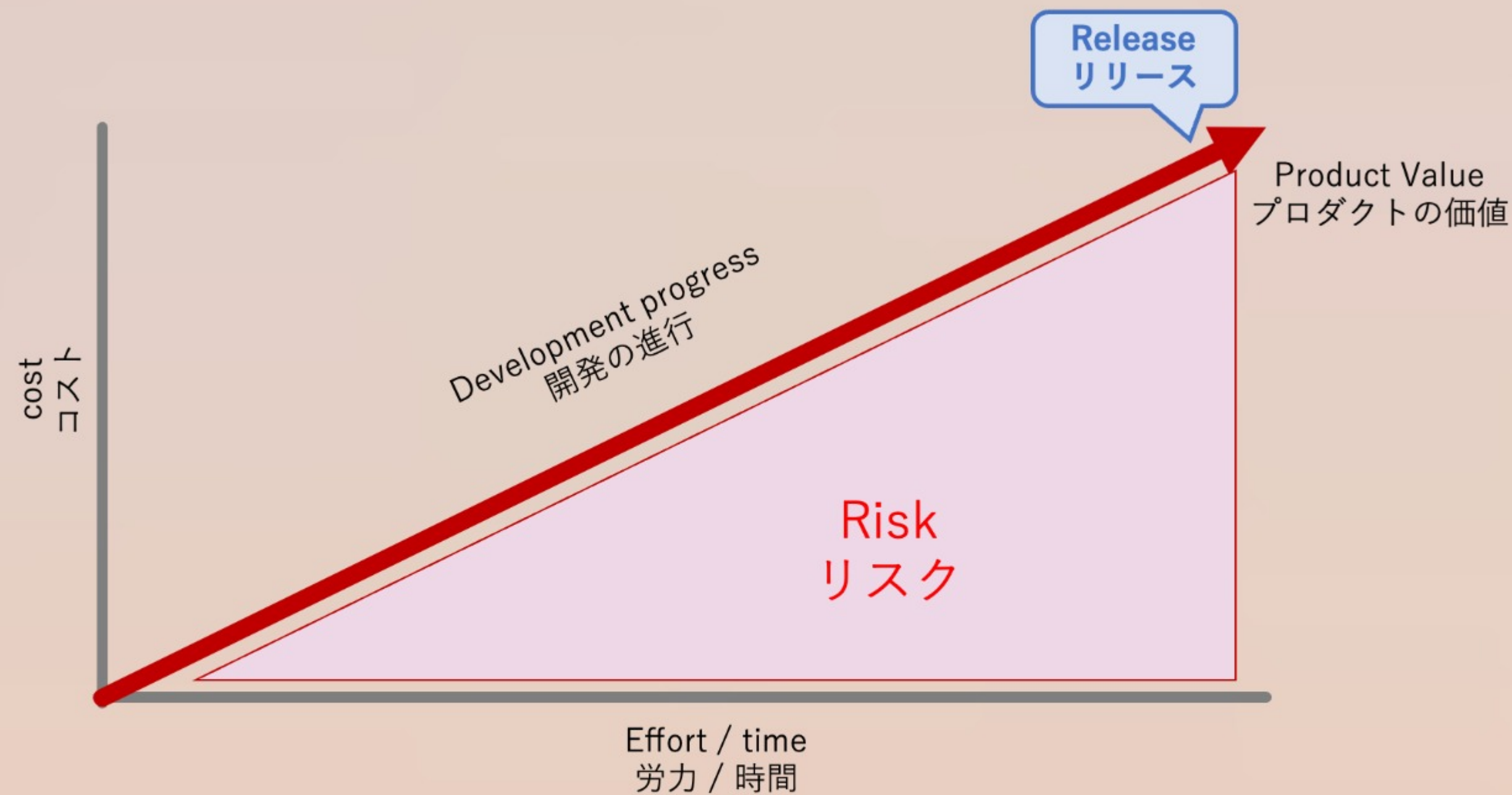
いかなる価値を社会や顧客に提供するのか？



# Agile reduces risk

**As business complexity increases, strategies must change. Scrum creates value by iteratively releasing and improving in small increments. This also minimizes risk.**

ビジネスの複雑さが増すにつれ、戦術も変化させなければなりません。スクラムは、小さな単位でリリース・改善を繰り返し価値を創出します。これによりリスクも最小限に抑えることが可能です。





# How can Agile be introduced into a company?

**This is a very common question. Unfortunately, there is more than one, as there are many different mindsets, bosses and customers, as well as corporate cultures. Sometimes it is a top-down approach, and sometimes it is a bottom-up approach to promoting Agile. Regardless of the method, it is important for people to understand Agile first.**

**That is why we are involved in agile training activities. I would like to show you some examples of what we do.**

アジャイルはどのように企業に導入すればいいのか？  
これは非常に多い質問です。  
残念ながら、様々な考え方の上司や顧客の考え方、また企業文化もあるので1つではありません。  
トップダウン型ともあれば、ボトムアップ型でアジャイルを推進していくこともあります。  
いずれの方法であれ、まずはアジャイルを理解してもらうことは重要です。

そのため、私たちはアジャイルトレーニングの活動しています。  
次からいくつかの例を紹介したいと思います。



# Let's look at a case study

Our trainer, Joe, was invited to join TRI-AD(\*1), Toyota's new agile center, as an agile coach. TRI-AD is responsible for the development of Toyota Group's automated driving technology, including development, implementation, market introduction and dissemination. All employees of the center and many suppliers attended the Scrum Master Training. Akio Toyoda, a former CEO, also came to the center to offer encouragement. In this way, the Toyota Group has established an Agile Group to promote the introduction of Agile. We also have an Agile training session scheduled for this June at Toyota's headquarters. Toyota is continually creating learning opportunities for their employees and trying to instill an Agile mindset.

※Toyota Research Institute Advanced Development will transition to a holding company structure with Woven Planet Holdings as the holding company from January 2021.

トヨタグループの自動運転技術の開発を担い、開発、実装、市場導入・普及を行う、新生トヨタのアジャイルセンターであるTRI-AD(\*1)のアジャイルコーチとして、私たちのトレーナージョーは招聘されました。同センターの全従業員と多くのサプライヤーが、スクラムマスタートレーニングを受講しました。当時の豊田章男CEOも激励に駆け付けています。このようにトヨタグループは、アジャイルグループを立ち上げ導入の推進をしています。

また私たちは今年の6月に、トヨタ自動車本社でのアジャイルトレーニングを予定しています。

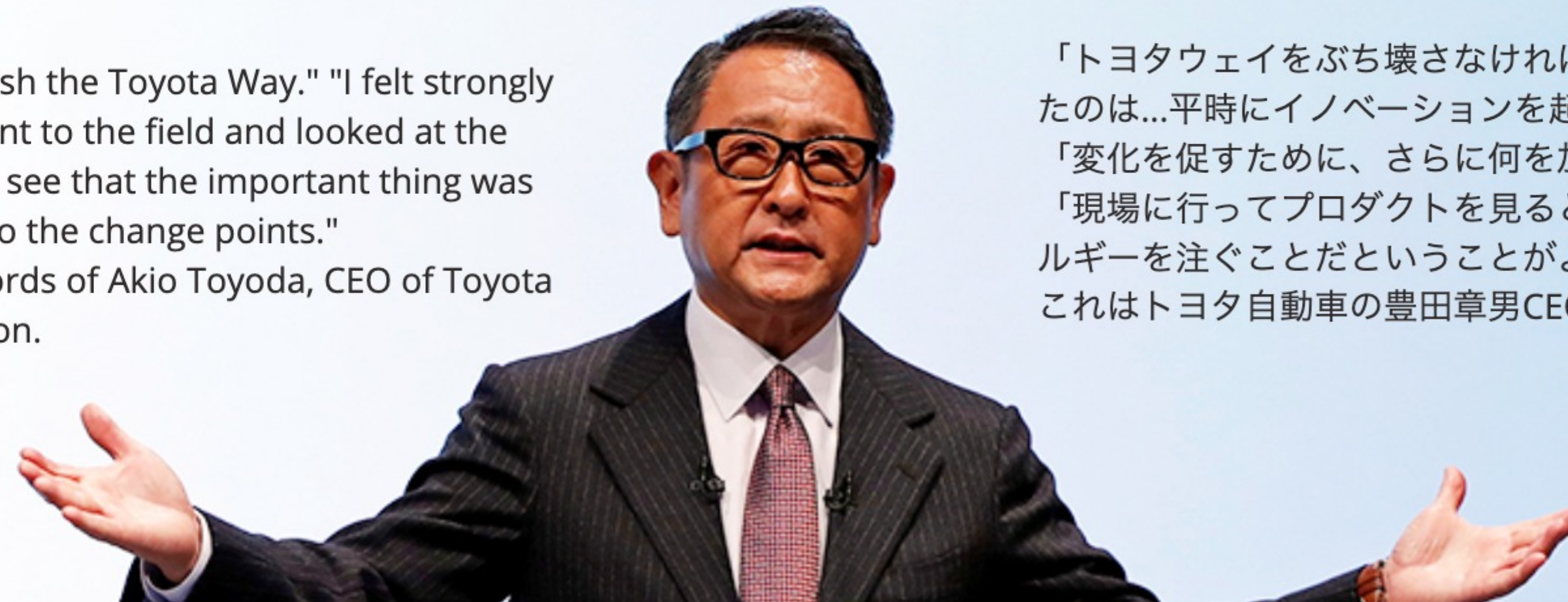
トヨタ自動車は継続的に社員の学習機会を設け、アジャイルのマインドセットを根付かせようとしています。

※トヨタ・リサーチ・インスティテュート・アドバンスド・デベロップメントは、2021年1月から「ウーブン・プラネット・ホールディングス」を持株会社とする体制に移行

# TOYOTA

"We have to smash the Toyota Way." "I felt strongly that... When I went to the field and looked at the products, I could see that the important thing was to put energy into the change points."

These are the words of Akio Toyoda, CEO of Toyota Motor Corporation.



「トヨタウェイをぶち壊さなければならない」「私が強く感じたのは...平時にイノベーションを起こすことの難しさだった」  
「変化を促すために、さらに何を加速すべきかが分かった」  
「現場に行ってプロダクトを見ると、重要なのは変化点にエネルギーを注ぐことだということがよく分かる」  
これはトヨタ自動車の豊田章男CEOの言葉です。

Image : [businessinsider](https://www.businessinsider.com)



# Let's look at a case study

At Sony Global Solutions, Inc., people who have taken our classes act as Agile coaches to promote Agile in their own companies.

First, they create a successful Agile case study with their own team, and then bring that case study to a neighboring team for a study session. Gradually, they expand the scale of the study sessions and continuously strive to promote Agile understanding within the company.

ソニーグローバルソリューションズ株式会社では、私たちのクラスを受講した方が、アジャイルコーチとして、彼ら自身の会社にもアジャイル推進を行っています。

まずは、自分自身のチームでアジャイルの成功事例を作り、その事例を持って隣のチームへ勉強会を開催。徐々に勉強会の規模を広げ継続的に社内のアジャイル理解促進に努めています。

**First, mentor their own teams and create best practices.**

まずは自身のチームを指導し成功事例をつくる

**They hold workshops with best practices.**

まずは自身のチームを指導し成功事例をつくる

**Expanding the size of the study group**

Expanding the size of the study group



**Sony Global  
Solutions Inc.**



# Let's look at a case study

Mercari Inc., which operates one of Japan's largest flea market apps, had a company-wide two-year plan to promote the introduction of agile.

Agile methods are used not only in software development, but also in the human resources department.

In the case of Mercari, Agile was promoted in a top-down manner, but it still took two years to implement.

It would be very difficult to change everything to Agile from tomorrow. Starting with what you can do now may be a good place to start.

日本最大級フリマアプリを運営する株式会社メルカリでは、全社で2年計画を立て、アジャイル導入推進を実施していました。

ソフトウェア開発だけに限らず、人事部門でもアジャイル手法が活用されています。

メルカリの場合は、トップダウン型でアジャイル推進をしていましたが、それでも2年かけて実施していました。

明日から全てをアジャイルに変えるというのは非常に難しいでしょう。まずはできるところから始めるというのが最初のスタートかもしれません。





# Executive Agile Transformation

We also offer classes to help executives of organizations understand Agile.

私たちは組織の上層部にアジャイルを理解してもらうためのクラスも実施しています。

ENHANCING ORGANIZATIONAL CAPABILITY

# EAT

EXECUTIVE AGILE TRANSFORMATION

Those who are in a position to receive proposals and consultation on Agile implementation from Scrum Masters and Product Owners.

Those who are in a position of outsourcing development to a client.

Those in upper management who are responsible for development in their organizations.

What we offer in the curriculum is "knowledge as a common language" that bridges the gap between members and executives or clients.

**It's essential !**

EAT FACILITATES AGILE KNOWLEDGE



# Let's watch the actual Agile Team!



**Now please watch a video of an actual fast agile team!**  
**This is an excellent agile team working in a department store called Nordstrom.**  
**The video is about 6 minutes.**

実際のアジャイルチームの動画を見てみましょう！  
ノードストロームというデパートで働く優れたアジャイルチームです。  
映像は訳6分程度です。

<https://youtu.be/hNy2h6GG4ng>

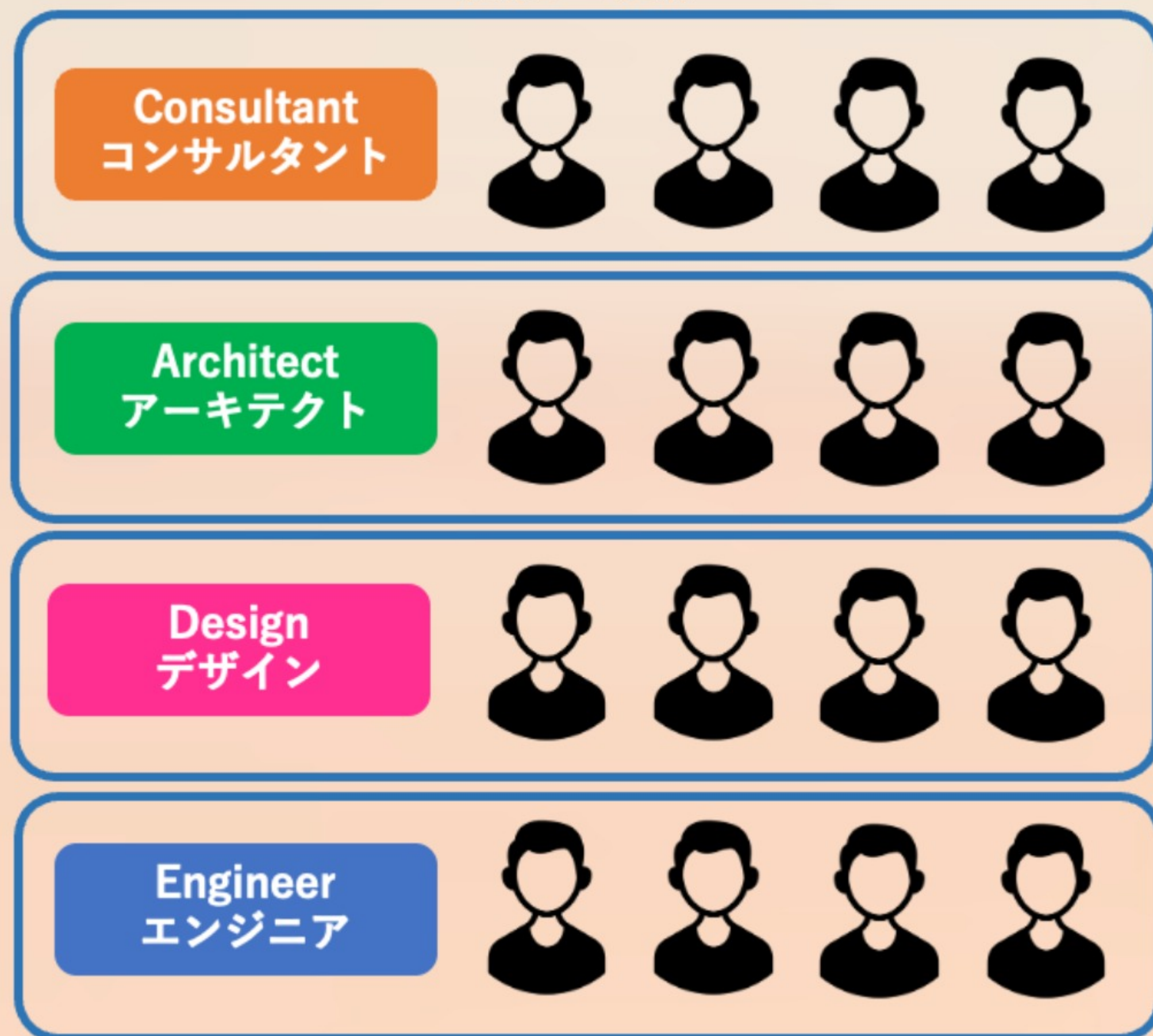


# Agile Team Structure

The team seen in the video was the cross functional team.  
A cross-functional team is one in which the team has the skills needed to release the product. In addition, members will aim to be full-stack individuals (so-called T-skills) who can take on multiple roles. Cross-functional teams also eliminate (or shorten) the need for coordination between departments. As a result, it promotes speed and innovation.

## Vertical Organization

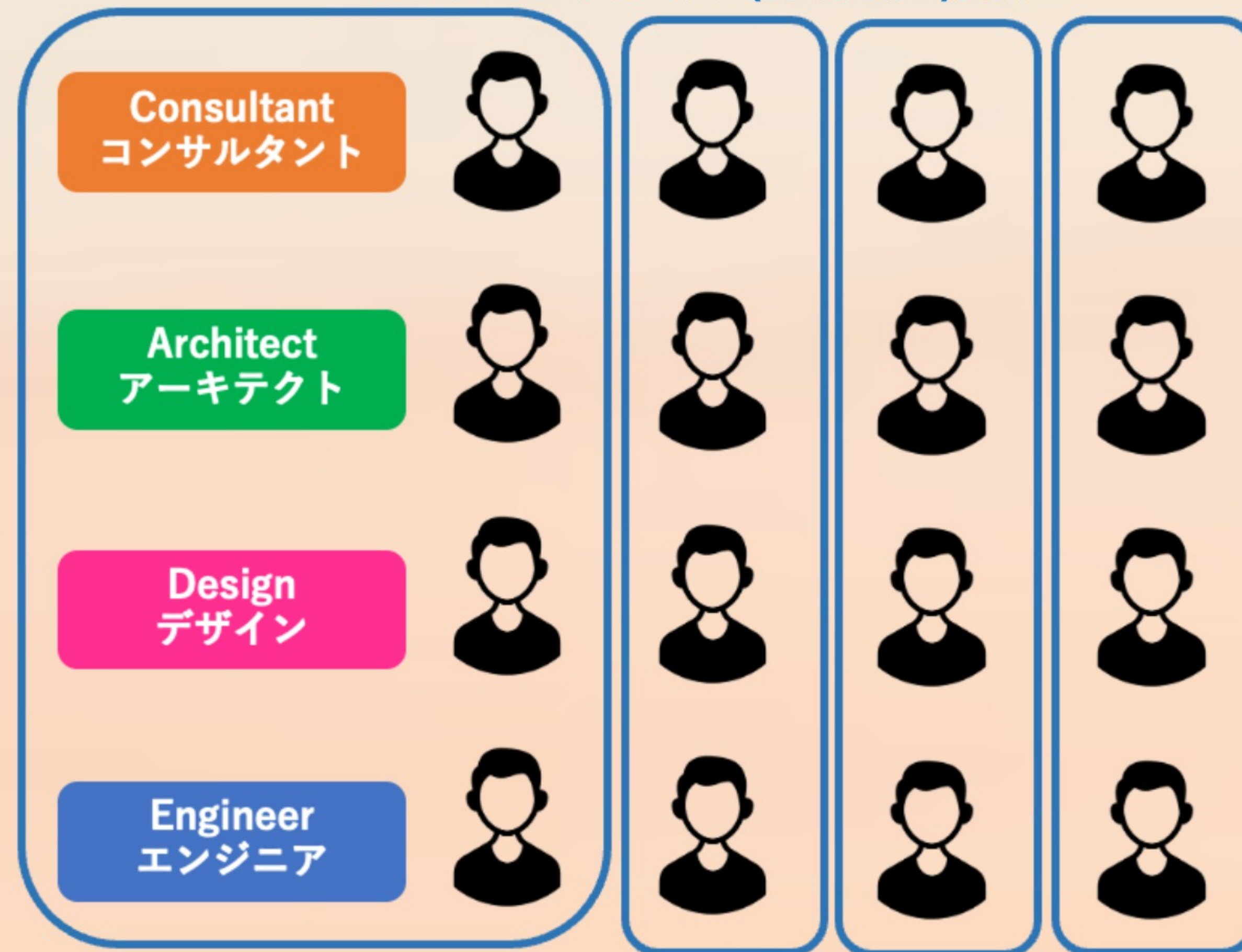
縦割りの組織



ビデオで見たチームはクロスファンクショナルチームでした。  
クロスファンクショナルチームは、リリースまでに必要なスキルがチーム内に備わっていることです。さらにメンバーは複数の役割を担えるフルスタック人材(T字型スキル)を目指していきます。またクロスファンクショナルチームは、各部門の調整が必要なくなり(短縮され)、スピードとイノベーションが促進されます。

## Cross functional Organization

クロスファンクショナル(機能横断型)組織





# Mobs to help you acquire skills

**Mob is a way of having everyone on a team work on the same thing at the same time, in the same space, and on the same computer.**

**You may have heard of mob programming, but teams that practice it use it in many situations, including definition creation as well as software development.**

**Mob is a evolved form from pair work! Pair work is two people working together, while mobs are three to five people working together.**

## Expected Benefits

### Can reduce the problems that teams have

Communication problems such as waiting for answers to questions, time spent exchanging email conversations, and potential misunderstandings from communicating via documents.

### Share knowledge with the team and develop cross-functional teams

Learn knowledge and skills from each other. Senior members may have technical skills and junior members may have creative skills.

### Increase speed

Context-switching (switching project work) among team members can lead to loss of time and focus. Mobs allow work to progress because even if one team member is asked to switch contexts, the rest of the team can continue.

### Provide rapid feedback.

There is always discussion and quick feedback from the members of the mob, and there is no need for code reviews.

## スキル習得に役立つモブ

モブ(Mob)とは、チーム全員が同じことを同時に、同じスペースで、同じコンピューターで作業する方法です。

モブプログラミングを聞いたことがある方も多いかもかもしれませんが、モブを実践するチームは、ソフトウェア開発だけではなく、プランニングや、定義の作成なども含めて多くの場面で使用しています。ペアから進化したのがモブです！ペアは2名で共同作業をしますが、モブは3人～5人で共同作業を実施します。

## 期待できる効果

### チームが持つ問題を軽減できる

質問への回答を待つ、電子メールでの会話のやりとりにかかる時間、ドキュメントを介し生じる可能性のある誤解などのコミュニケーションの問題。

### チームで知識を共有しクロスファンクショナルなチームの育成

お互いから知識やスキルを学ぶことができます。シニアメンバーは、技術力を持ち、ジュニアメンバーは創造性を持っているかもしれません。

## スピードの向上

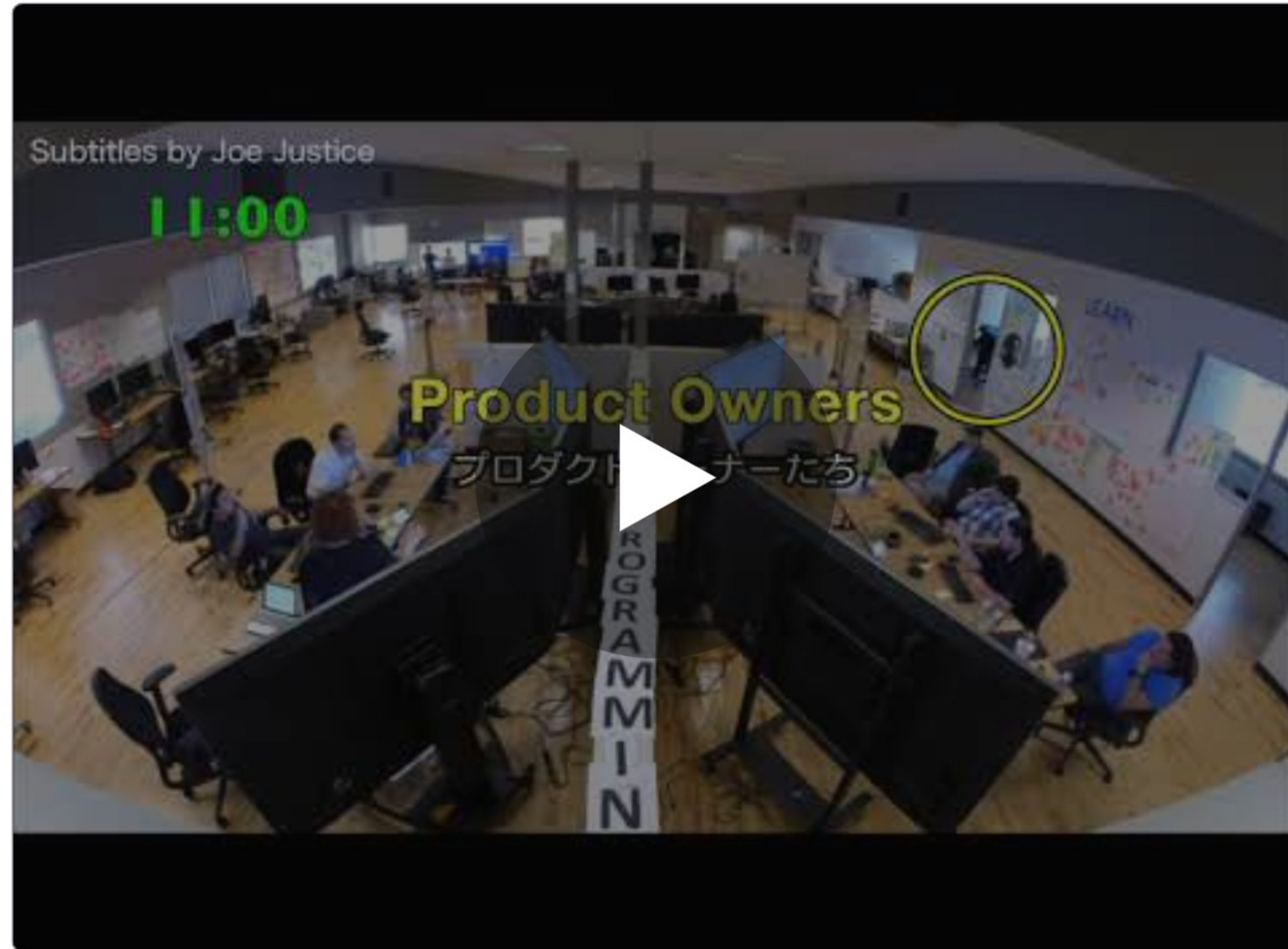
チームメンバーのコンテキストスイッチ(プロジェクト作業の切り替え)は、時間と集中力の喪失につながります。モブは、1人のチームメンバーがコンテキストの切り替えを求められた場合でもチームの他のメンバーが続行できるため、作業は進行できます。

## 迅速なフィードバックの提供

常にモブのメンバーから議論があり、迅速なフィードバックが得られ、コードレビューや、情報共有のための会議などは必要ありません。



# Let's watch the actual mob!



Please see a video from Hunter in Silicon Valley to see what great Mob looks like in an office. Hunter is now 100% remote, which looks like this class with zoom and break out rooms. This video in their former office is easier to see how the mobs work.

The video is about 3 minutes in translation. There is no audio.

シリコンバレーにあるHunterのビデオで、素晴らしいMobがどのようなオフィスなのかをご覧ください。ハンターは現在100%リモートで、ズームやブレイクアウトルームを使ったこのクラスのような感じです。以前のオフィスでの映像は、モブがどのように動いているのかがわかりやすいと思います。

ビデオは訳3分程度です。音声はありません。

<https://youtu.be/HEaz71juXiM>



# Mobs to help you acquire skills

The mob has the following roles

**Driver:** the person who uses tools

**Navigator:** the role of giving directions and advice

Again, there are many advantages to mobbing, as indicated below!

モブには下記の役割があります。

ドライバー：打ち込み役

ナビゲーター：指示やアドバイス役

繰り返しですが、以下に示した通りモブの利点はたくさんあります！





# Let's try mob

Please write texts together with Mob!

<scenario>

How would an agile team benefit your business?

Suppose you go back to your office tomorrow and have to convince your boss and team. Write a draft!

アジャイルなチームは、ビジネスにどのようなメリットをもたらすでしょうか？  
オフィス戻り、上司やチームを説得するための原稿を作ってください。

- ナビゲーター：記載する内容を指示する。
- ドライバー：スクリーンシェアをして、ナビゲーターの指示に従いタイプする。
- モブ：クオリティをチェックしたりリサーチして、アドバイスする。

■Navigator: Provide instructions on what to include in the email text.

■Drivers: Screen-share and type according to the navigator's instructions.

■Mob: Check the quality, do research, and give advice.

Rotation in 3 minutes  
Please have someone from  
your team volunteer to  
measure 3 minutes.

navigator  
ナビゲーター

Tomoko

Driver  
ドライバー

Joe

Mob  
モブ

Miho

## How would an agile team benefit your business?



# **Please share what your team has written!**

**The last driver, please share your  
script which your team wrote  
together with Mob !**

最後にドライバーをした方、チームが  
書いた原稿を読んでもください！



# Mobs increase team speed and fun

I understand the benefits, but it's going to reduce the number of outputs and slow down the process.

メリットはわかったけどアウトプットの数が減ってスピードダウンしそう

As people become accustomed to mobbing, meetings to share information become unnecessary. Also, since development is done while reviewing, there is no need for rework. In addition, since the team executes items in order of priority, the highest priority items can be completed first.

モブに慣れてくると情報共有のための会議などは不要になります。またレビューしながら開発をしているような状況のため手戻りもなくなります。さらにチームで優先度の高いアイテムから実行するため、優先度の高いものから順に完了させることができます。

## Solo Work ソロワーク

Dev A

Dev B

Dev C

Information  
Sharing  
Who does  
what?  
情報共有  
誰が何をやる？

Priority  
1

Development  
開発

Priority  
2

Development  
開発

Priority  
3

Development  
開発

Review  
Information  
Sharing  
レビュー  
情報共有

Priority  
1

Work to fix  
修正

Priority  
2

Work to fix  
修正

Priority  
3

Done  
完了！

**Sprint ends!**  
スプリント終了！

## Mob Work モブワーク

Dev A

Dev B

Dev C

Priority  
1

Development  
(Review)  
(information sharing)  
開発  
(レビュー)  
(情報共有)

Priority  
2

Development  
(Review)  
(information sharing)  
開発  
(レビュー)  
(情報共有)

Priority  
3

Development  
(Review)  
(information sharing)  
開発  
(レビュー)  
(情報共有)



# Mob Case Studies

Below is an actual case study. Please use it as a reference.

Facts about Mob Efficiency:  
Economies of Swarming and Mobbing:  
[https://www.youtube.com/watch?v=B\\_KWyptanAM](https://www.youtube.com/watch?v=B_KWyptanAM)

Mobbing in Scrum, How could this possible be efficient?  
<https://www.youtube.com/watch?v=h-baijxbBYs>

mob effectiveness:  
<https://trello.com/b/1lfMkCOh/software-profession-resources>

<Mob Experiences shared by the past student👉>

Here are some experiences from past students  
I've been working on Mobs for 4 hours a day and I can finish my work faster.  
I can see the benefits especially in code reviews.  
Advantages  
The members are there and they can see it on the spot.  
No need to have another meeting.  
It costs more to verbalize and communicate later.  
Some people are not good at verbalizing, so it is quicker to review and correct on the spot.  
Since skills and knowledge are shared, work can continue even if a member is absent (no need to wait).



下記は実際の事例です。参考にしてみてください。

サイボウズ株式会社事例:  
[http://www.jaspic.org/event/2020/SPIJapan/session5/5-2\\_ID006.pdf](http://www.jaspic.org/event/2020/SPIJapan/session5/5-2_ID006.pdf)

Yahoo!JAPAN事例:  
<https://techblog.yahoo.co.jp/entry/2020052730002064/>

ZOZO事例:  
<https://techblog.zozo.com/entry/mob-programming>

<過去の受講生の経験談👉>

1日4時間Mobをしています。とにかく早く仕事が終わる!  
特にコードレビューで効果が見えます。  
利点:  
メンバーがその場にいるのでその場で見てもらえる  
改めて会議をする必要がない  
言語化して後で伝えるのはコストもかかる  
言語化するのが苦手な人もいたのでその場で見て直した方が早い  
スキルや知識を共有できているのでメンバーが休んでも仕事が続行できる(待たなくてよい)





**Did the mob work?**  
**Think about Kaizen to make**  
**it even better!**





# Kaizen

**"Kaizen"** As many of you know, it is one of the most famous Japanese words. Kaizen is the process of eliminating waste in a task or operation, and carrying out activities to change the way a task or operation is performed for the better.

**There are always problems lurking in the field. If you work in the field every day, you may become accustomed to such problems and may not consider them as problems anymore. Therefore, it is essential to continuously engage in kaizen activities to recognize and share problems.**

「カイゼン」知っている方も多いと思いますが、有名な日本語の一つです。カイゼンとは、作業や業務の中にあるムダを排除し、作業や業務のやり方をより良く変える活動を行うことです。

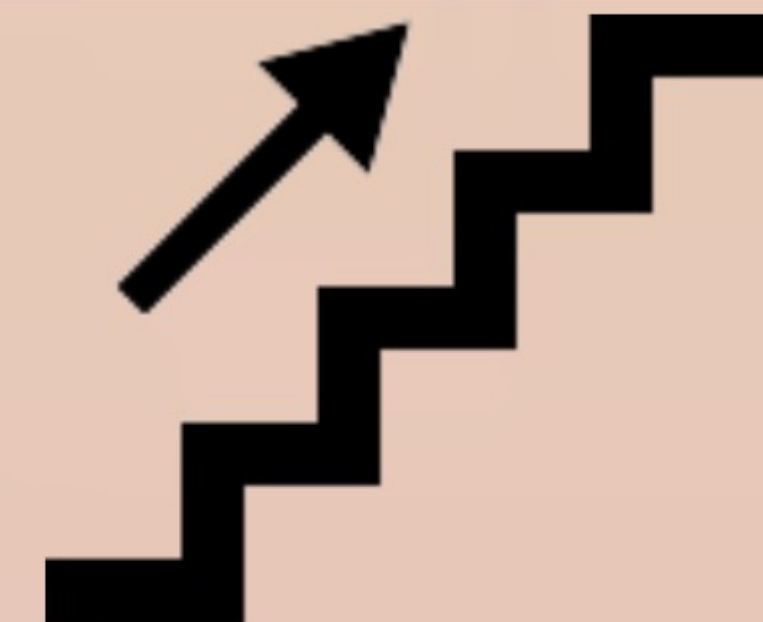
現場には必ず問題が潜んでいます。毎日現場で働いていると、その問題にも慣れてしまい、問題を問題と思えなくなってしまうことがあります。従って、問題に気づき共有するカイゼン活動の取り組みを継続的に行うことが欠かせません。

## improvement



**Turning a bad situation into a good one.**  
悪い状態を良い状態に変える

## Kaizen



**Never be satisfied with the current status  
and continue to change for the better**  
現状へ満足せずより良い状態へ変化し続ける



# Let's try kaizen

Think about mob kaizen using the "KPTA" used in one kaizen format!

一つの改善フォーマットで使われる「KPTA」を使って、モブの改善について考えてみましょう！

## Keep

良かったこと・  
続けること

What were the good things, and  
what do you want the team to  
continue doing?

## Problem

不満なこと・  
改善すべきこと

What was the problem or what  
does the team want to improve?

### STEP1

Share "Keep" and "Problem" within the team

Each team member writes "Keep" and "Problem" on a chart while looking back on the mob. Share "Keep" and "Problem" within the team.

### STEP2

Share "Tries" within the team

Write and display improvement proposals for "Keep" and "Problem" in "Try". Even if it is not related to "Keep" and "Problem", also write "Try" for new challenges.

Share "Tries" with your team.

### STEP3

Decide "Action"

"Try" is an improvement, and you don't have to do all of the things listed. The selected "Try" is put into action and materialized. Write this in "Action" and display it in the table.

## Try

試したいこと

What would you like to challenge  
next?

## Action

アクション

What exactly should the team do?





# Let's try kaizen

## STEP1

**Share “Keep” and “Problem” within the team**

**Each team member writes “Keep” and “Problem” on a chart while looking back on the mob. Share “Keep” and “Problem” within the team.**

## STEP2

**Share “Tries” within the team**

**Write and display improvement proposals for “Keep” and “Problem” in “Try”. Even if it is not related to “Keep” and “Problem”, also write “Try” for new challenges.**

**Share “Tries” with your team.**

## STEP3

**Decide “Action”**

**“Try” is an improvement, and you don’t have to do all of the things listed. The selected “Try” is put into action and materialized. Write this in “Action” and display it in the table.**

## STEP1

チーム内で、「Keep」と「Problem」を共有

チームメンバーは各自、モブを振り返りながら、「Keep」と「Problem」を書いて表に表示します。チーム内で、「Keep」と「Problem」を共有します。

## STEP2

チーム内で、「Try」を共有

「Keep」「Problem」に対する改善案を「Try」に書いて表示します。「Keep」「Problem」とは関連しないが、新たにチャレンジしたいことも「Try」として書きます。

チーム内で、「Try」を共有します。

## STEP3

「Action」決定する

「Try」は改善案であり、挙げたことを全て行う必要はありません。選択した「Try」をアクションに落とし込み具体化します。これを「Action」に書いて表に表示します。



# Example

会話しながら  
楽しくできた  
We had fun  
while  
conversing.

できる限りの時間の中で回せた。  
We were able to turn around in  
as little time as possible

K  
無言の時間が  
無かった。  
There was no  
silent time.

K  
思ったよりも、文章  
になった感じがする  
。 I feel like I got more  
out of the text than I  
thought I would.

結論まで  
まとまった  
文章ができた  
I was able to make a  
coherent sentence  
to a conclusion

楽しく会話  
で進められた  
Conversation  
was enjoyable.

書くことに  
集中  
できなかった  
Couldn't  
concentrate on  
writing.

P  
仕方ないが、もっと  
スムーズに出来たら  
、素晴らしいと思う  
No choice, but it  
would be great if it  
were smoother

Problem

不満なこと・

やっぱり  
モブは  
同室で  
やりたい  
I still want to do the  
mob in the same  
room

あ、ボタン押してない★  
があったw  
気づけたよね、全員w  
I didn't press the  
button. You noticed  
that, didn't you?



あ、ボタン押してない★  
があったw  
気づけたよね、全員w  
I didn't press the  
button. You noticed  
that, didn't you?

タブのところに時間  
が出ているか、モバ  
ーが確認する。  
Mober checks to see  
if the time is  
showing at the tabs.

Try  
試したいこと

モバーの一人をタイムキー  
パーにする。Driverの下の人  
がタイムキーパーをする。  
One of the Mover's will be  
the timekeeper; the person  
under Driver will be the  
timekeeper.

モバーの役割  
->ナビゲータの  
補助的  
The Role of the  
Mover  
->Navigator's  
Assistant

Action  
アクション

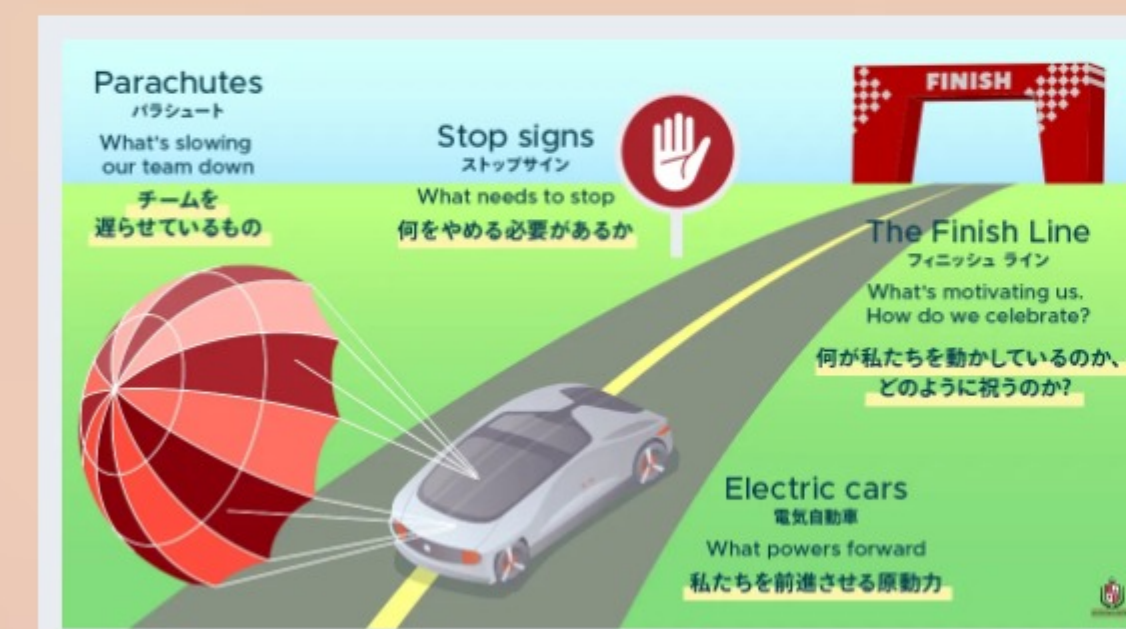
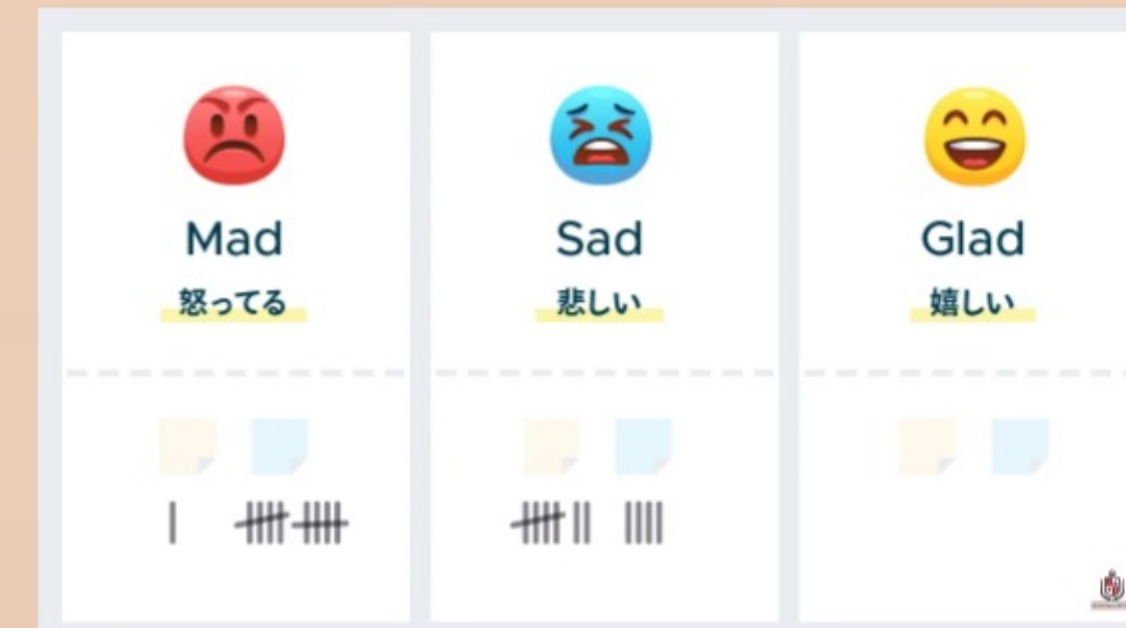
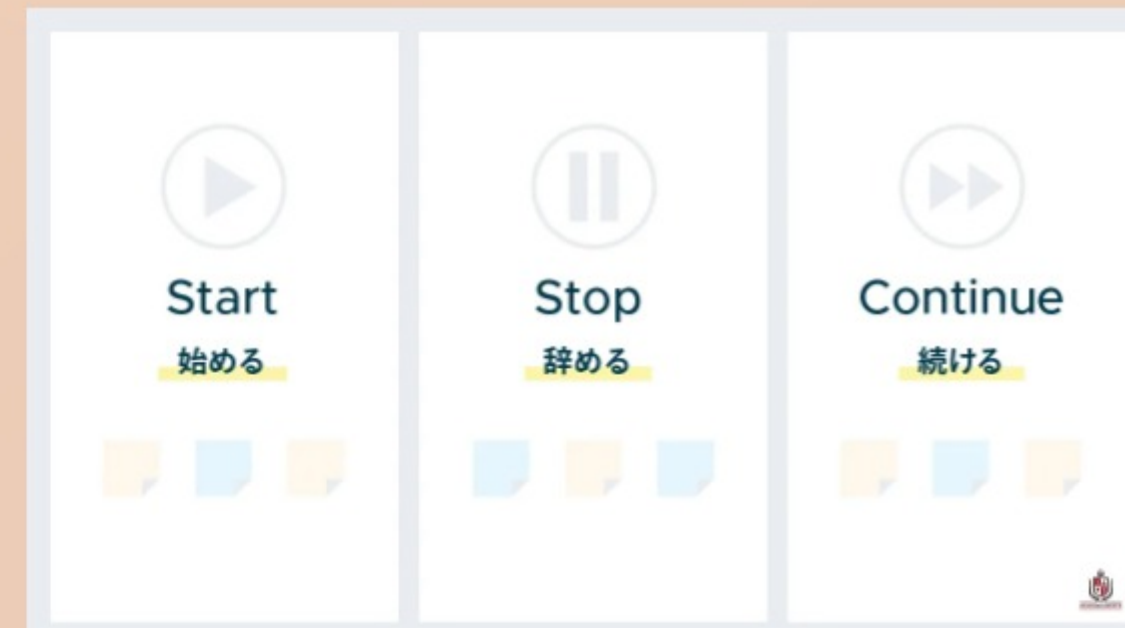
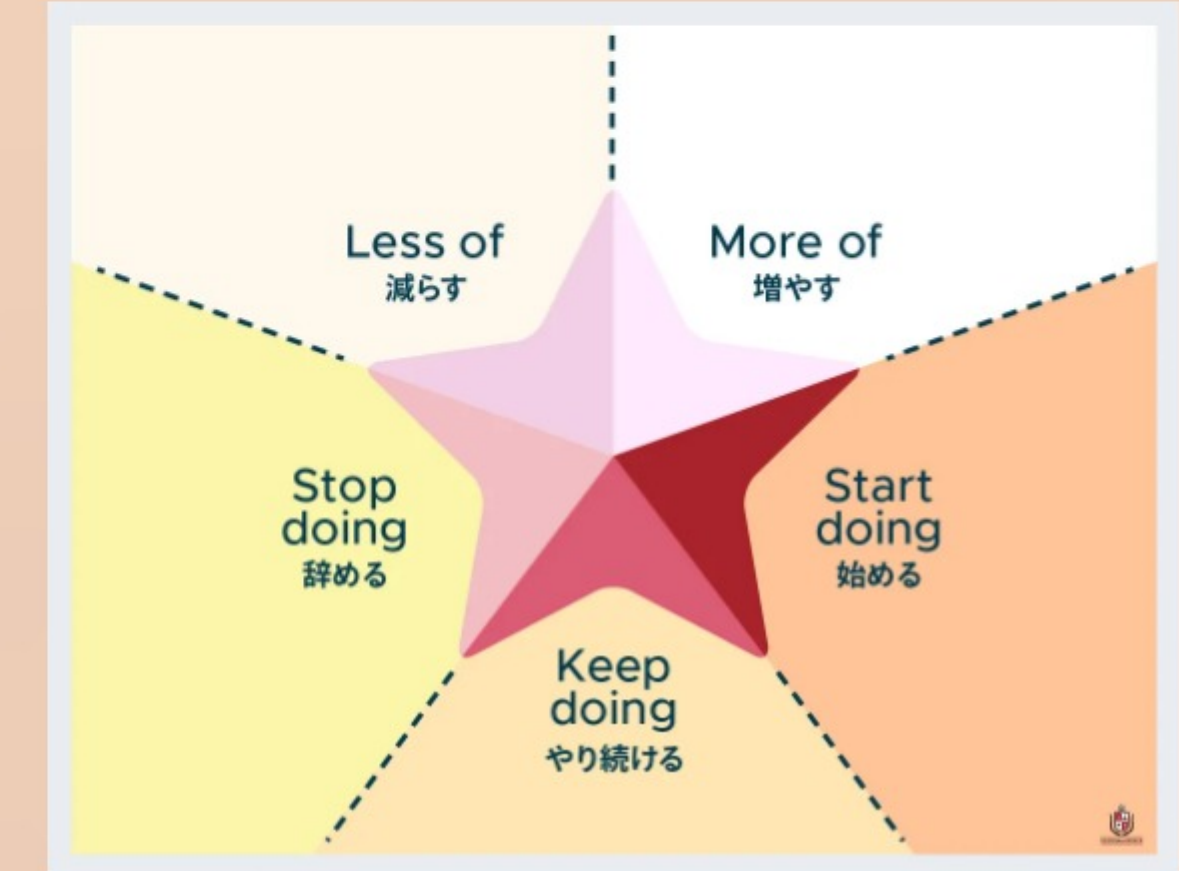




# There are many ways to identify improvements

There are many ways to identify improvements within a team.  
I have also written an article on the subject, which I hope you will open the link below and go through.

チーム内で改善を特定するやり方はたくさんあります。記事も書いてますので、読んでいただければ嬉しいです。



<https://en.abi-agile.com/retrospective/>



# # No Estimate

## #見積もりなし

The #NoEstimates movement began with [WoodyZuill's](#) 2012 Twitter, a post that questioned the validity of using estimates to predict project costs and timelines. Many people supported this and there is still a lot of debate going on even now.

Trying to complete work as estimated can sometimes put unhealthy pressure on you, and can lead to glitches due to prioritizing deadlines, and in the long run, it could lead slow progress.

#NoEstimatesムーブメントは、[ウッディズィール](#)の2012年のTwitterから始まり、プロジェクトのコストと納期を予測するために見積もりを使用することの有効性を疑問視した投稿から始まりました。これを支持する人は多く、現在でも多くの議論が起っています。

見積もりどおりに作業を完了しようとすることは、時に不健康な圧力を与え、納期を優先したことによる不具合の発生、そして長期的には進歩の低下につながる可能性があります。



# Is the current estimate really useful?

There is no estimate or plan that is ever 100% accurate. A lot of uncertainty exists, especially in the early stages of a project.

Agile estimation is meant to acknowledge that uncertainty and provide correct estimates.

In other words, we need to think about providing estimates that are useful and correct (in the true sense of the word, including uncertainty), rather than estimates that are 100% precise but actually incorrect.

This will reduce uncertainty about delivery as the project progresses, as knowledge is acquired, and as decisions are made and re-planned.

## 現在の見積りは本当に有用なのか？

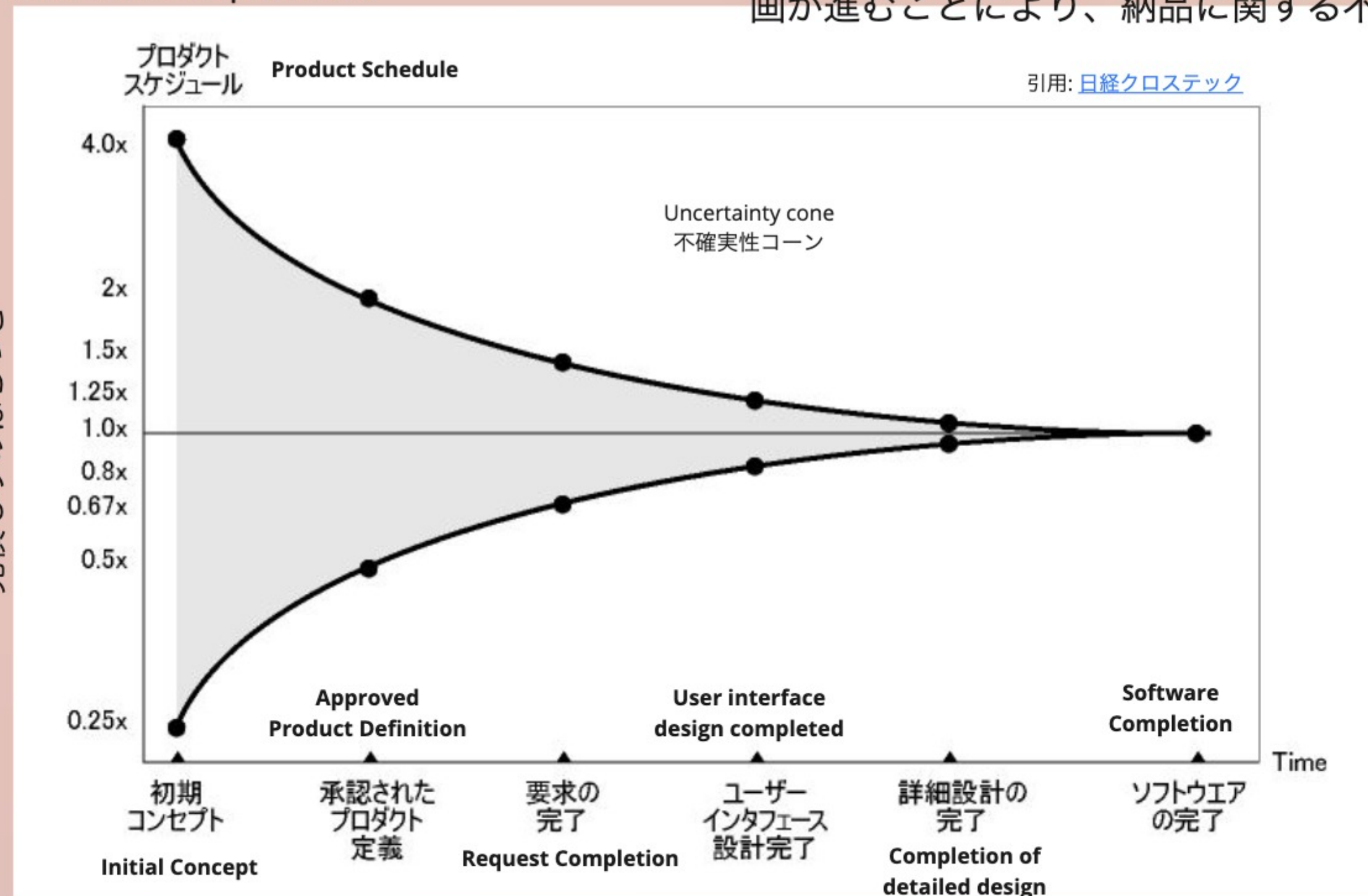
100%正確な見積もりや計画はありません。特にプロジェクトの初期段階では、多くの不確実性が存在しています。

アジャイルの見積りは、その不確実性を認めて正しい見積もりを提供することに意味があります。

つまり、目標を100%正確な見積もりから、有用で(不確実性を含めて本当の意味で)正しい見積もりの提供を考える必要があります。

これにより、プロジェクトが進展する中で知識を習得し、意思決定や再計画が進むことにより、納品に関する不確実性を減らすことができます。

Estimate Uncertainty  
見積もりのばらつき



There is no need to estimate all of the deeper, more detailed items in the backlog. By the time the team actually starts working on these items, the requirements may have changed, and the application will certainly have changed. Therefore, previous estimates are not very accurate. Estimating work that is subject to change is a waste of time.

バックログのより深い細かいアイテムを、全て見積もる必要はありません。実際にチームが、これらのアイテムに取り組み始めるまでに、要件が変更される可能性があり、アプリケーションは確かに変更されます。したがって、以前の見積もりはそれほど正確ではありません。変更の可能性のある作業の見積もりは時間の無駄になってしまいます。



# Analogize the size

Estimate the size, not the period. Use relative, not absolute, estimates. An estimate is only a guess based on good knowledge. Use all the knowledge and experience at hand to estimate the size.

Rather than looking at every new work item individually, compare it to previously completed work items. People are much better at making analogies than guessing the actual size of things.

For example, is it similar to a smaller one from a previous Project A?  
Or is it closer to the big one in Project B?

Relative estimates not only reduce the time it takes to estimate a task, but also improve the accuracy of the estimate.

Our brains are not capable of making absolute estimates. Humans always think of estimates in terms of comparing new things in relation to the knowledge and experience they already know.

## サイズを類推する

期間を見積もるのではなく、サイズを見積もります。絶対的な見積もりではなく、相対的な見積もりを使用します。

見積もりは、十分な知識に基づいた推測にすぎません。手元にあるすべての知識と経験を使用して、そのサイズを推測します。

すべての新しい作業アイテムを個別に見るのではなく、以前に完成した作業アイテムと比較してみます。物の実際のサイズを推測するよりも、人間は類推する方が遥かに優れています。

たとえば、それは過去のプロジェクトAの小さなものに近いか？それとも、プロジェクトBの大きなものに近いか？

相対的な見積もりを行うと、作業の見積もりにかかる時間が短縮されるだけでなく、見積もりの精度も向上します。

私たちの脳は絶対的な推定を行うことはできません。人間は常にすでに知っている知識や経験との関係で、新しいことを比較して見積りを考えているのです。

プロジェクトA  
(Example: fix a bug)  
(例：○○1つのバグ修正)

New  
Project

プロジェクト B  
(Example: New app  
with 10 features)  
(例：10の機能を搭載した新アプリ)

S

M

L

XL





# replace with a number

## Fibonacci sequence estimate

This sequence of numbers, devised by the mathematician Fibonacci, is the sum of the previous two numbers, such as 5, 8, and 13, which becomes the next number. It is found in nature and is also used to estimate population growth and predict the behavior of financial markets.

**1) Standardization:** Instead of estimating person-hours or time, estimate the complexity and effort points of the work. This makes it possible to standardize the estimate since it does not depend on individual skills, etc., but can only be determined by correlation with other tasks.

**2) Clarification:** 2P can be reasonably estimated to be twice as complex as 1P. But how much more complex is 12P compared to 11P? The exponential nature of the Fibonacci sequence makes it easy for the whole team to understand the meaning of the numbers assigned. For example, they can understand that 1 has minimal complexity and can complete its task quickly, while 13 is complex and may take weeks to complete.

## 数字に置き換える。フィボナッチ数列 見積もり

数学者フィボナッチが考え出したこの数列は、5、8、13のように直前の2つの数の和が次の数となる。自然界に多く見られ、人口増加を推定や、金融市場の行動を予測でも使われている。

**1) 標準化：**工数や時間を見積もるのではなく、その作業の複雑さや労力ポイントを見積もります。そのため個人のスキルなどに依存せず、他の作業との相関で決めるしかないので、見積もりを標準化することができます。

**2) 明確化：**2Pは、1Pの2倍複雑であると合理的に見積もることができます。しかし、12Pは11Pと比べてどれほど複雑なのでしょう。フィボナッチ数列の指数関数的な性質により、チーム全体が割り当てられた数値の意味を簡単に理解できます。たとえば、1は複雑さが最小限であり、迅速にその作業を完了でき、13は複雑で完了するまでに数週間かかる可能性があることを理解できます。

S

M

L

XL

1

2

3

5

8

13

21

34

55

89

144

233



# Consider the delivery date

One reason for estimating would be to know the delivery date.

A release burndown chart can be used to come up with a delivery date.

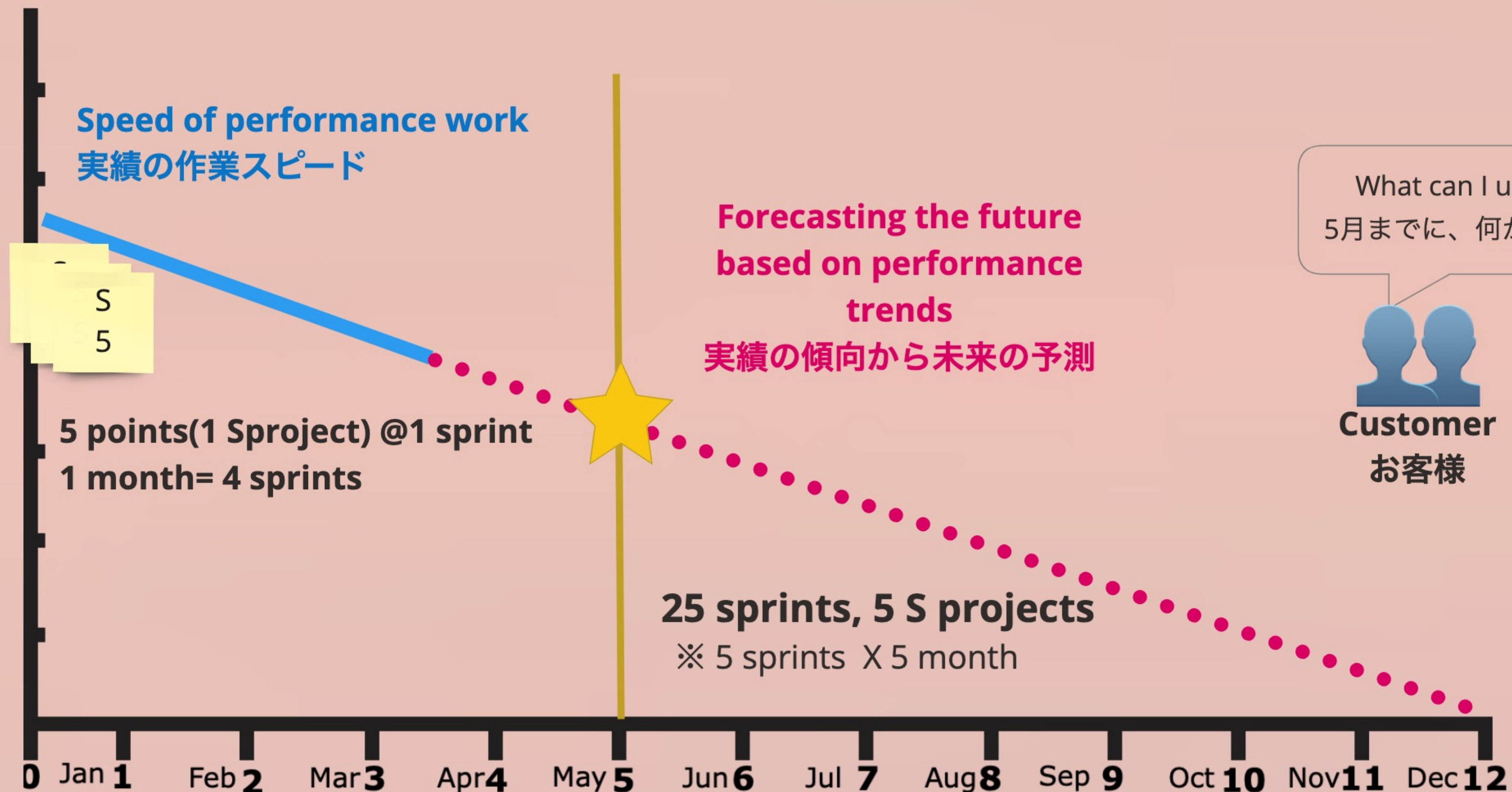
By showing work speeds and trends, future projections can be made.

## 納期を考える

見積もりをするのは納期を知りたいことが理由の一つでしょう。

リリースバーンダウンチャートを使用して納期を考えることが可能です。

作業スピードや傾向を示すことで、未来の予測を立てられるようになります。







**#NOBUDGETS**

**#予算なし**





# Beyond Budgeting

Performing all operations according to a predetermined budget can be difficult given an uncertain environment.

BeyondBudgeting started a movement in the 1990s that criticized the traditional budget management system.

Today, some companies are adding evolution to BeyondBudgeting.

For example, during the Covid19 crisis, budgeting for 2020 would have made little sense. Traditional budgeting has proven to be of little value in a VUCA (volatile, uncertain, complex, and ambiguous) world.

あらかじめ決まった予算にしたがって、すべての業務を遂行することは、不確実な環境では困難となる場合があります。

BeyondBudgetingは伝統的な予算管理システムのあり方を批判したムーブメントを1990年代に起こしました。

現代では、脱予算といった進化をしていく企業もあります。

たとえば、コロナウイルス危機の間、2020年の予算編成はほとんど意味のないものではなかったでしょうか。従来の予算編成はVUCA時代(不安定で、不確実で、複雑で、あいまい)ではあまり価値がないことが明らかになりました。

## What are some of the problems with traditional budgeting?

- Are changing market conditions taken into account on a case-by-case basis?
- Is time and effort wasted on budgeting?
- Does keeping the budget lead to business growth?
- Are there any unnecessary expenditures to spend the budget?

## 伝統的な予算管理の問題点はどのようなところにあるのでしょうか？

- ・ 市場の変化状況は都度考慮されている？
- ・ 予算編成のために時間と労力を浪費していない？
- ・ 予算を守ることがビジネスの成長に繋がるのか？
- ・ 予算消化のための不必要な支出はないか？



# Alternatives to traditional budgeting

Supporting agile decision making is probably inseparable from budgeting. Perhaps it is inevitable that many organizations in a variety of industries are looking to improve their approach to budgeting.

## Rolling budgets

Organizations budget on an ongoing basis at regular and short intervals. Forecasts are prepared on a monthly, quarterly, or weekly basis rather than annually. With smart rolling budgets (AI-automated budgeting), everyone has access to up-to-date data, so changes can be tracked and adapted quickly.

## Members conduct budgeting

Rolling budgeting is delegated to teams, not to senior management.

Teams are empowered to make decisions and respond to changes in the business environment, fostering a culture of innovation.

俊敏な意思決定をサポートすることは予算と切り離すことはできないでしょう。さまざまな業界で多くの組織が予算編成のアプローチを改善しようとしていることは必然的なかもしれません。

## 従来の予算編成の代替案

### ローリング予算

組織は定期的かつ短い間隔で継続的に予算編成を行います。予測は、年次ではなく月次、四半期、もしくは週単位で作成されます。スマートローリング予算(AIで自動化した予算管理)を使用すると、誰しもが最新のデータを入手できるため、変化をすばやく追跡して適応できます。

### メンバーが予算編成を実施

ローリング予算の編成は、経営幹部ではなくチームに委任される。


チームはビジネス環境の変化に対応し、意思決定を行う権限が与えられ、イノベーションの文化が育まれていきます。




Below are leadership principles and management processes Reference.

以下は、リーダーシップの原則と管理プロセス 参考にしてみてください。

Beyond Budgeting - enabling business agility	
Leadership principles	Management processes
1. <b>Purpose</b> - Engage and inspire people around bold and noble causes; <b>not around short-term financial targets</b>	7. <b>Rhythm</b> - Organise management processes dynamically around business rhythms and events; <b>not around the calendar year only</b>
2. <b>Values</b> - Govern through shared values and sound judgement; <b>not through detailed rules and regulations</b>	8. <b>Targets</b> - Set directional, ambitious and relative goals; <b>avoid fixed and cascaded targets</b>
3. <b>Transparency</b> - Make information open for self-regulation, innovation, learning and control; <b>don't restrict it</b>	9. <b>Plans and forecasts</b> - Make planning and forecasting lean and unbiased processes; <b>not rigid and political exercises</b>
4. <b>Organisation</b> - Cultivate a strong sense of belonging and organise around accountable teams; <b>avoid hierarchical control and bureaucracy</b>	10. <b>Resource allocation</b> - Foster a cost conscious mind-set and make resources available as needed; <b>not through detailed annual budget allocations</b>
5. <b>Autonomy</b> - Trust people with freedom to act; <b>don't punish everyone if someone should abuse it</b>	11. <b>Performance evaluation</b> - Evaluate performance holistically and with peer feedback for learning and development; <b>not based on measurement only and not for rewards only</b>
6. <b>Customers</b> - Connect everyone's work with customer needs; <b>avoid conflicts of interest</b>	12. <b>Rewards</b> - Reward shared success against competition; <b>not against fixed performance contracts</b>

 © BBRT 2016 – All rights reserved | www.bbirt.org1

Beyond Budgeting 脱予算のステップ - enabling business agility ビジネスアジリティを可能に	
リーダーシップの原則 Leadership principles	管理プロセス Management processes
1. 目的:リーダーは、短期的な財務目標ではなく、大胆で高貴な目的について人々を巻き込み、刺激します	7. リズム:管理プロセスは、暦年と会計年度だけでなく、ビジネスのリズムと活動を中心に動的に編成されます。
2. 価値観:リーダーは、厳格な規則や規制ではなく、共通の価値観と健全な判断を導きます。	8. ターゲット:リーダーは、固定されたカスケードされた目標ではなく、方向性があり、野心的で、相対的なターゲットを選択します。
3. 透明性:リーダーは、自己規制、創造性、学習、および制御のために情報に簡単にアクセスできるようにします。	9. 計画と予算:リーダーは、厳格で政治的な演習ではなく、クリーンで偏りのないプロセスを計画および予測します。
4. 組織:リーダーは、階層的な制御と官僚主義を取り除くために、強い帰属意識を確立し、説明責任のあるチームを促進します。	10. リソースの割り当て:リーダーは、コスト意識を高め、必要に応じてリソースを利用できるようにします。具体的な年間予算配分ではありません
5. 自律性:リーダーは行動する自由を持った人々を信頼します。	11. パフォーマンス評価:パフォーマンスは、全体的に評価され、学習と能力開発のためのピアフィードバックが使用されます。測定のみに基づくものではなく、報酬のみに基づくものでもありません。
6. 顧客:リーダーは、すべての行動が顧客のニーズにリンクされ、利益の衝突が回避されることを確認します。	12. 報酬:リーダーは、固定されたパフォーマンス目標に対してではなく、競争に対して共有された成功に報酬を与えます。

 © BBRT 2016 – All rights reserved | www.bbirt.org1

<https://www.bbirt.co.uk/beyond-budgeting/bb-bbo.html>



DX, or Digital Transformation, is now endorsed by the government of Japan.  
The Japan government now endorses an agile contract!  
<https://drive.google.com/file/d/1dVqC8v0SUp7LVcBvk20rzVEhWGqZbwuJ/view?usp=sharing>

モデル取引・契約書見直し検討部会

DX 対応モデル契約見直し検討 WG

～情報システム・モデル取引・契約書＜アジャイル開発版＞～

アジャイル開発外部委託モデル契約

2020 年 3 月  
(2021 年 10 月 6 日 更新)

独立行政法人情報処理推進機構

経済産業省

<https://www.ipa.go.jp/files/000081484.pdf>



# Let's talk about Agile at the team meeting!

**Finally, discuss agile with your team!**

最後は、アジャイルに関してチームと話し合いをしましょう！





# Do you have any of these problems in your team meetings?

- Agenda preparation is difficult
- Long discussions
- There is a lot of unnecessary discussions
- Some people don't participate in the discussion



- アジェンダの準備が大変
- 議論が長引く
- 無駄な議論が多い
- 話に参加しない人がいる

## Lean coffee helps !

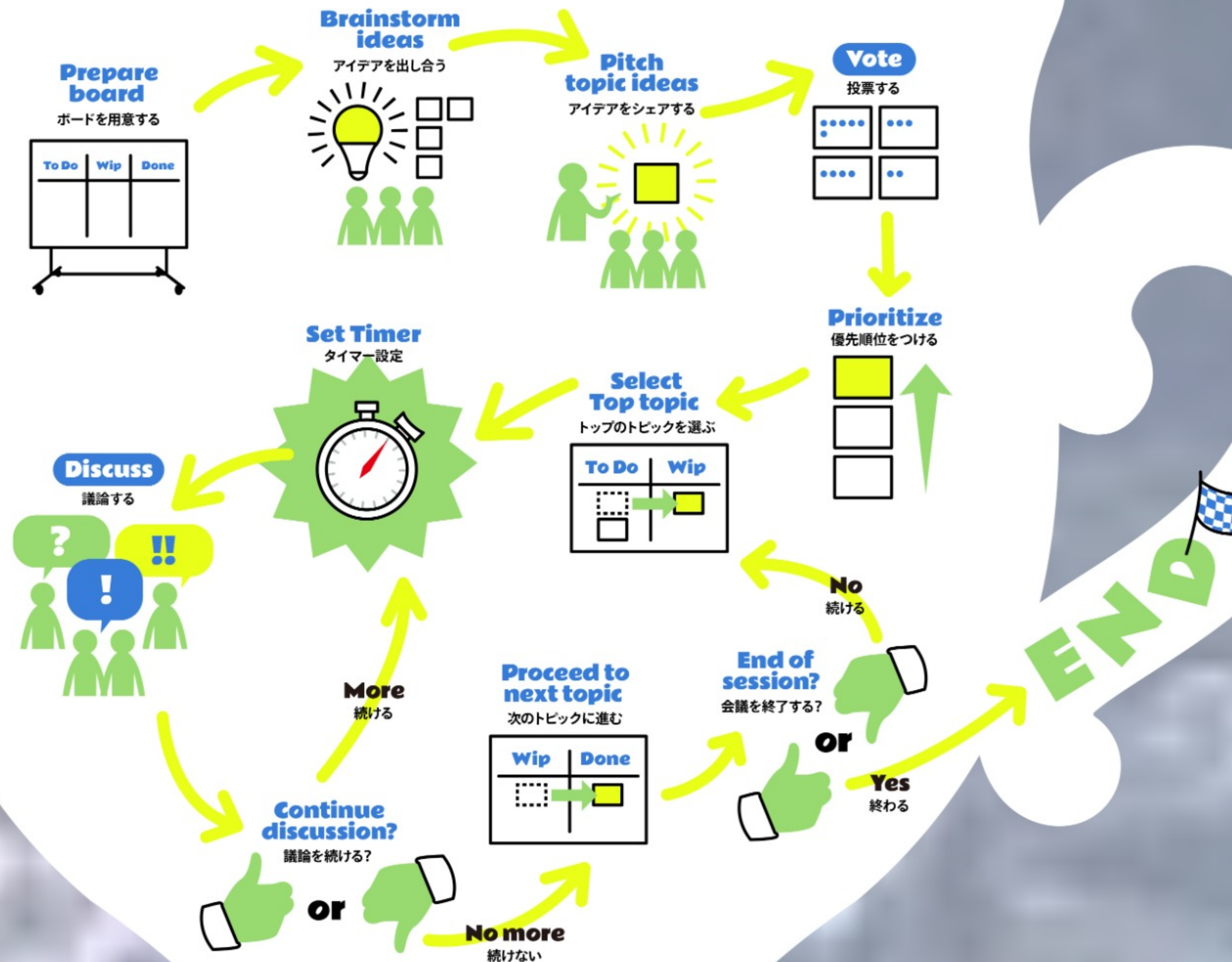
- Agenda is set by the team
- Discussions are managed by a timer
- Teams choose the agenda
- Show awareness of interest or lack of interest



- アジェンダはチームで決める
- 議論はタイマーで管理する
- 議題はチームが選択する
- 関心があるかないか意識表示する



# Let's try Lean Coffee!



1. All participants write their names and the topic they wish to discuss on a card. There is no limit to the number of topics. (One topic per card; for readability, please keep the number of words per card to a minimum.)
2. Place those cards in the "To-Do" column.
3. Each person should introduce the topic and explain the idea of each card in a nutshell.
4. Each participant will put a dot vote to determine the topic to be discussed by the team. (Each participant has 3 votes and may cast all their votes on one topic or vote for more than one card.)
5. Arrange the topics in the order in which the votes were collected, starting at the top of the "To-Do" column.
6. Now that you have a genda, move the top item to the center "Doing" column.
7. Set a timer (e.g. on your smart phone) for 5 minutes. Or set it to whatever length the group deems appropriate. This is the first time box for the discussion.
8. When the time limit is reached, take a simple Roman poll (thumbs up, thumbs sideways, or thumbs down) to see if there is interest in continuing the discussion. If so, set the timer for a shorter time (e.g., 3 minutes) and continue the discussion. This procedure can be repeated as many times as necessary.
9. When you are done discussing a topic, move the card to the right ("Done") column. Next move the card with the highest priority from "ToDo" to "Doing" and repeat the process.
10. If necessary, at the end of the session, write down important points or action items from the group in the "Action" column.

1. 参加者全員が、自分の名前と話したいトピックをカードに書きます。トピックの数に制限はありません。(カードごとに1つのトピック、読みやすくするために、カードあたりの単語数を最小限に抑えてください。)
2. それらのカードを「ToDo」の列に置きます。
3. 各自トピックを紹介し、各カードのアイデアを端的に説明してください。
4. 各参加者はドット投票をして、チームで議論するトピックを決定します。(1人あたり3つの票をもち、1つのトピックにすべての票を投じたり、複数のカードに投票しても構いません)
5. 投票が集まった順に、トピックを「ToDo」列の上から並べます。
6. ジェンダができたので、一番上の項目を中央の「Doing」列に移動します。
7. タイマー(スマートフォンなど)を5分間に設定します。または、グループが適切と判断した長さに設定します。これは、議論の最初のタイムボックスです。
8. 制限時間に達したら、簡単なローマ式投票(親指を上に向ける、横向きにする、または親指を下に向ける)を行い、話し合いを続けることに関心があるかどうかを確認します。続ける場合は、タイマーをより短い時間(たとえば、3分)に設定し、話し合いを続けます。この手順を必要な回数繰り返すことができます。
9. トピックの議論が完了すれば、そのカードを右側(「完了」)の列に移動します。次に優先順位の高いカードを「ToDo」から「Doing」に移動し、プロセスを繰り返します。
10. 必要であれば、セッションの最後に、グループから重要なポイントやアクションアイテムを「Action」の列に書き出します。



# Example

1. All participants write their names and the topic they wish to discuss on their cards. (There is no limit to the number of topics. One topic per card; for ease of reading, keep the number of words per card to a minimum.) 2 minutes.
2. Each participant should introduce his/her topic and explain the idea of each card in a nutshell. 3 min.
3. Each participant will cast a dot vote to determine the topics to be discussed by the team. Arrange the topics in the order in which the votes are collected, starting at the top of the "To-Do" column. (Each person has 3 votes and may cast all their votes on one topic or on multiple cards.) 1 minute
4. Now that you have an agenda, move the top item to the center "Doing" column.
5. Start the discussion. Set a timer (e.g. smart phone) for 5 minutes. Or set it to whatever length the group deems appropriate. This is the initial time box for the discussion: 5 minutes.
6. Once the time limit is reached, conduct a simple Roman Poll (thumbs up, thumbs sideways, or thumbs down) to see if there is interest in continuing the discussion. If more than half of the team wishes to continue, set the timer for a shorter time (e.g., 3 minutes) and continue the discussion. This procedure can be repeated as many times as necessary: 1 minute to confirm and 3 minutes to discuss.
7. When the discussion of a topic is complete, the card is moved to the "Done" column. Next, move the highest priority card from "ToDo" to "Doing" and repeat the process.
8. If necessary, at the end of the session, write down important points or action items from the group in the "Action" column.



## Sample

How do you get total company buy in to start / implement Agile?  アジャイルを導入するために、会社全体の賛同を得るにはどうすべきか?
How can I increase communication in remote work?  テレワークにおいて、コミュニケーションを活性化させるにはどうすべきか?
Please provide toolset usage in Agile operations, including advice.  業務の一部を受託開発(固定納期、固定スコープ)で外注している場合、スクラム開発を行うことが難しいのか? この場合、どのようにスクラム開発を運営すべきか?
(Scrum does not require middle management) What is the middle manager role in scrum? (スクラムでは中間管理職を必要としないが) 中間管理職がいる場合、
Please provide toolset usage in Agile operations, including advice. アジャイル業務に役立つサービスやツールは何か? また、どのように使うのが効果的なのか?

Your own topics  
各自のトピック

Your own topics  
各自のトピック

To Do To discuss 議論すること  Priority: High 優先度:高	Doing Discussing 議論中  Your own topics 各自のトピック	Done Discussed 議論済み  Your own topics 各自のトピック	Action As needed 次のアクションとして必要であれば
Your own topics 各自のトピック  Priority: Low 優先度 :低			Optional オプション

1. 参加者全員が、自分の名前と話したいトピックをカードに書きます。(トピックの数に制限はありません。カードごとに1つのトピック、読みやすくするために、カードあたりの単語数を最小限に抑えてください。) 2分
2. 各自トピックを紹介し、各カードのアイデアを端的に説明してください。3分
3. 各参加者はドット投票をして、チームで議論するトピックを決定します。投票が集まった順に、トピックを「ToDo」列の上から並べます。(1人あたり3つの票をもち、1つのトピックにすべての票を投じたり、複数のカードに投票しても構いません) 1分
4. アジェンダができたので、一番上の項目を中央の「Doing」列に移動します。
5. 議論を開始します。タイマー(スマートフォンなど)を5分間に設定します。または、グループが適切と判断した長さに設定します。これは、議論の最初のタイムボックスです。5分
6. 制限時間に達したら、簡単なローマ式投票(親指を上に向ける、横向きにする、または親指を下に向ける)を行い、話し合いを続けることに関心があるかどうかを確認します。チームの半数以上が続行を希望する場合、タイマーをより短い時間(たとえば、3分)に設定し、話し合いを続けます。この手順を必要な回数繰り返すことができます。1分で確認し3分間話し合い
7. トピックの議論が完了すれば、そのカードを「完了」の列に移動します。次に優先順位の高いカードを「ToDo」から「Doing」に移動し、プロセスを繰り返します。
8. 必要であれば、セッションの最後に、グループから重要なポイントやアクションアイテムを「Action」の列に書き出します。



# Thank you for attending

**Thank you all for attending my session.**

**We offer classes on Agile.**

**We can also offer classes in English, so please feel free to contact us anytime.**

**Thank you very much for your time today.**

皆様私のセッションにご参加いただきありがとうございました。

私たちはアジャイルに関するクラスを提供しております。

英語での開催も可能ですので、いつでもお気軽にお問い合わせください。

本日は本当にありがとうございました。



**AGILE BUSINESS INSTITUTE**

Drives Enterprise Speed and Innovation

**<https://en.abi-agile.com>**