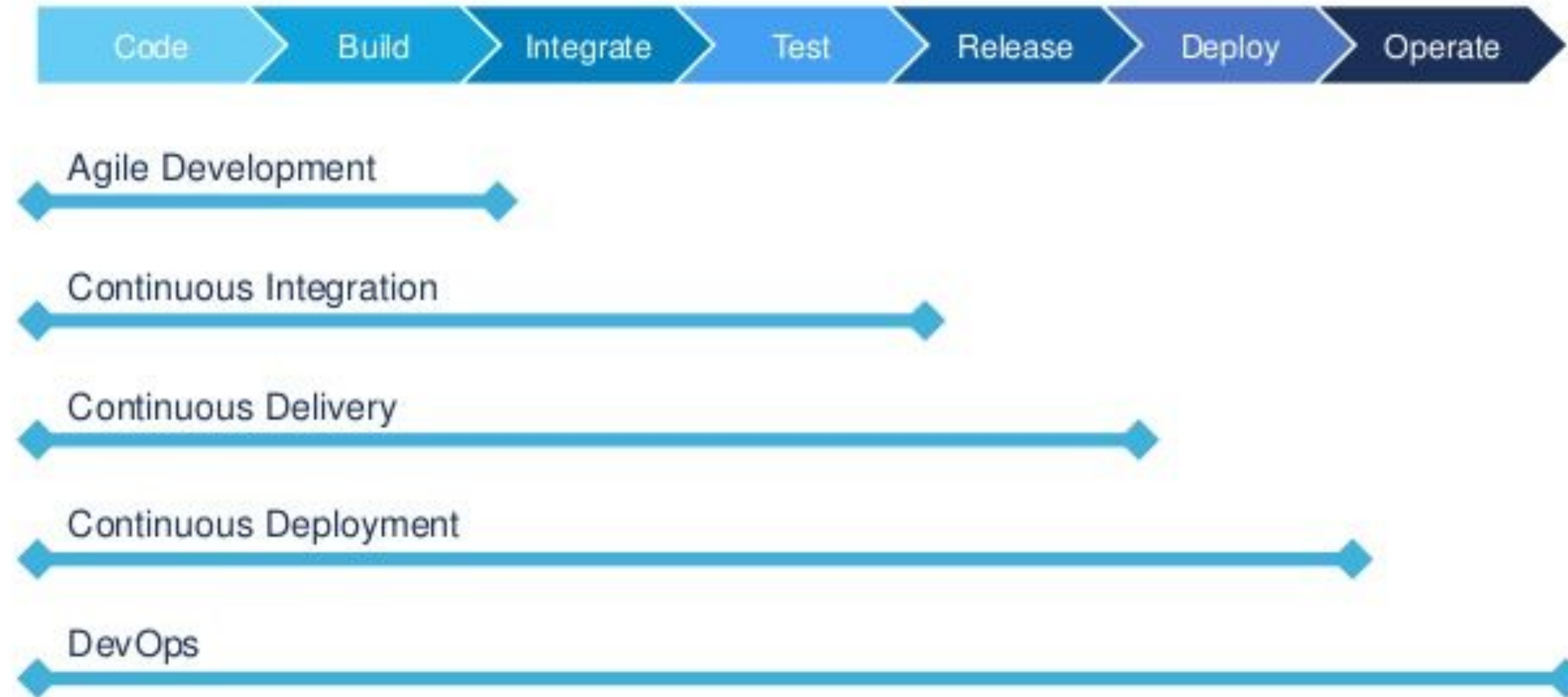
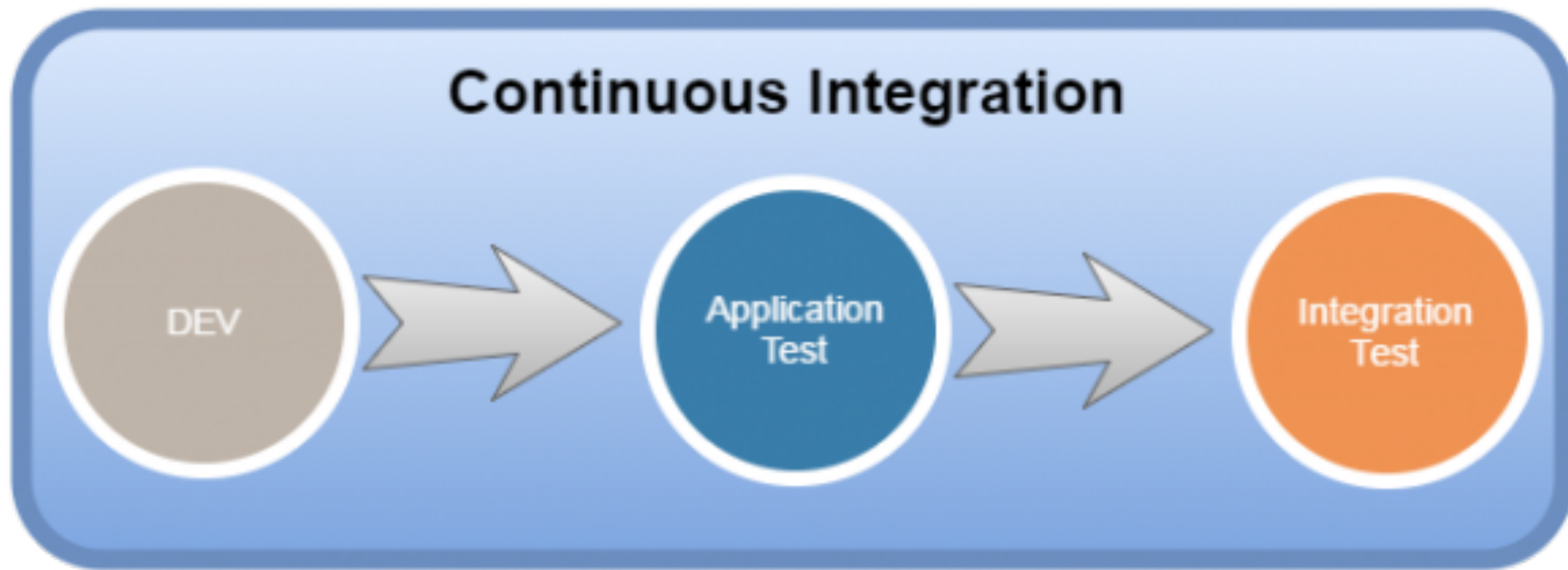


Что такое DevOps?

- Инженерные практики
- Доставка и сопровождение
- Jenkins / Kubernetes
- Особая служба
- Только один человек сказал, что это — культура

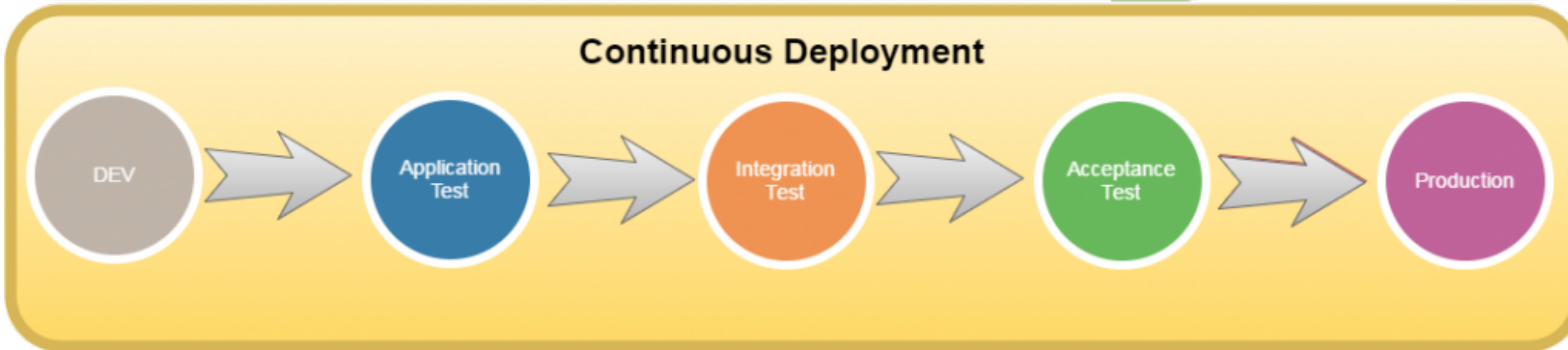
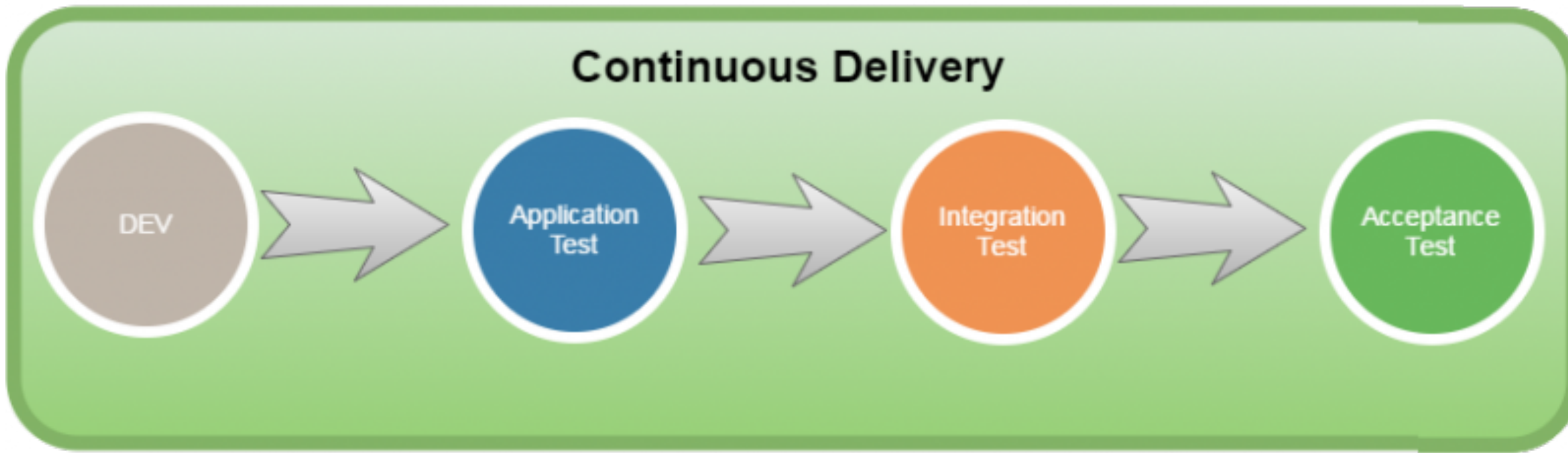
Your Roadmap to Application Agility

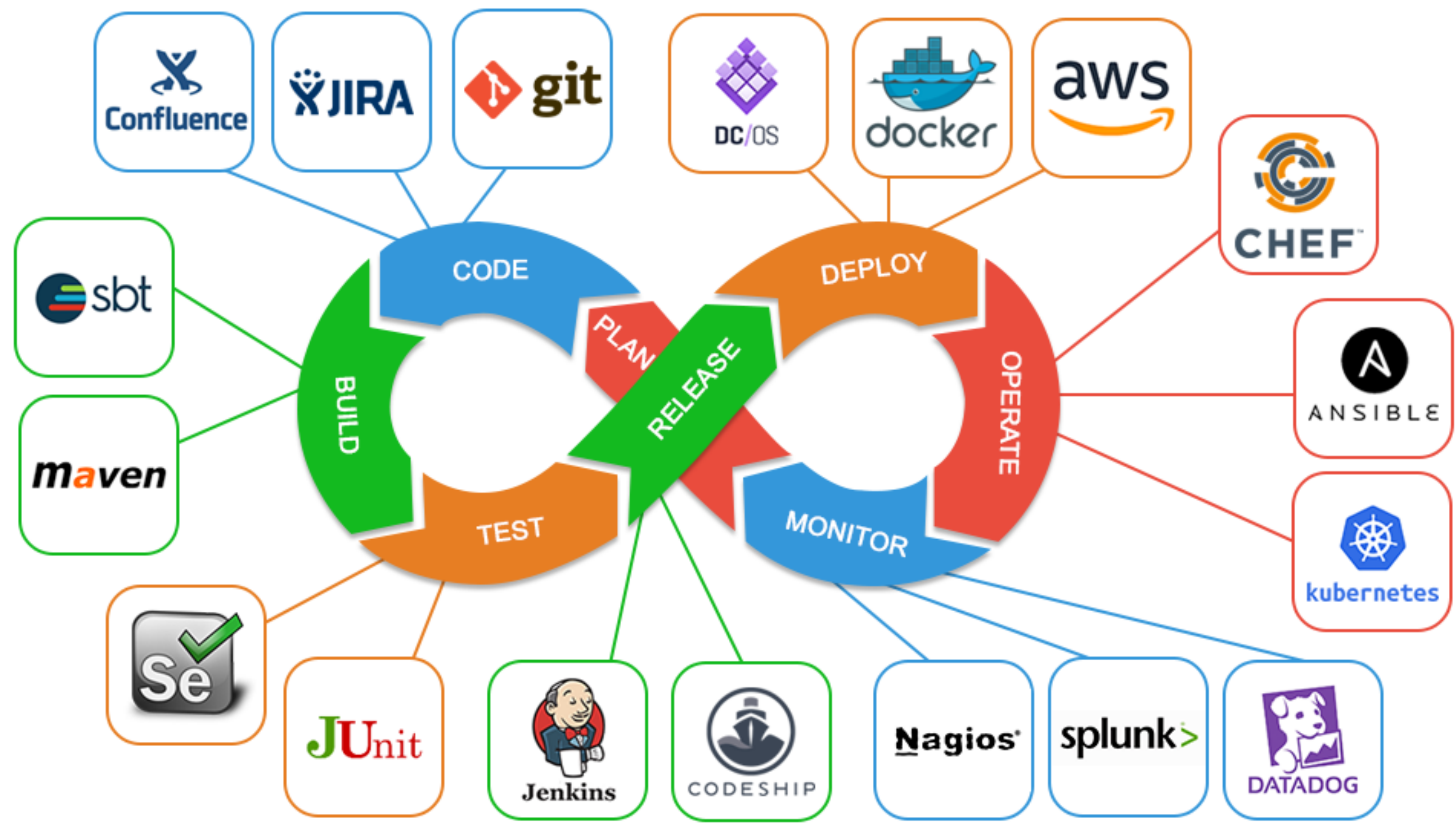




Automatic trigger

A grey arrow pointing to the right, with the text 'Automatic trigger' centered below it.





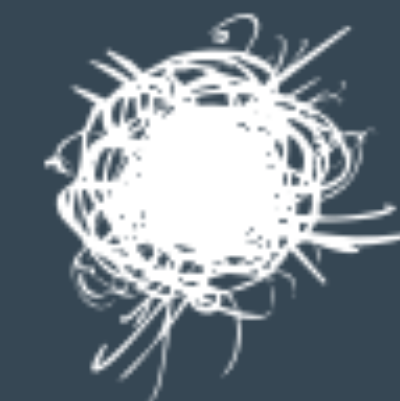
Три пути DevOps

Виталий Дмитриев

СТО Липтсофт



Хабр



Кто такие DevOps?

Администрирование

???





Кто ограничение?

Ошибок нет!





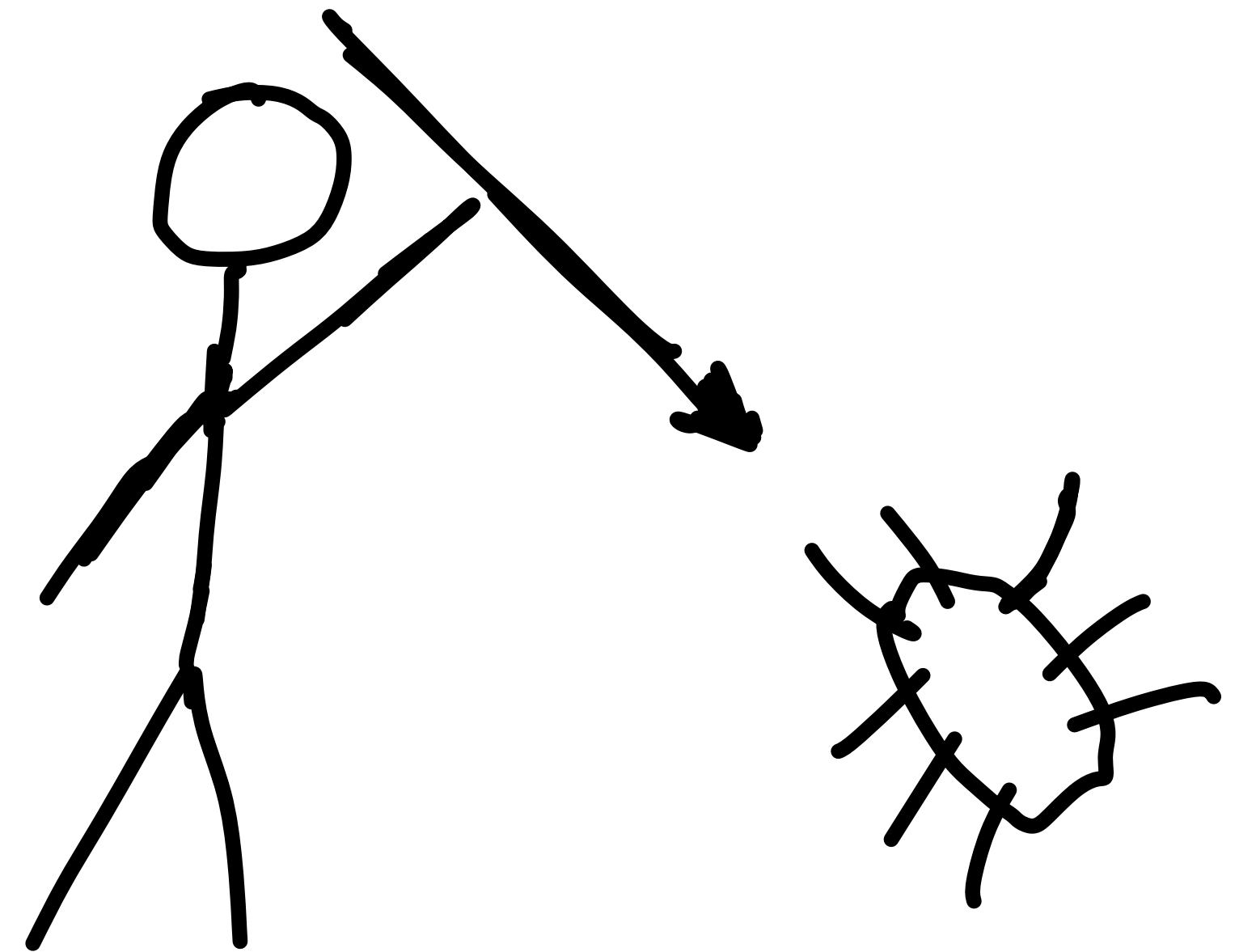
\$50/22.

x2



\$50/22.

x2

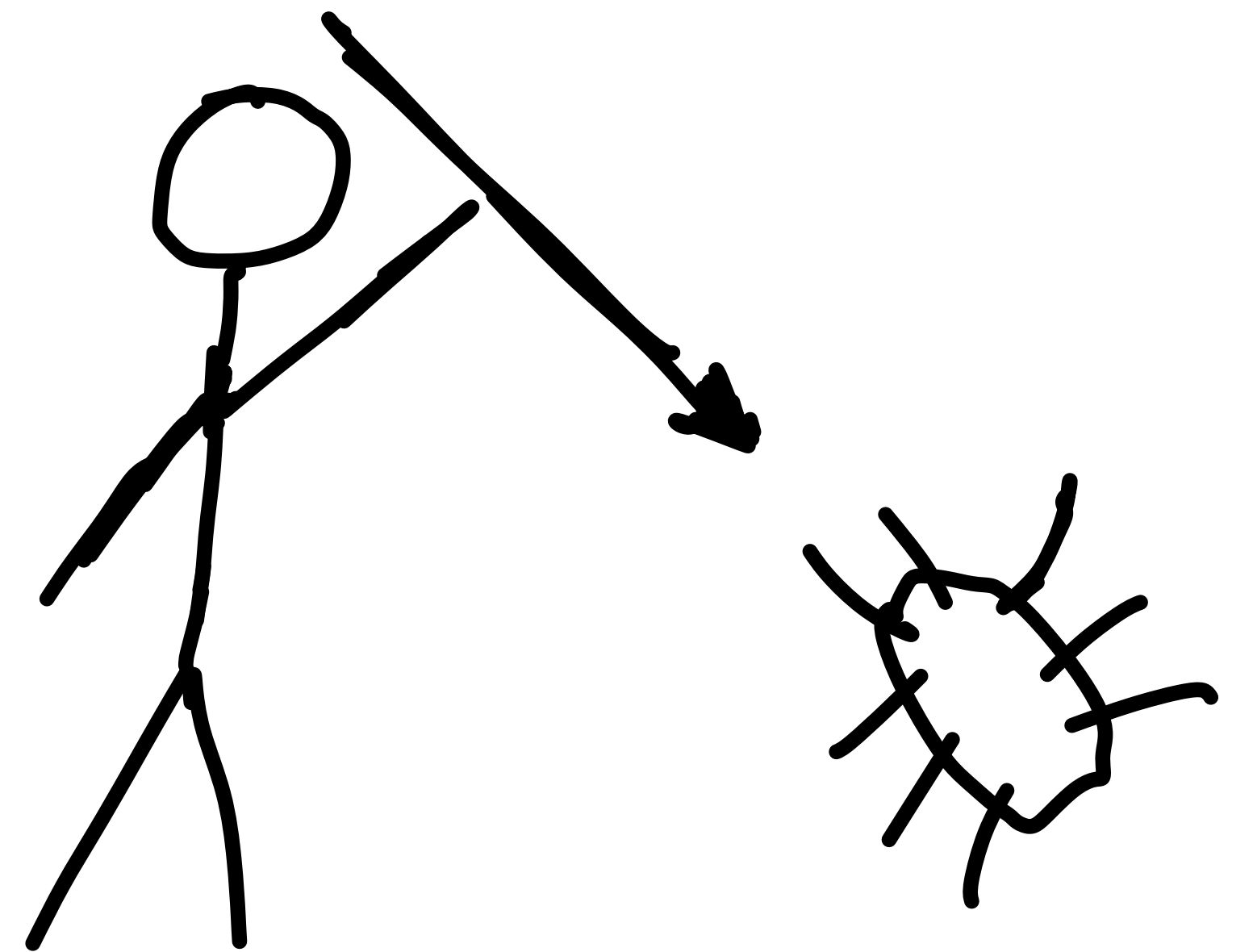


\$50/22.

x2



\$10/22.

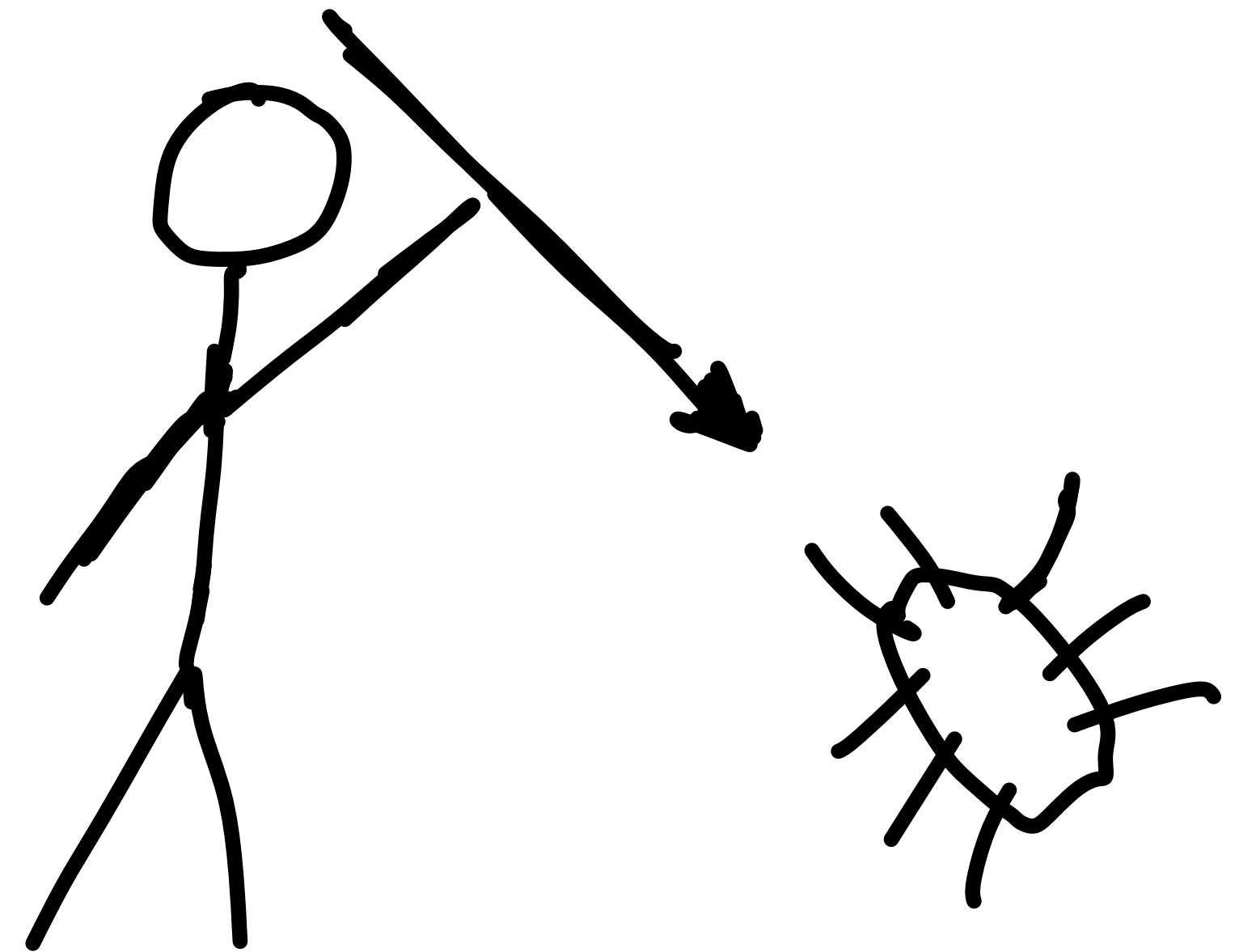


\$50/22.

x2

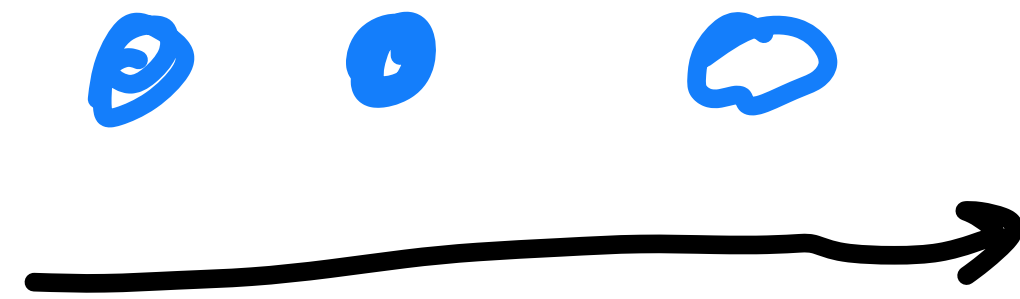


\$10/22.



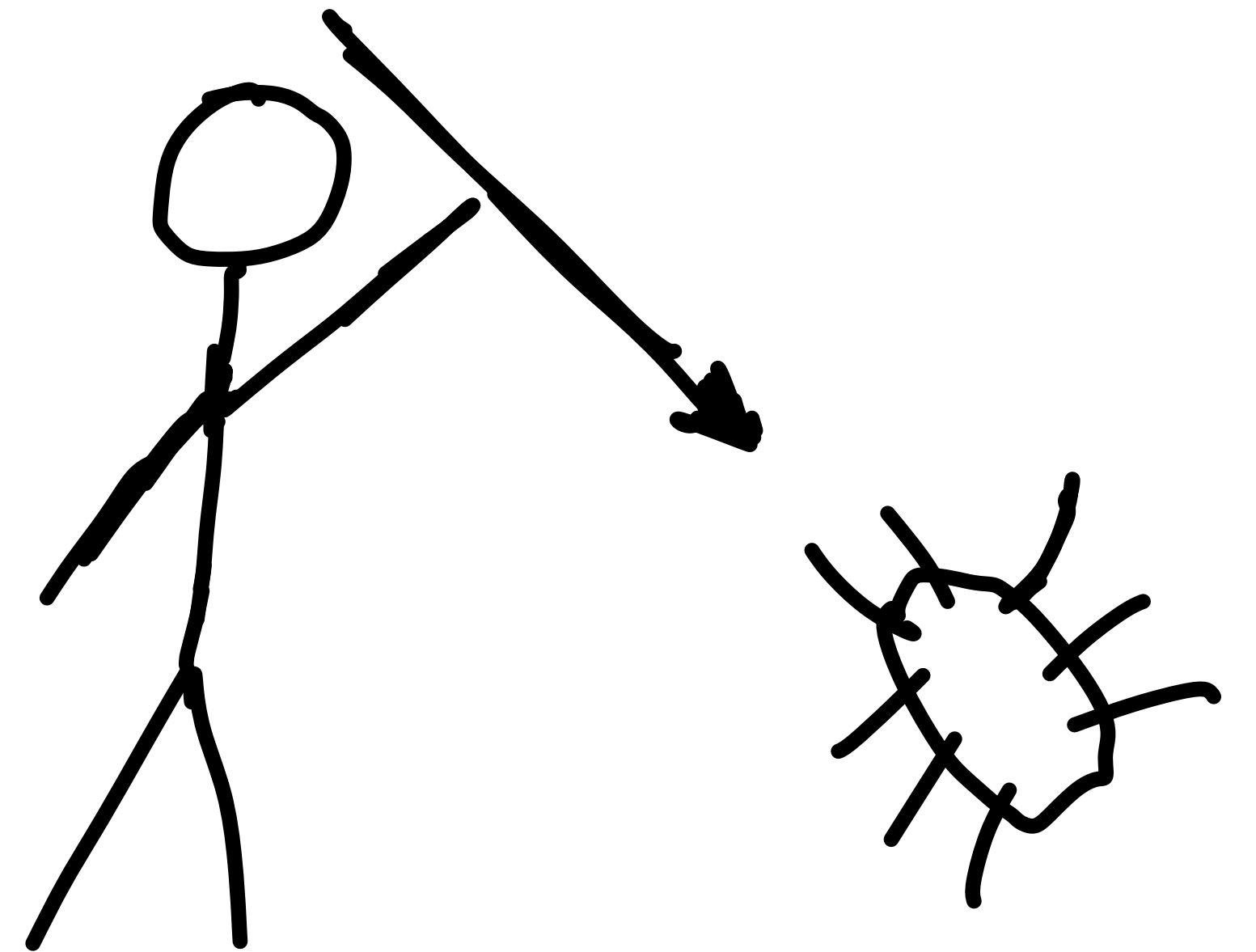
\$50/22.

x2



10/4

\$10/22.



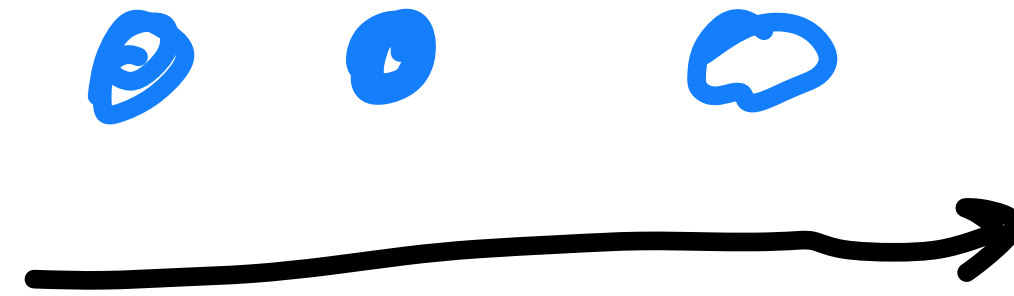
2/4

\$50/22.

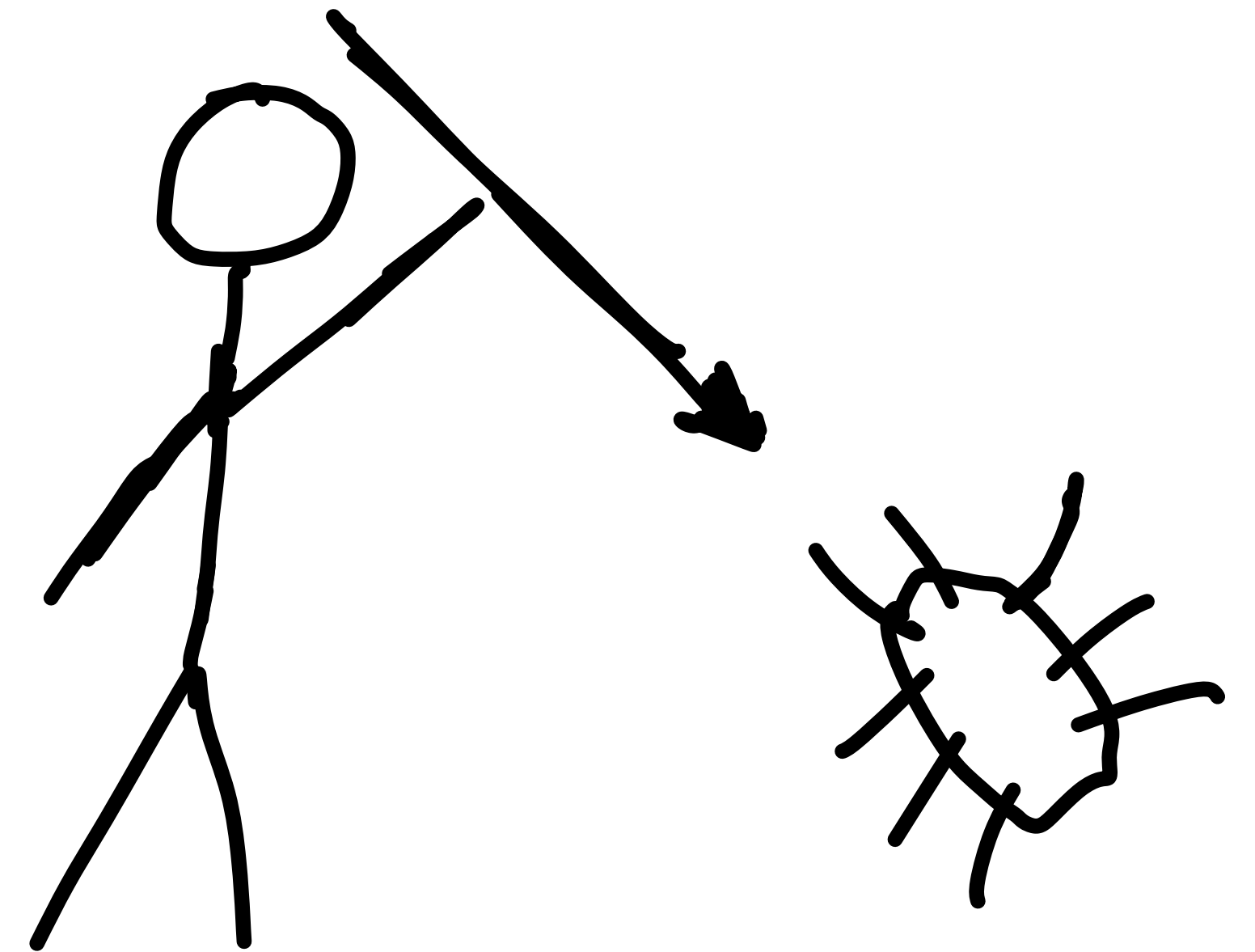
x2



\$10/22.



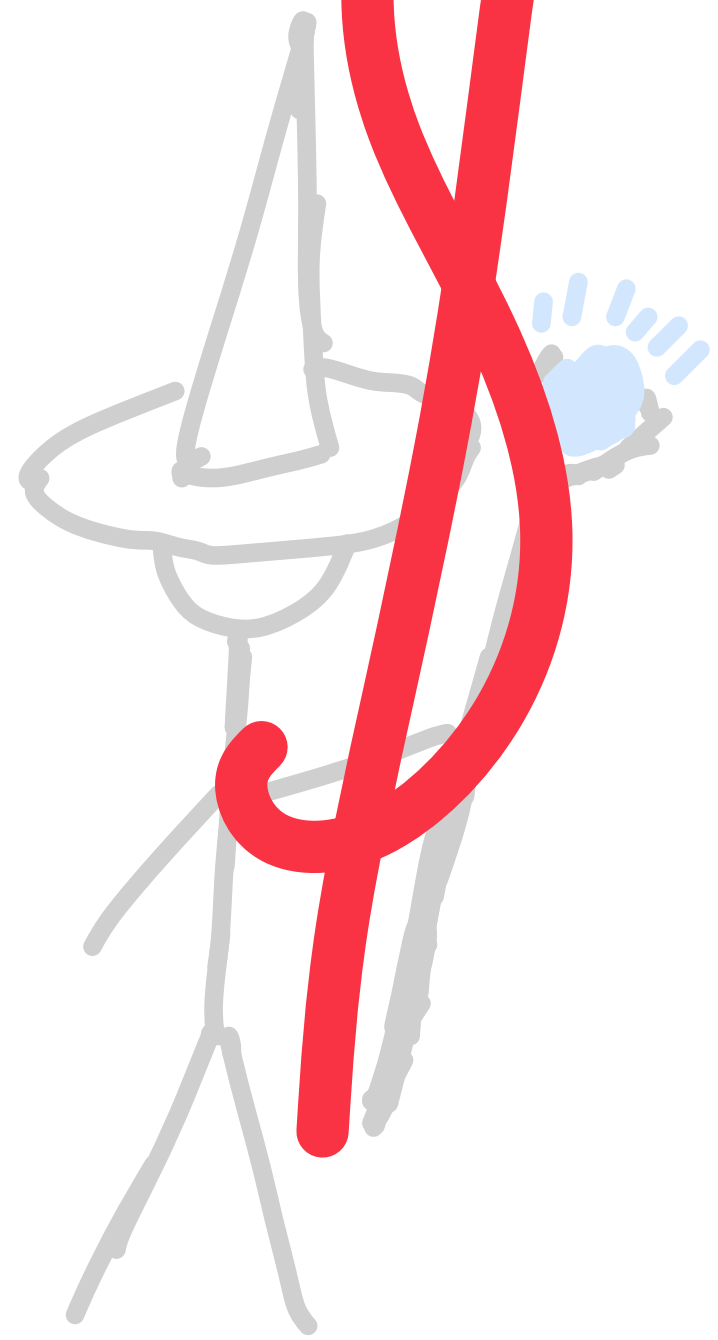
10/4



2/4

\$50/2.2.

x2



\$10/2.2.

4400/k



10/h



2/h

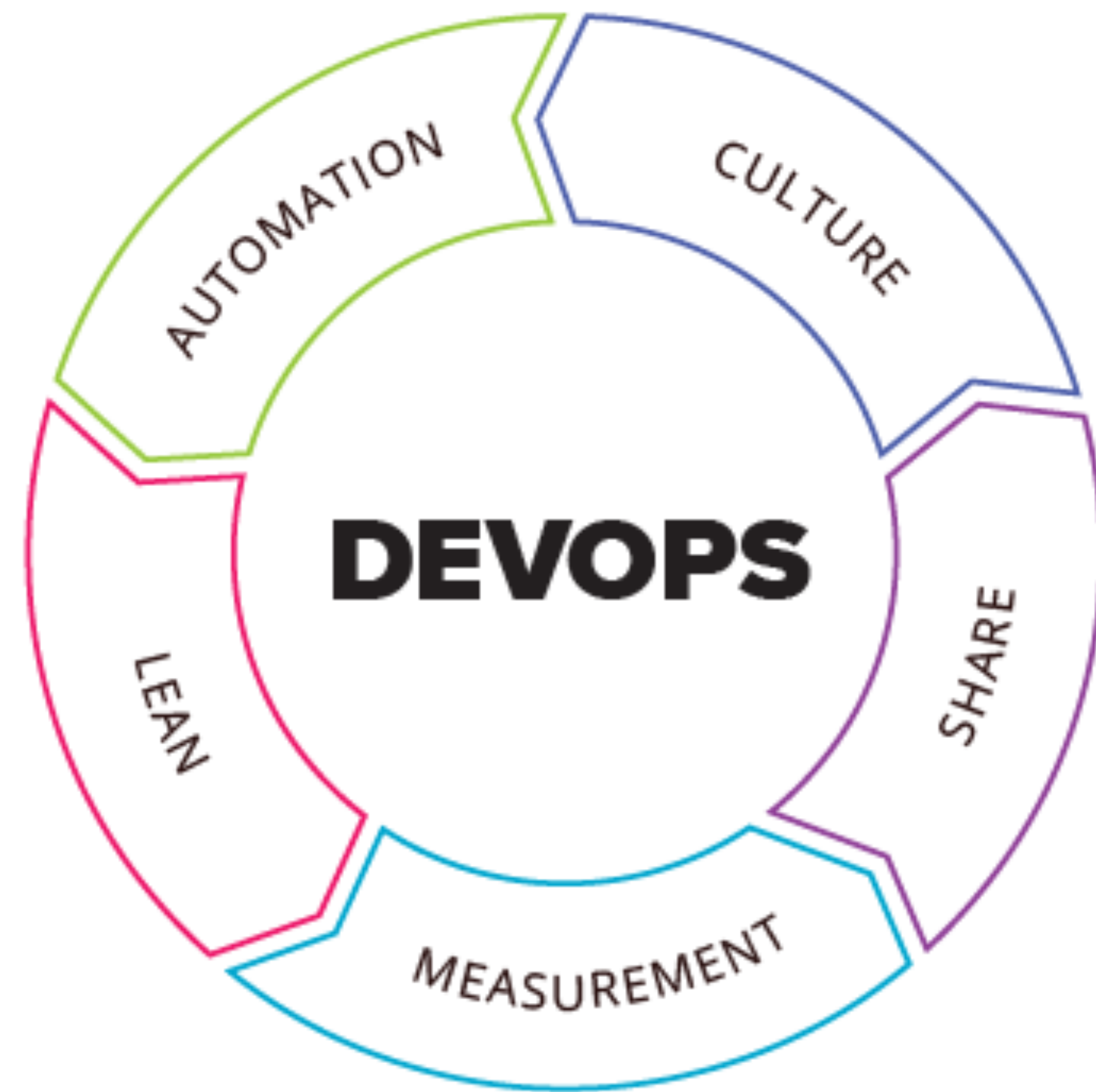
- Тестируем продукт, а не задачу
- Разгружаем тестирование через редкие релизы

Итого

- Задач на тестирование нет
- Разработчики страдают
- Тестирование страдает

???





CALMS Model

Culture

- Shared responsibility
- Embrace Change

Automation

- CI/CD
- Infrastructure as Code

Lean

- Focus on producing value for the end-user
- Small Batch Sizes

Measurement

- Measure everything
- Show the improvement

Sharing

- User-friendly communication channels between development and operations.

LOOK, I BOUGHT A
BOX OF DEVOPS



YOU CAN'T JUST BUY A
BOX OF DEVOPS



@acloudguru



LOOK, HE BOUGHT A
BOX OF DEVOPS



WOW!

COOL!

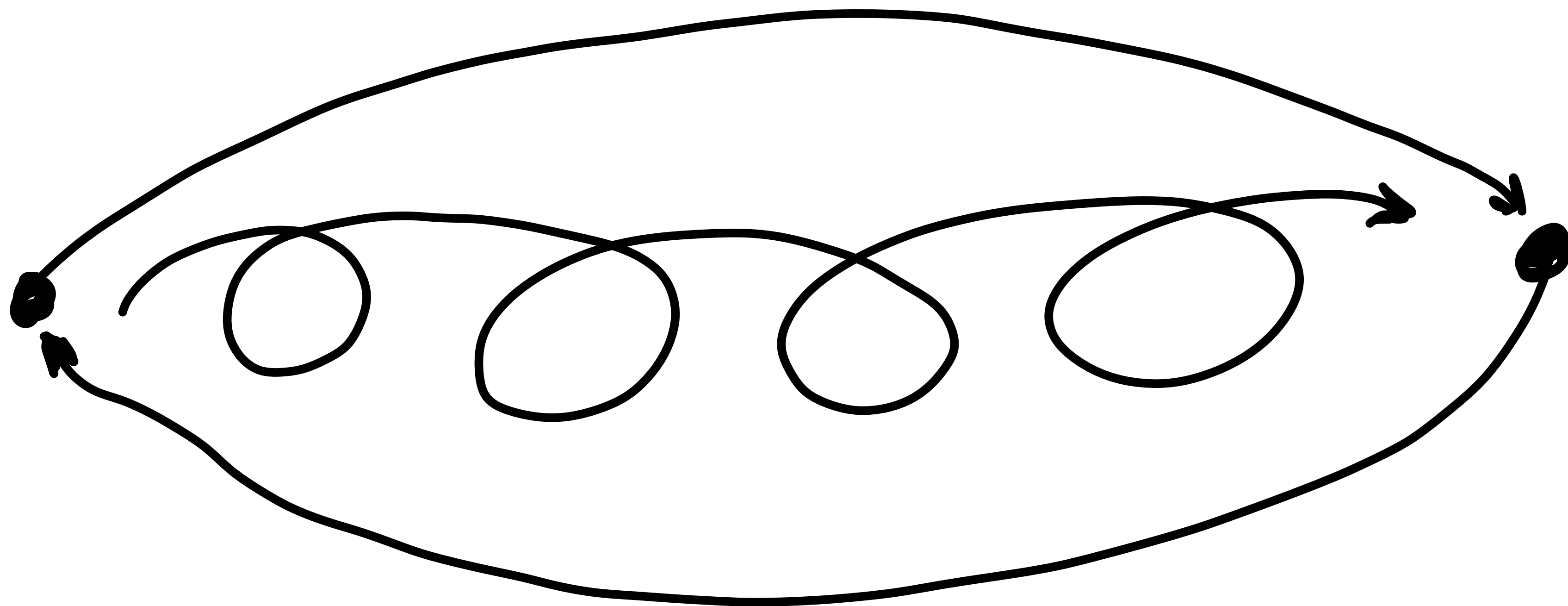
DevOps - это прилагательное

Можно быть DevOps

Три пути

- Непрерывный поток создания ценности
- Поток непрерывной обратной связи
- Поток непрерывного обучения

Три пути



Проблемы

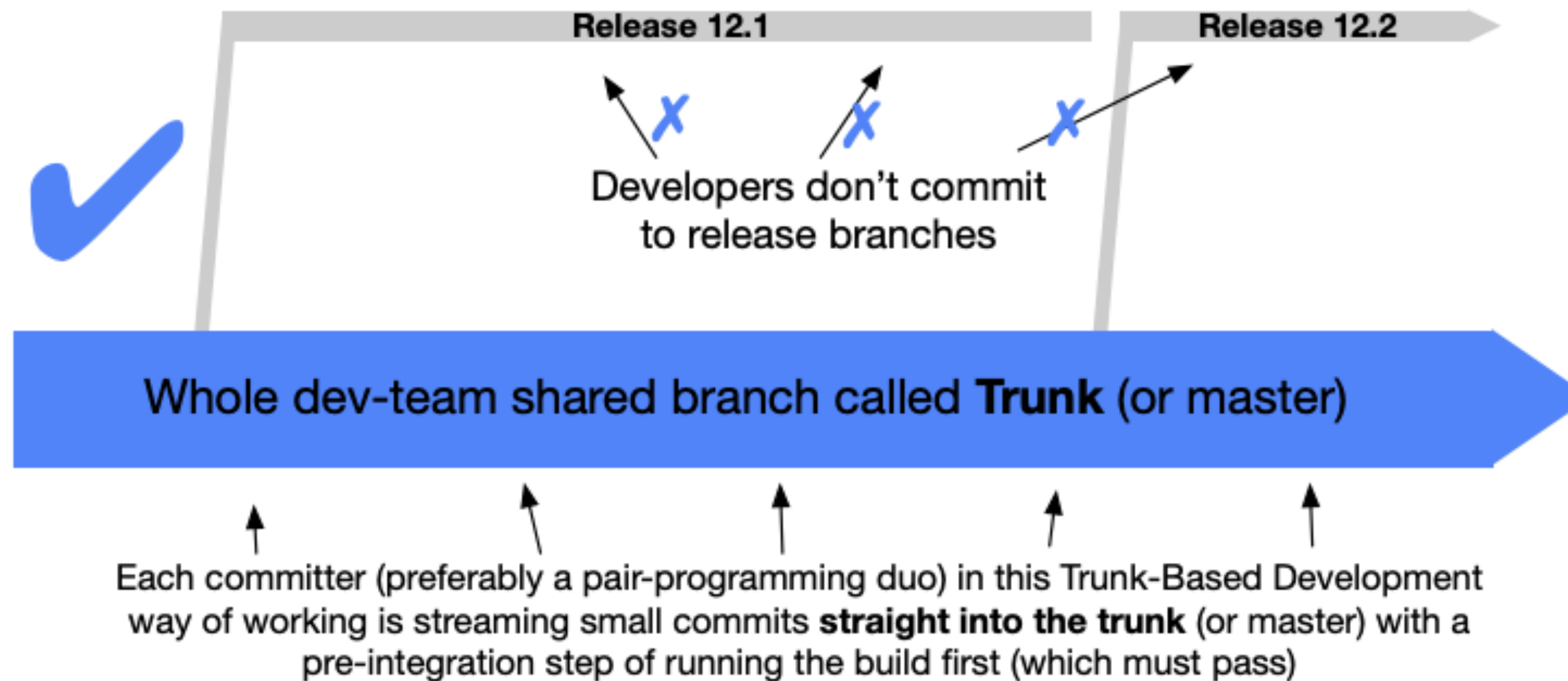
- Нет продуктовых команд
- Мало авто-тестов
- Gitflow / Feature Branches
- SCRUM / Kanban
- Нет CI/CD, хотя есть CI/CD Pipelines
- Нет Canary, Monitoring, etc...



DevOps

DevOps everywhere

Trunk Based Development



Когда пришло время



Пушить в мастер



Wikipedia

In [software engineering](#), **continuous integration (CI)** is the practice of merging all developers' working copies to a shared [mainline](#) several times a day. [Grady Booch](#) first proposed the term CI in [his 1991 method](#), although he did not advocate integrating several times a day. [Extreme programming \(XP\)](#) adopted the concept of CI and did advocate integrating more than once per day – perhaps as many as tens of times per day

Мифы про Git

Мержить в мастер надо только готовый код, причём степень и критерии готовности - это еще и на усмотрение заблуждающегося

Код надо выкладывать релизами

Под каждую версию на проде должен быть отдельный бранч или тэг

Code Review - обязательный и необходимый в любой команде

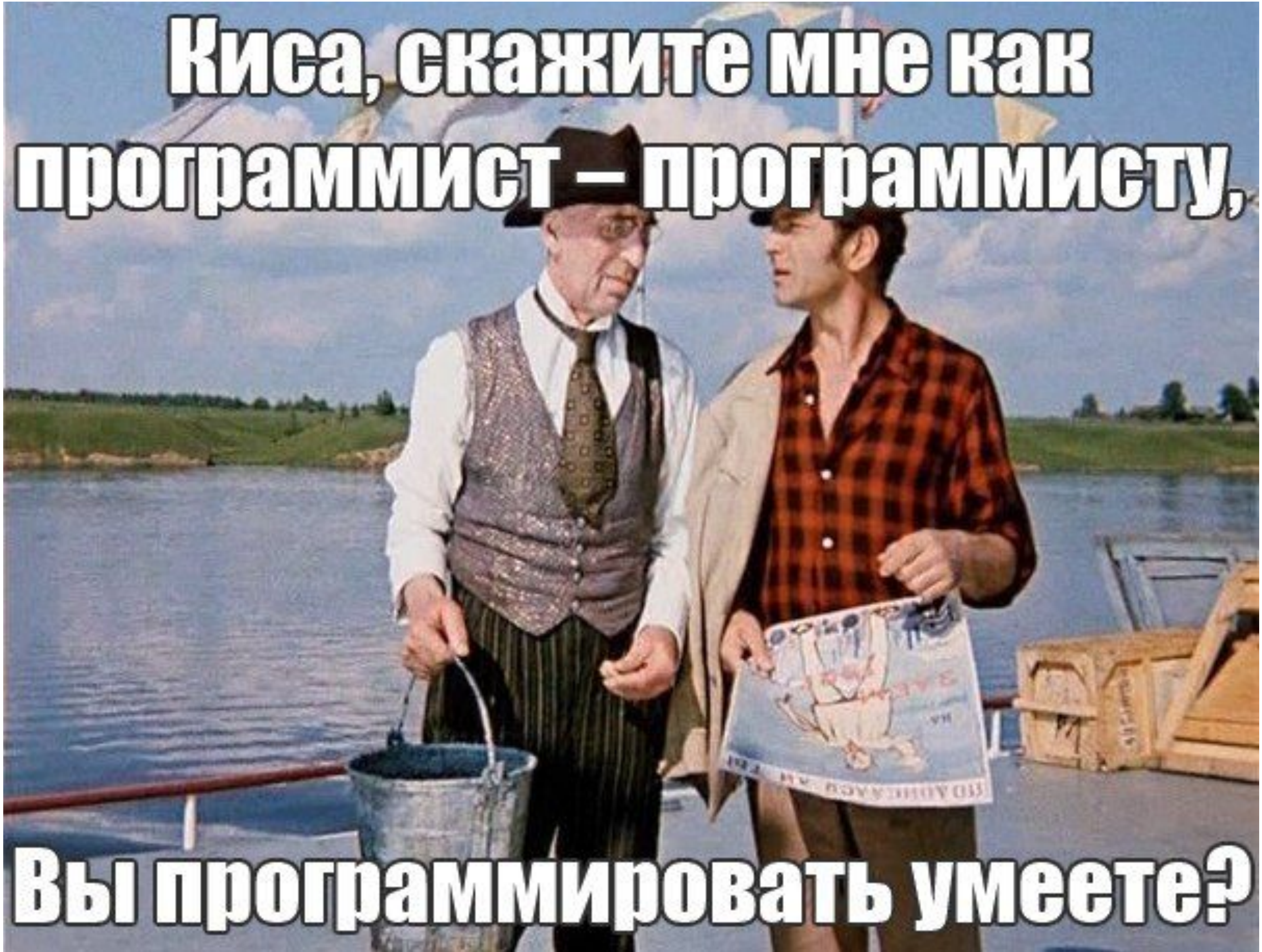
Важно

Этапы проверок и согласований только увеличивают риски сбоев

Что с тестирующими?

**Киса, скажите мне как
программист – программисту,**

Вы программировать умеете?

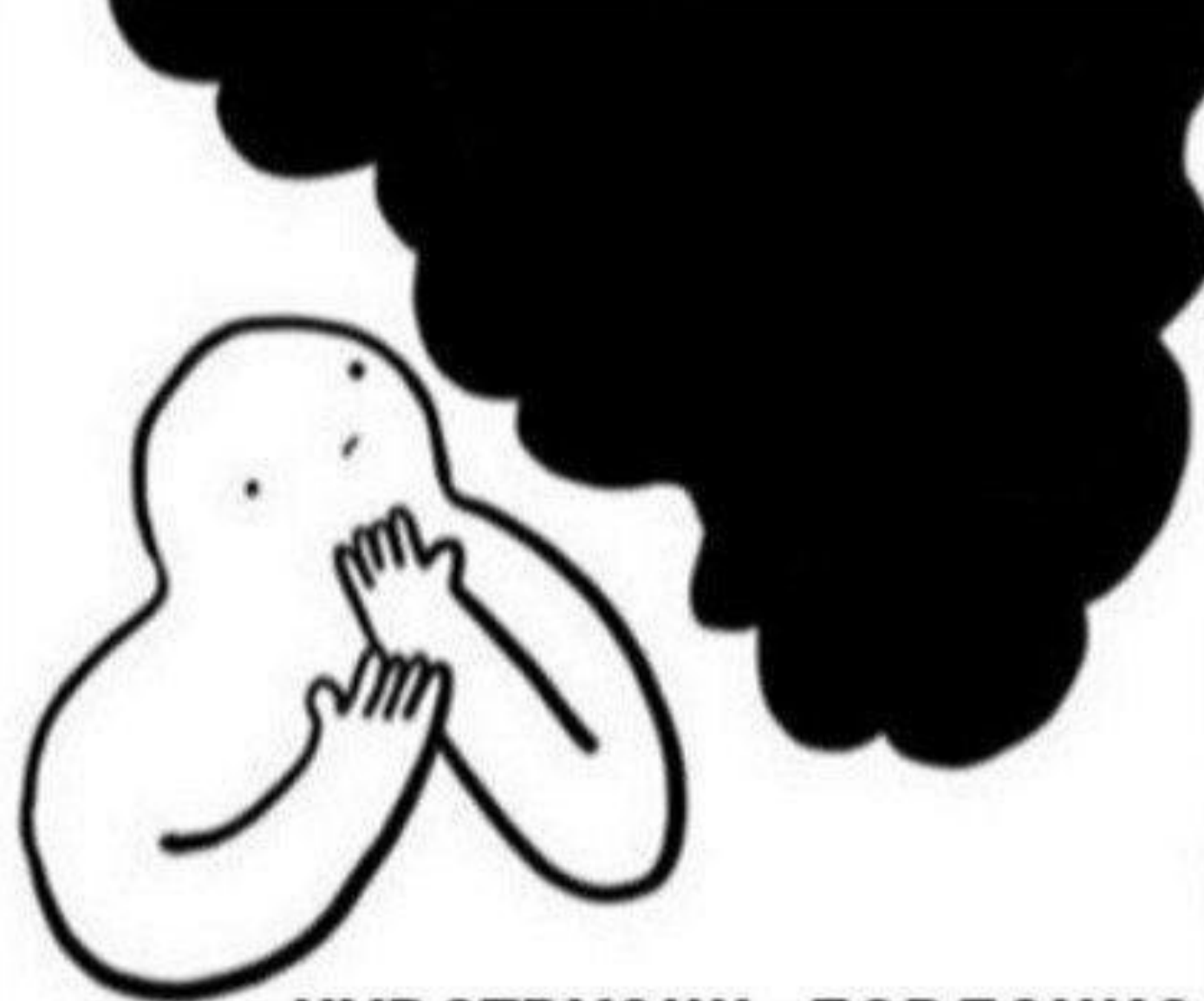


ISO 9001

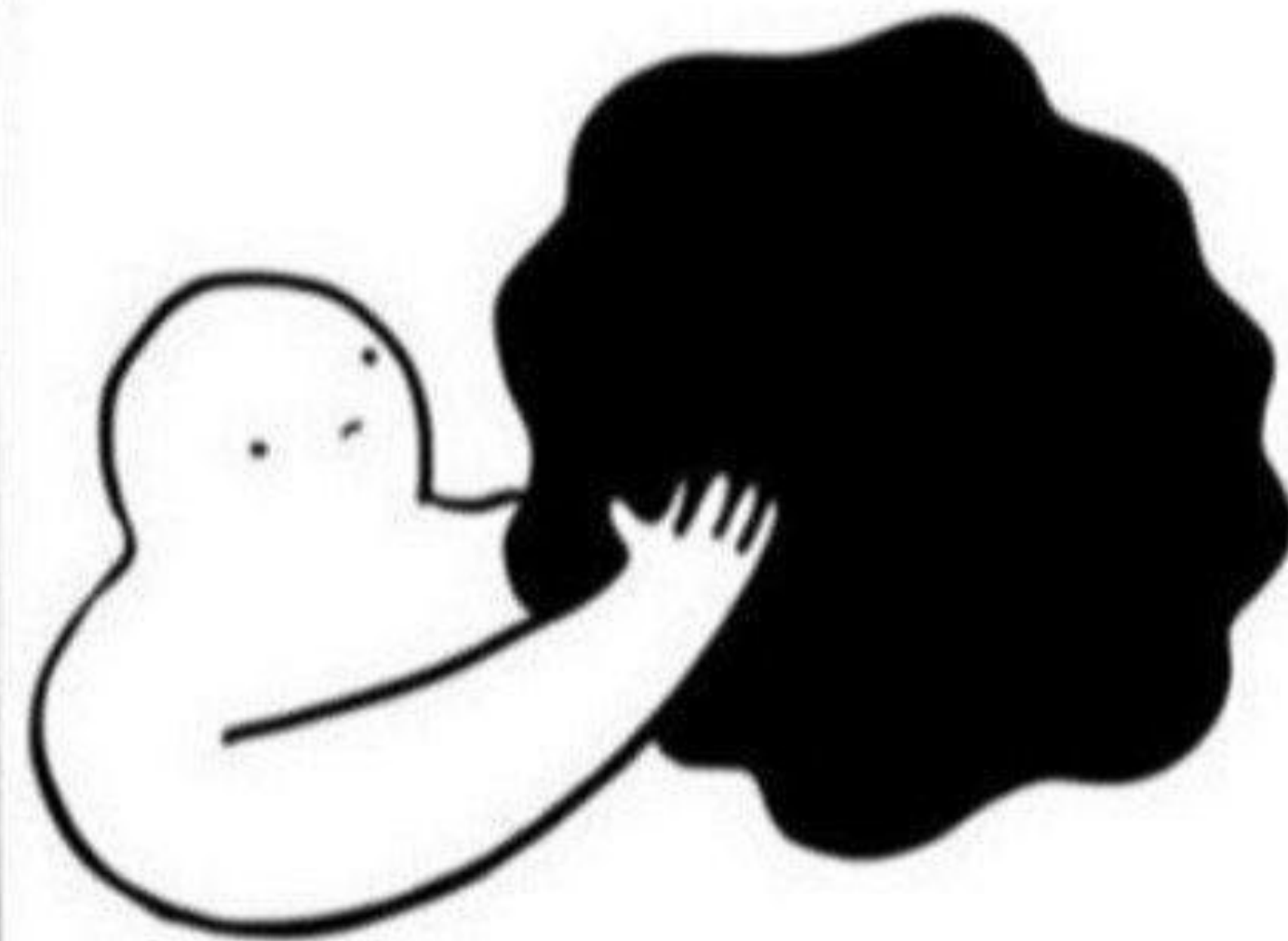
TQM

PDCA

**После привнесения одной
практики, за ней неизбежно
потянутся следующие**



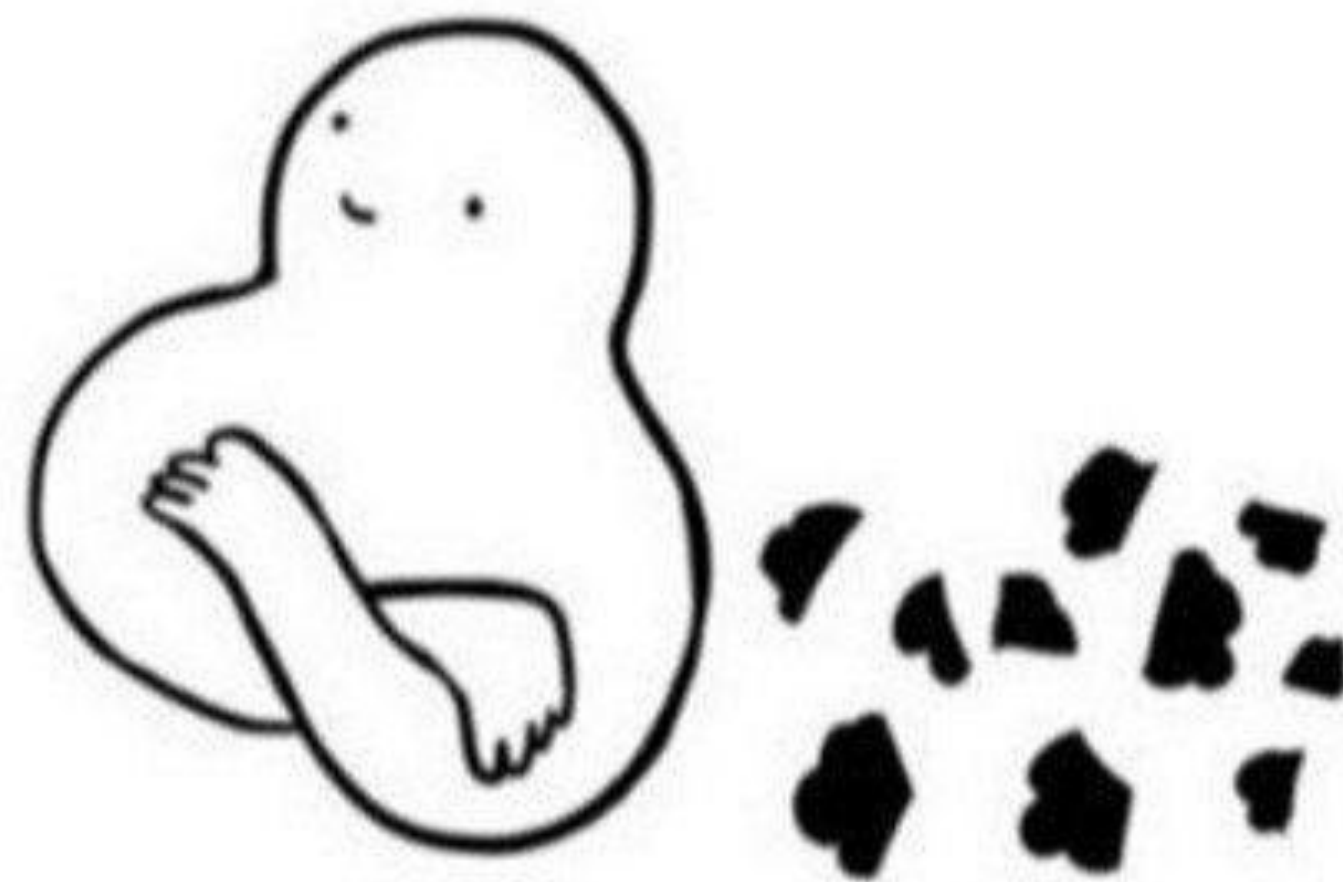
чувствуешь давление
большой задачи?



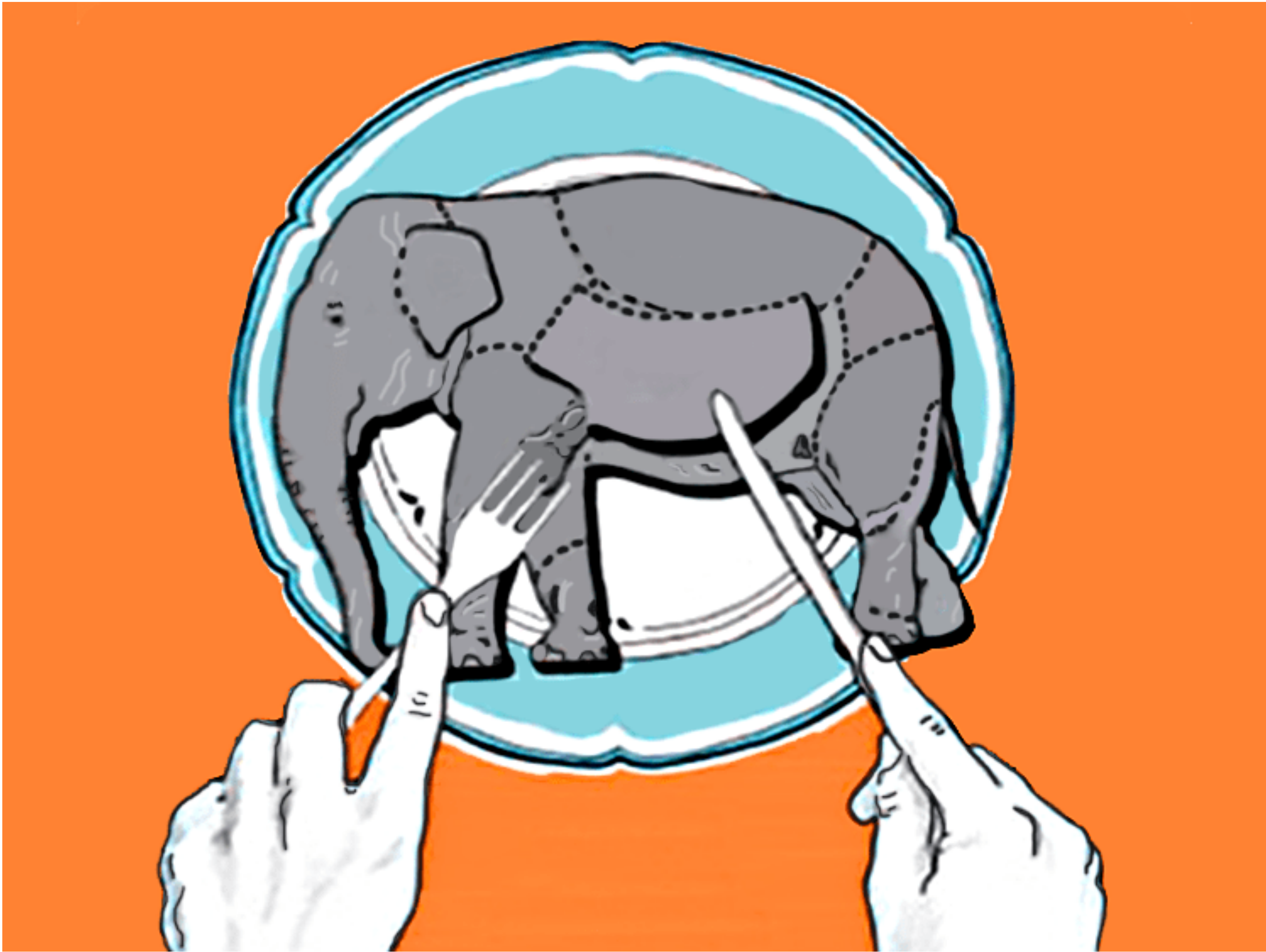
возьми её



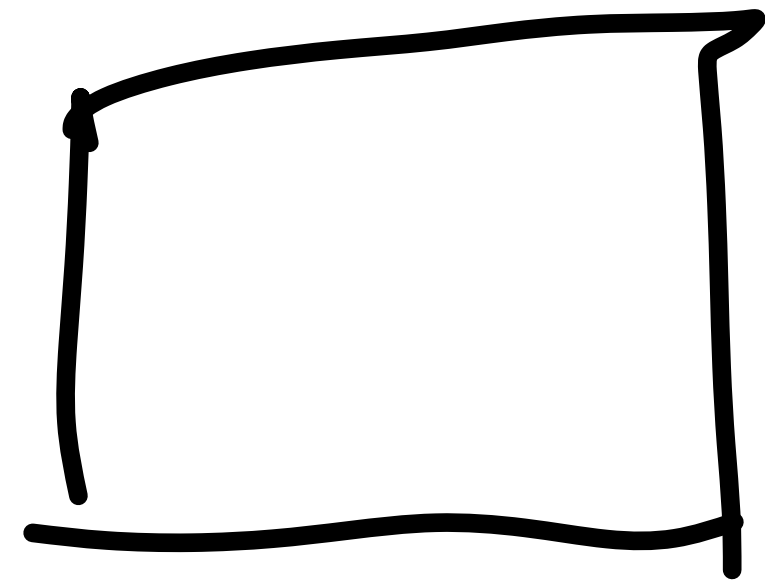
и раздели её на несколько
небольших задач



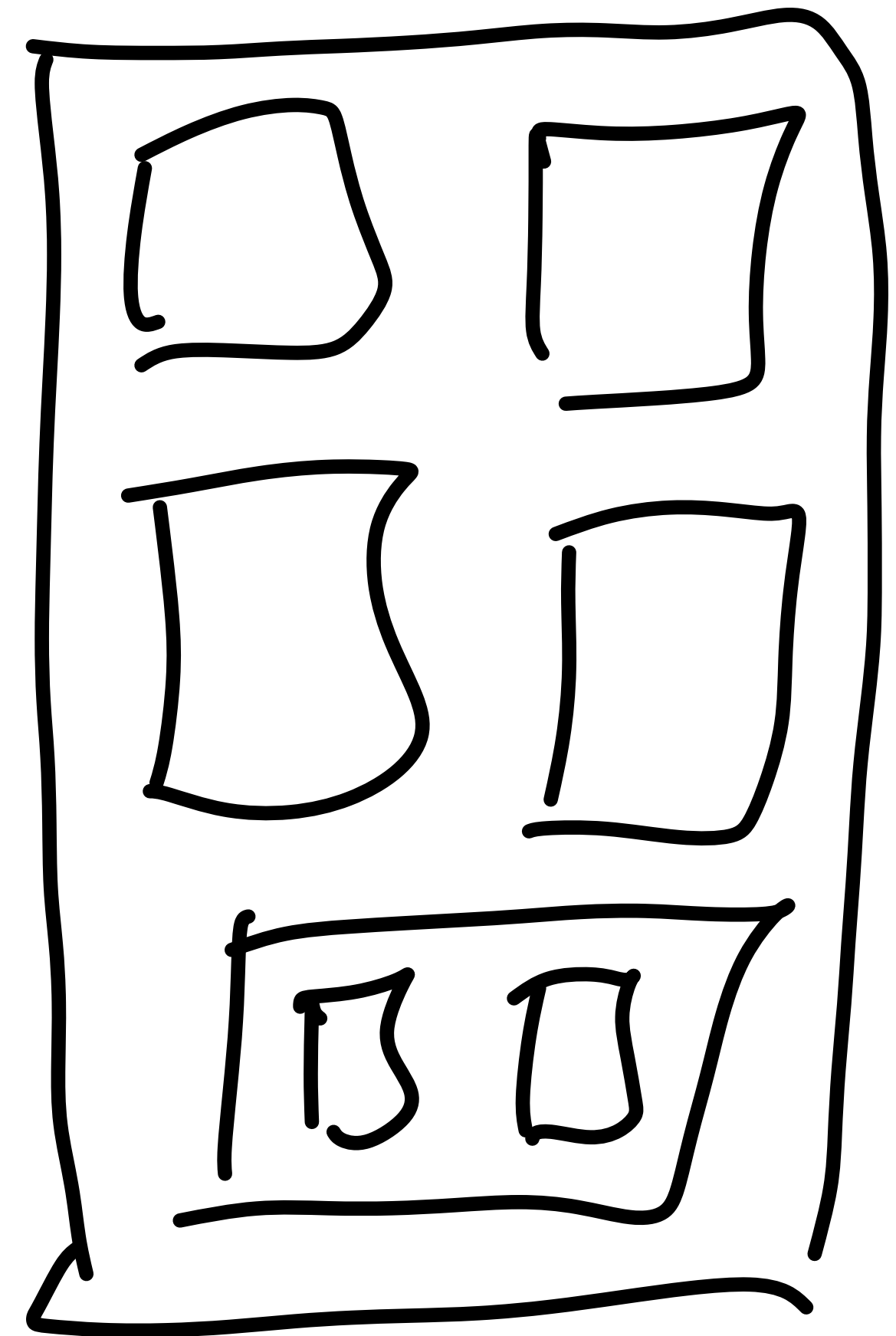
ну вот и всё, маленькие
задачи легче игнорировать

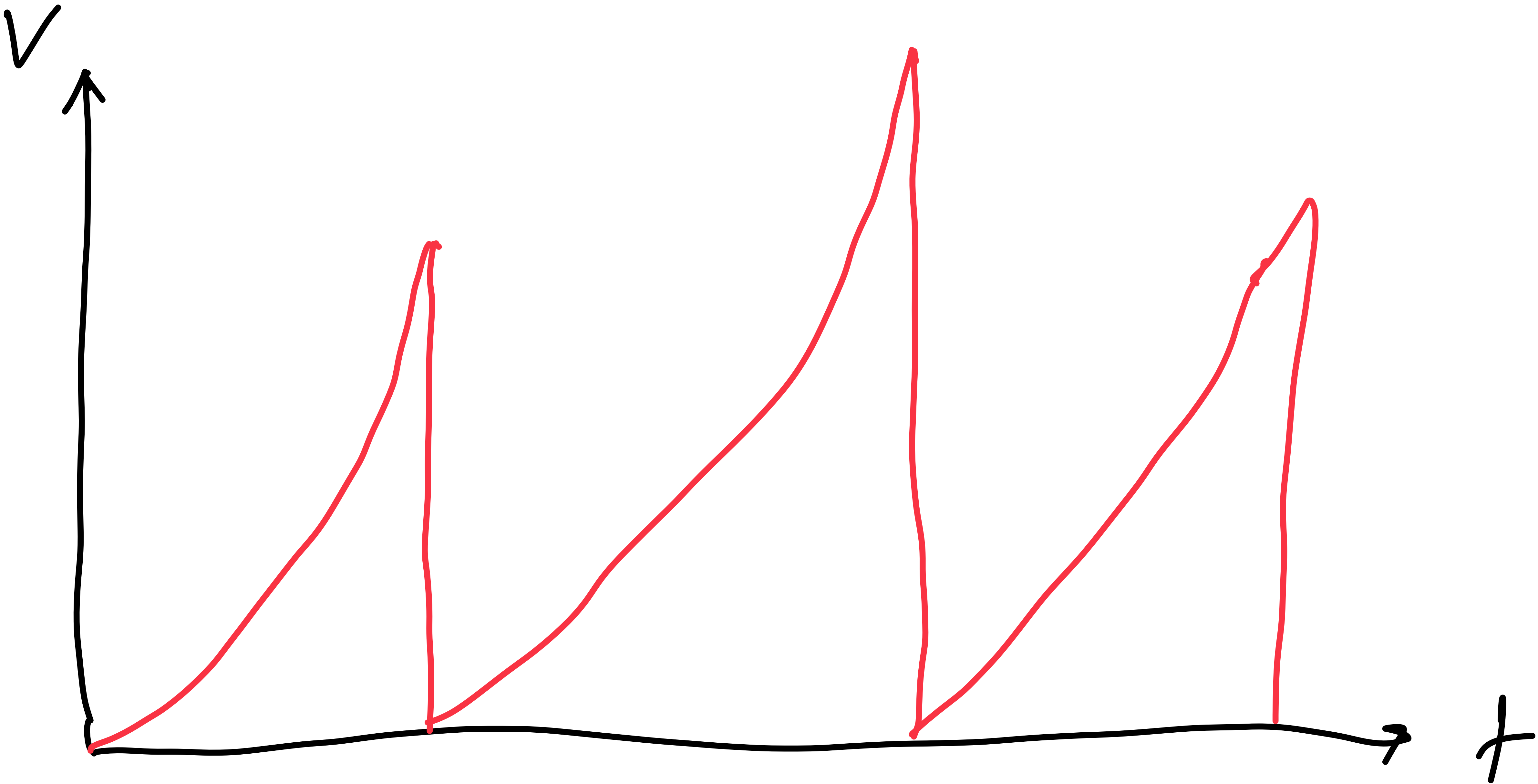


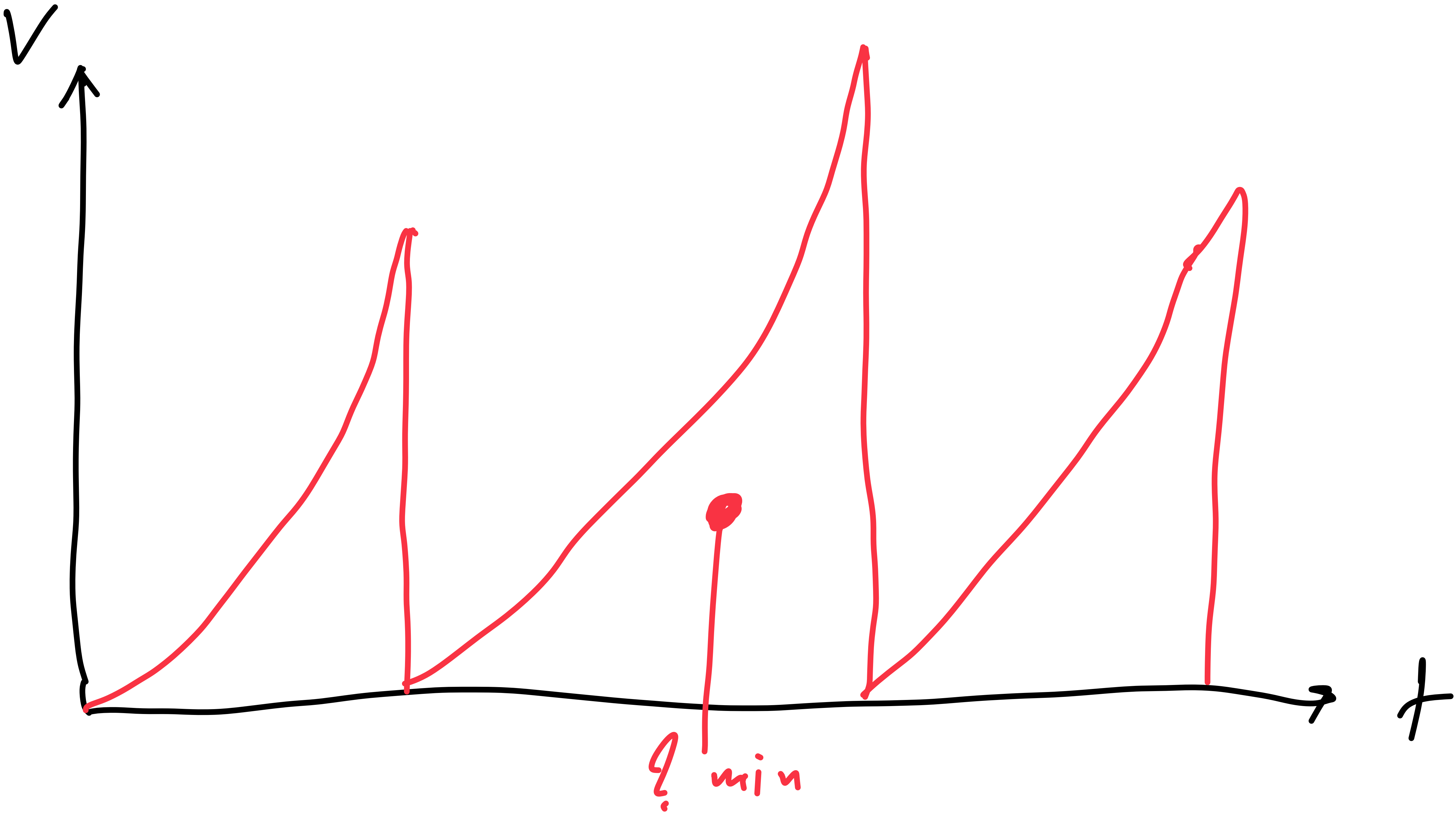
Task ≤ 3

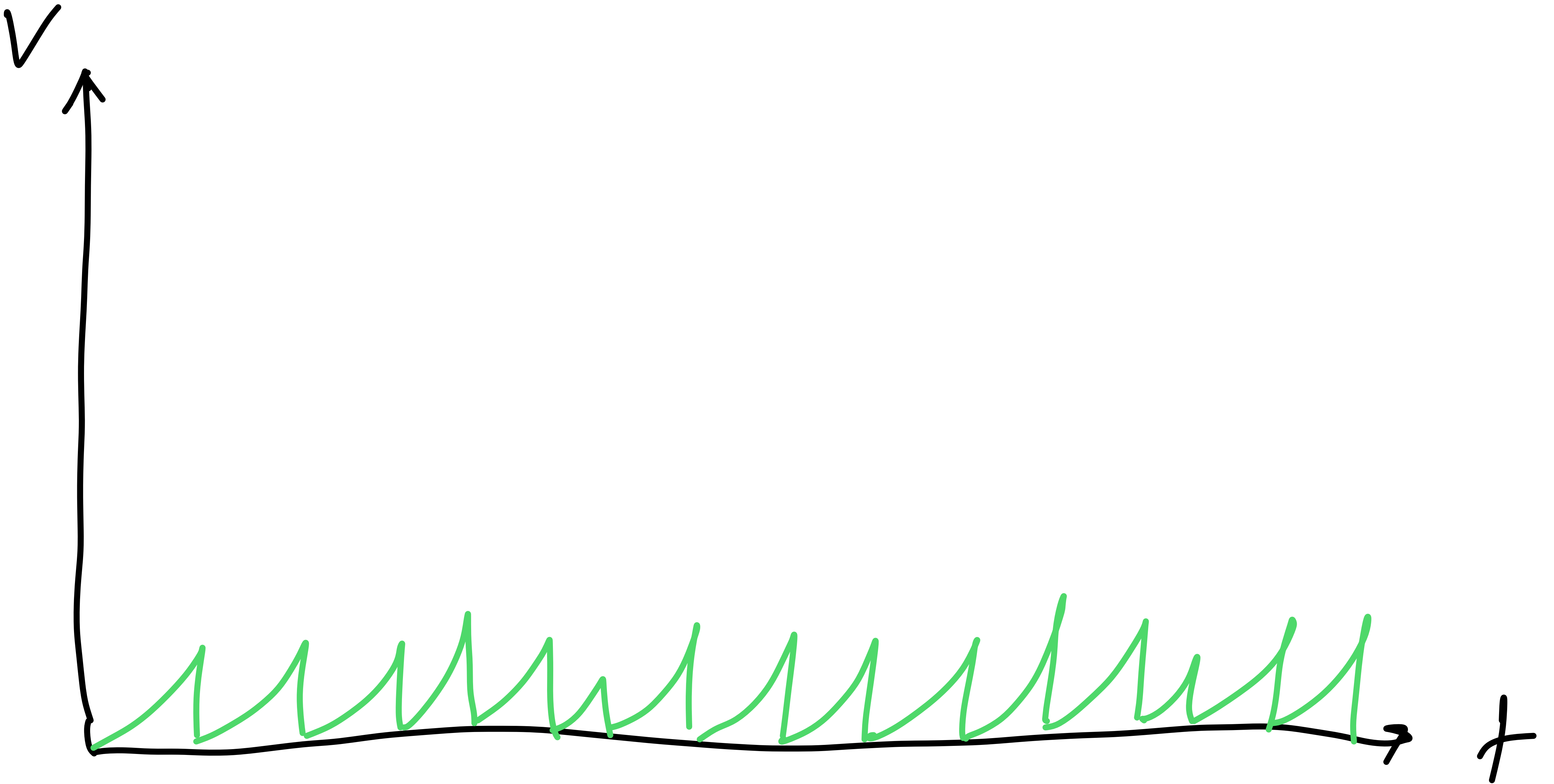


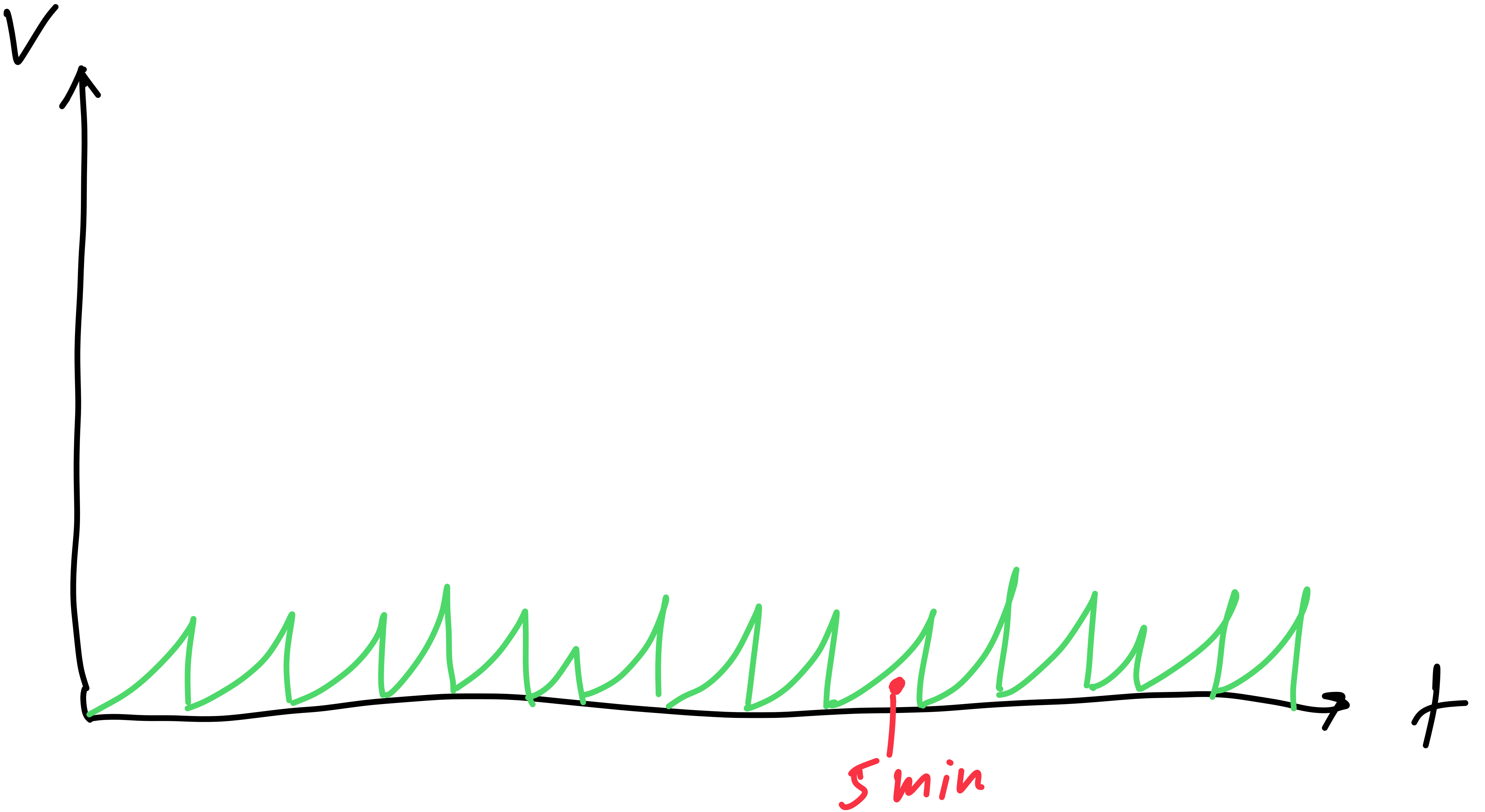
3 \leq Project



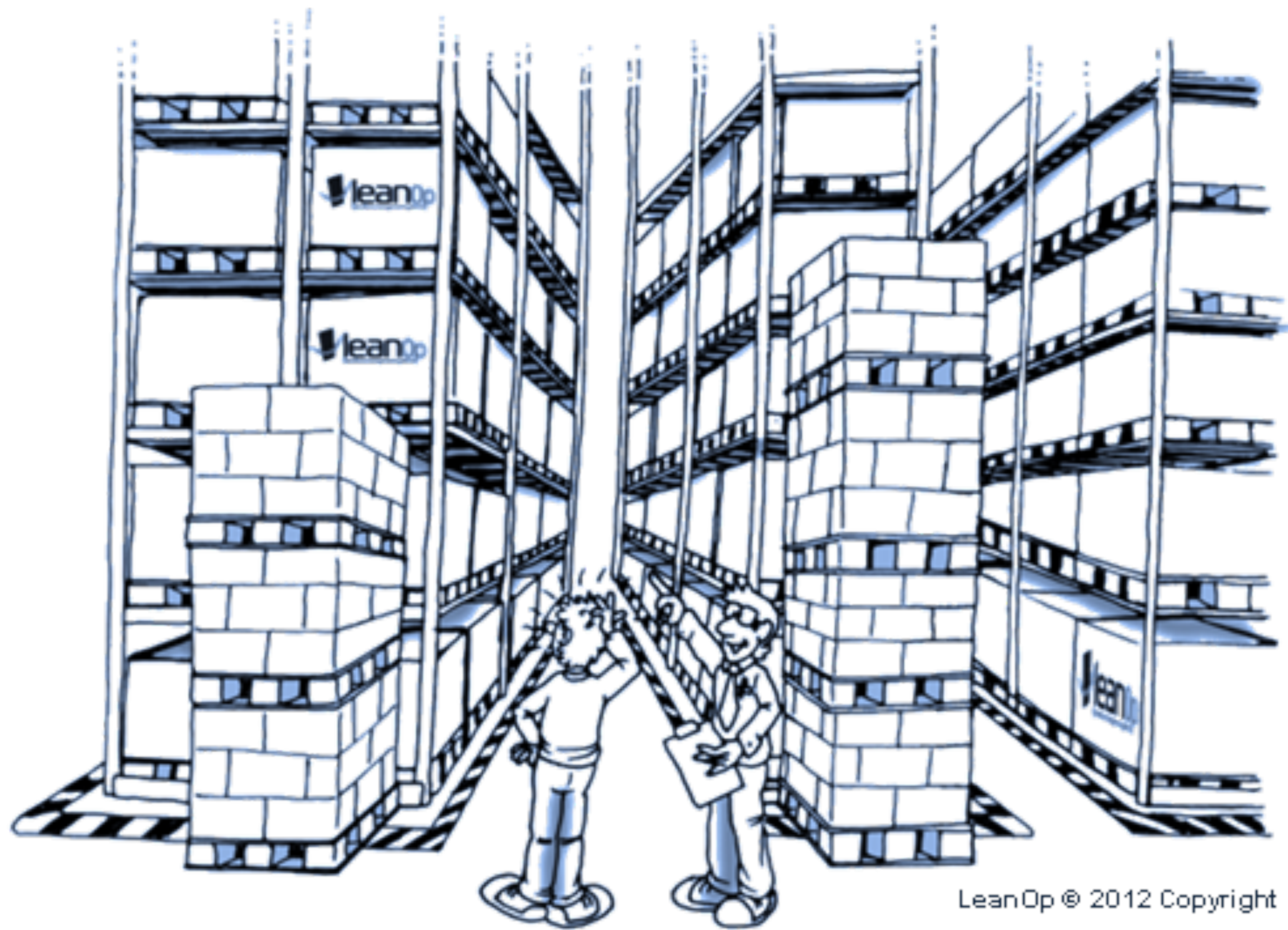




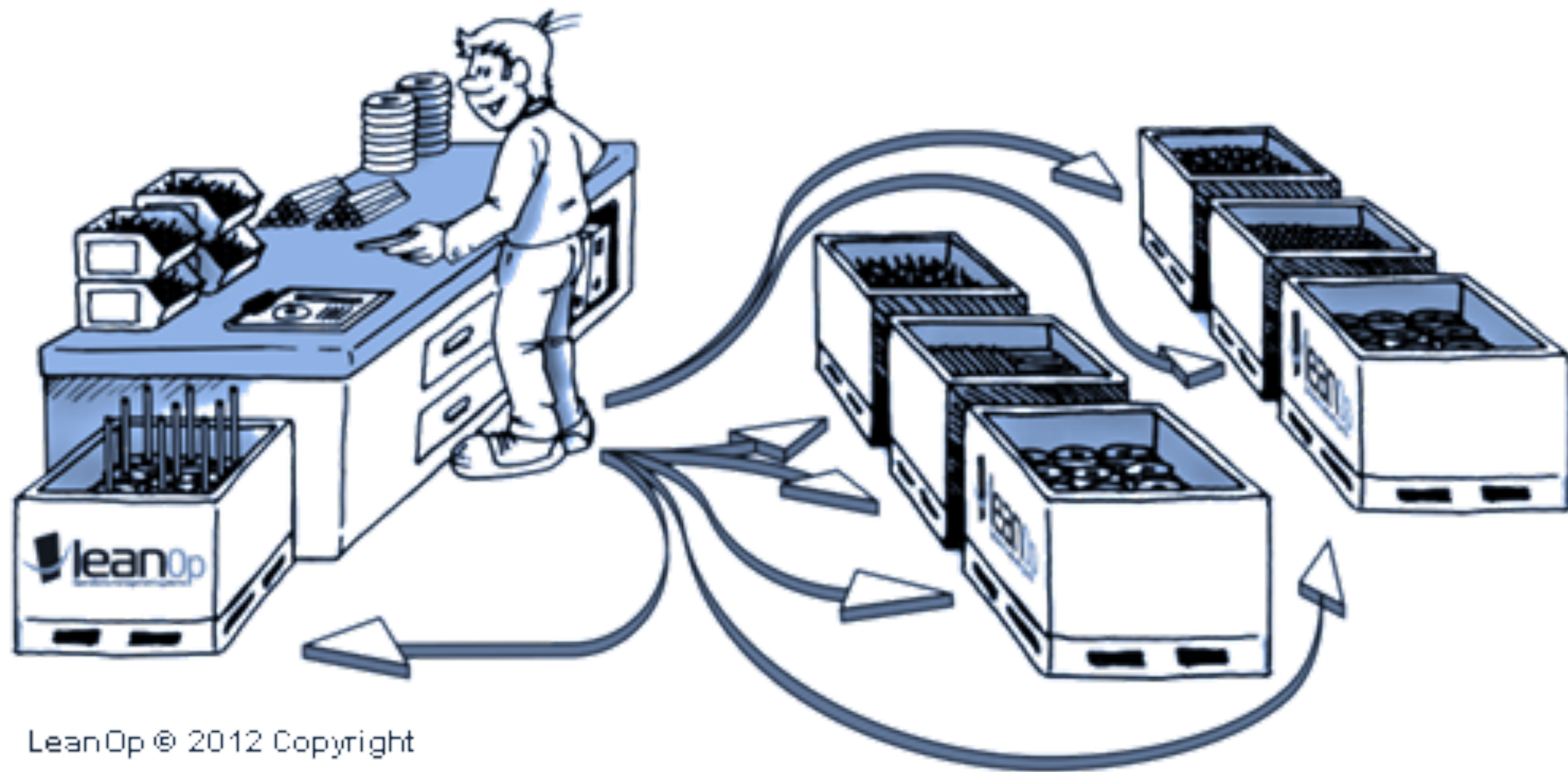




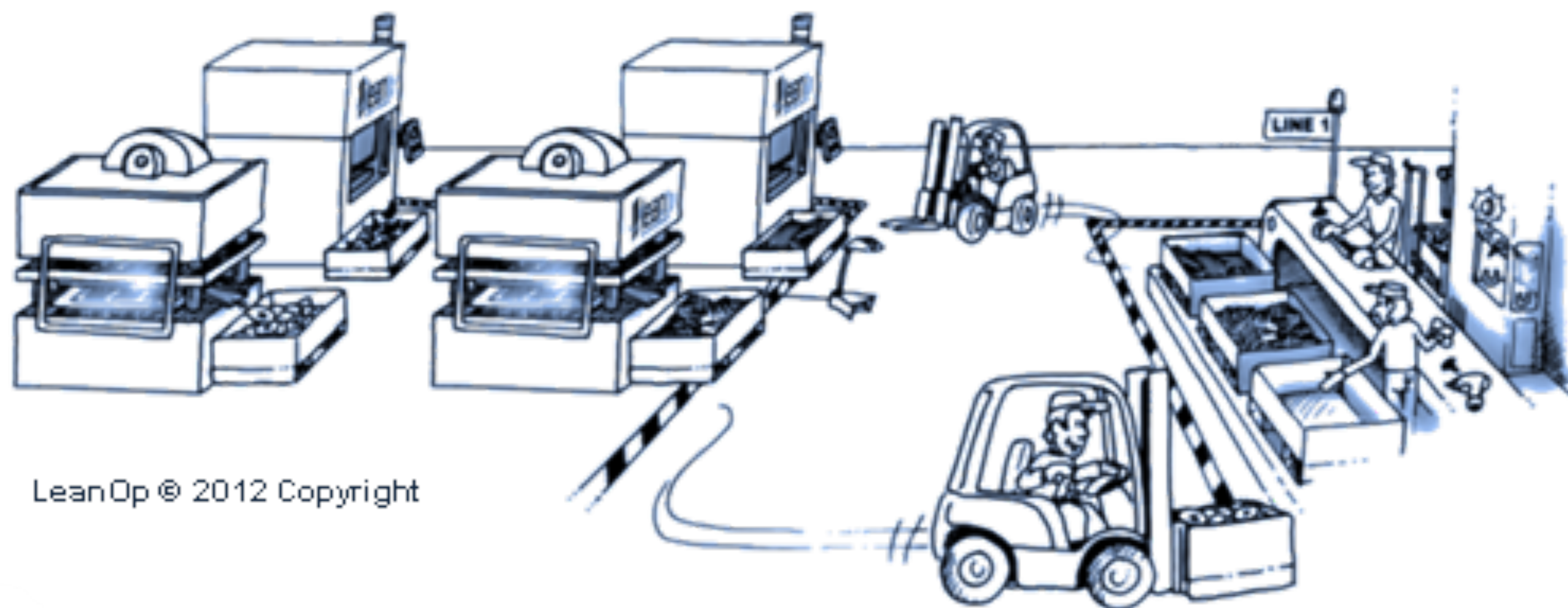
Lean Management



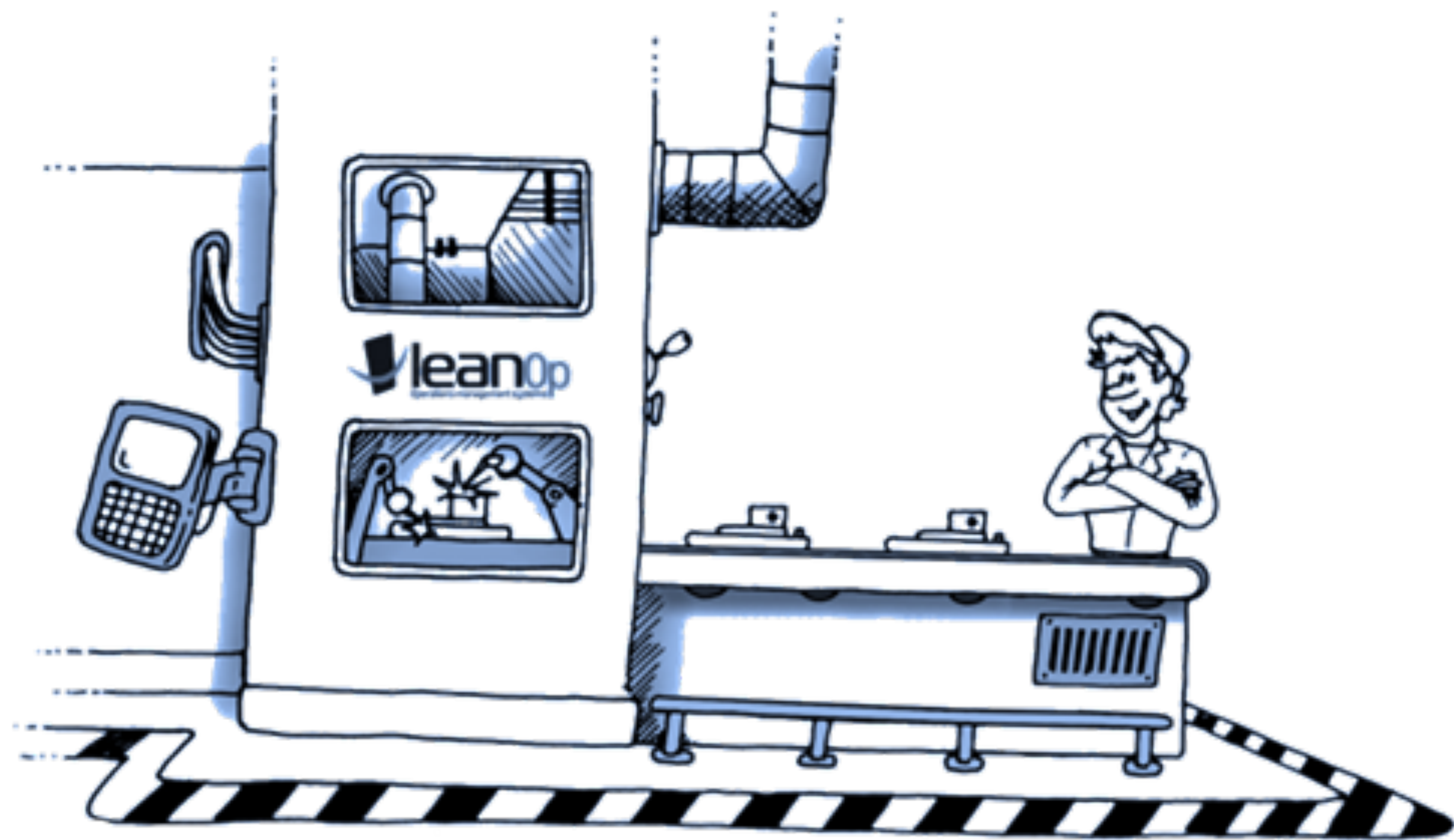
LeanOp © 2012 Copyright



LeanOp © 2012 Copyright



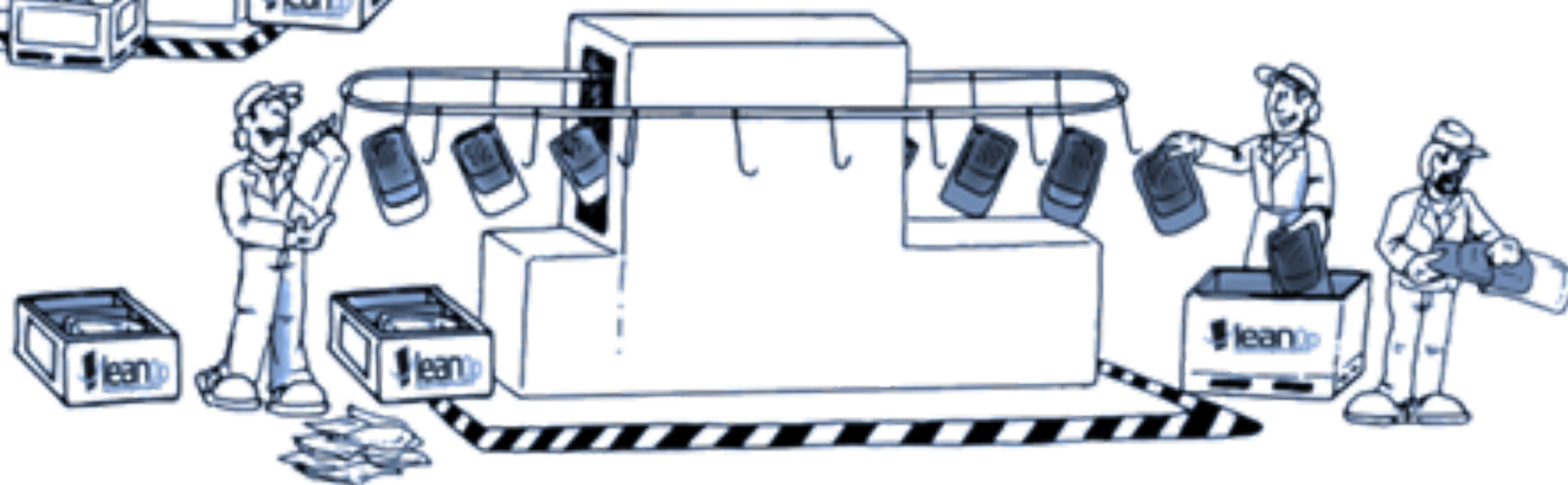
LeanOp © 2012 Copyright



LeanOp © 2012 Copyright



LeanOp © 2012 Copyright

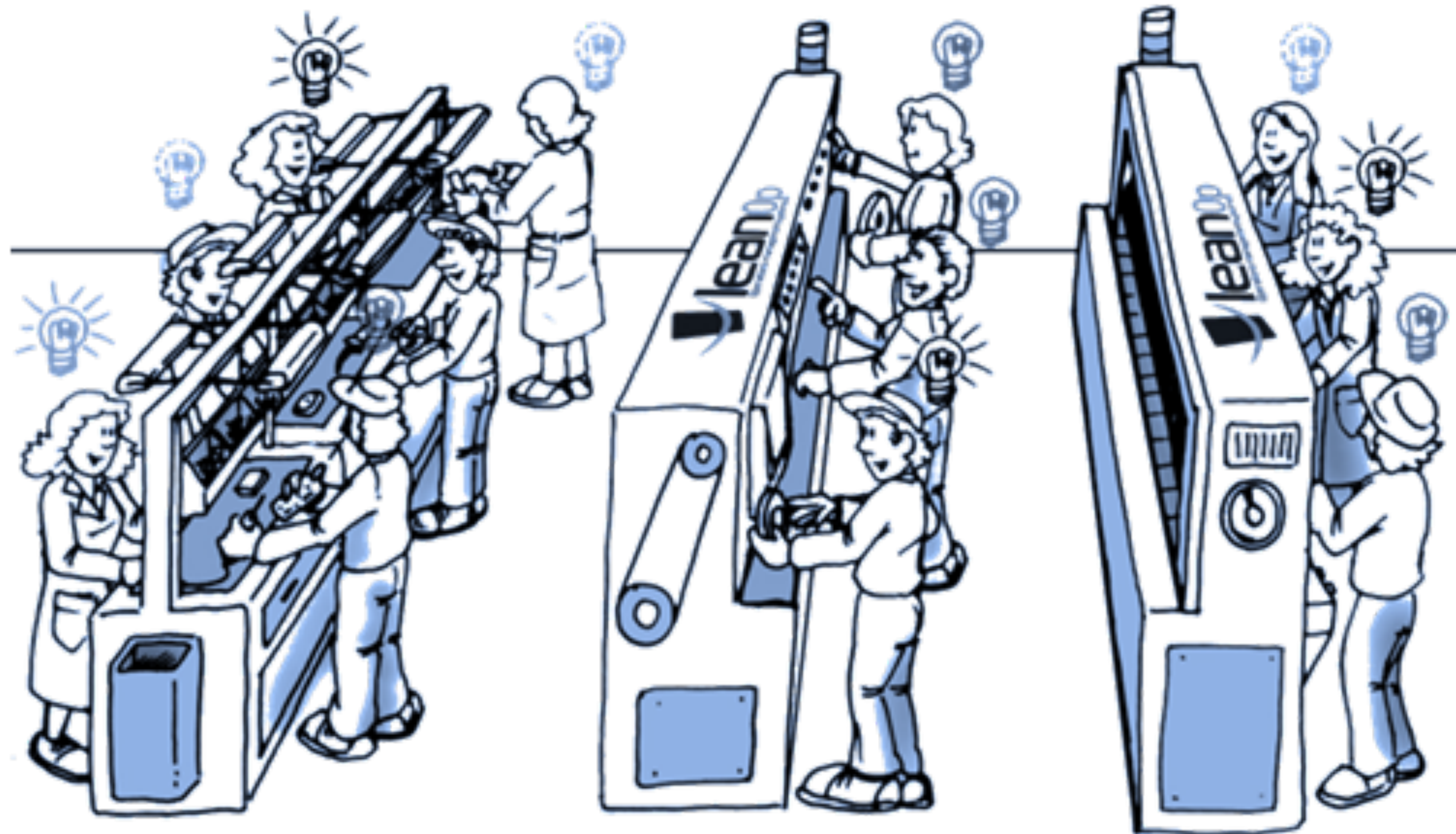




LeanOp © 2012 Copyright

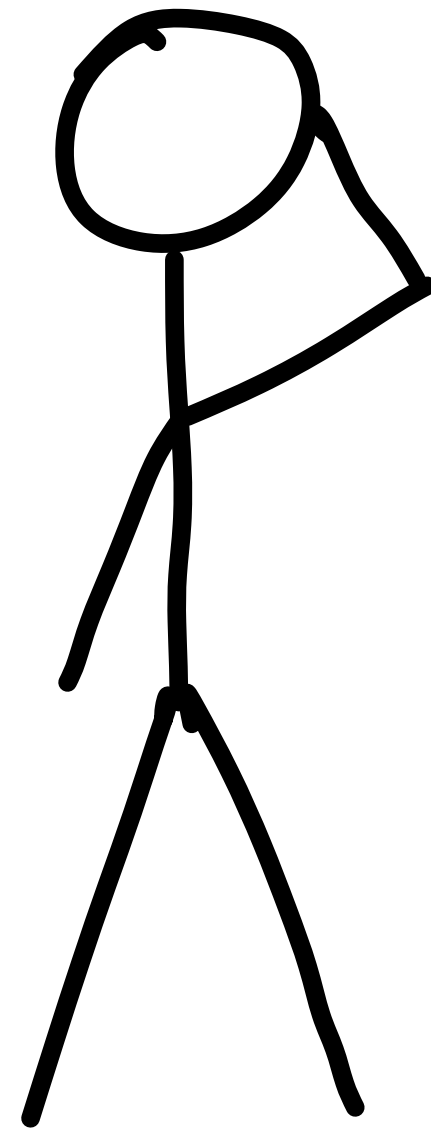


LeanOp © 2012 Copyright



LeanOp © 2012 Copyright

?



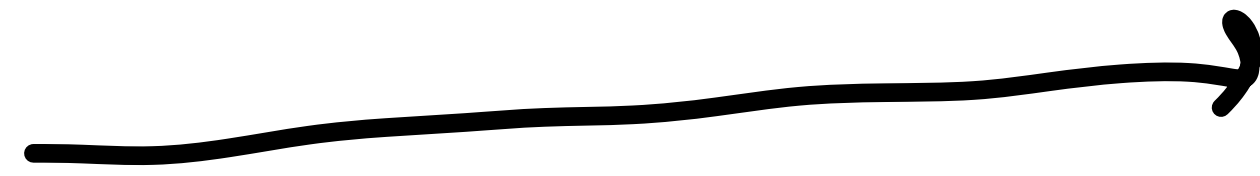
No test

No test

30 min

No test

30 min



48 h

g

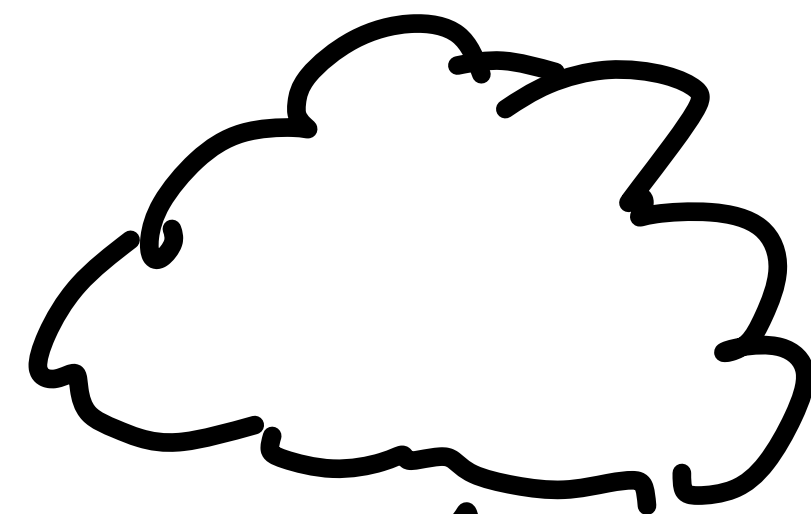
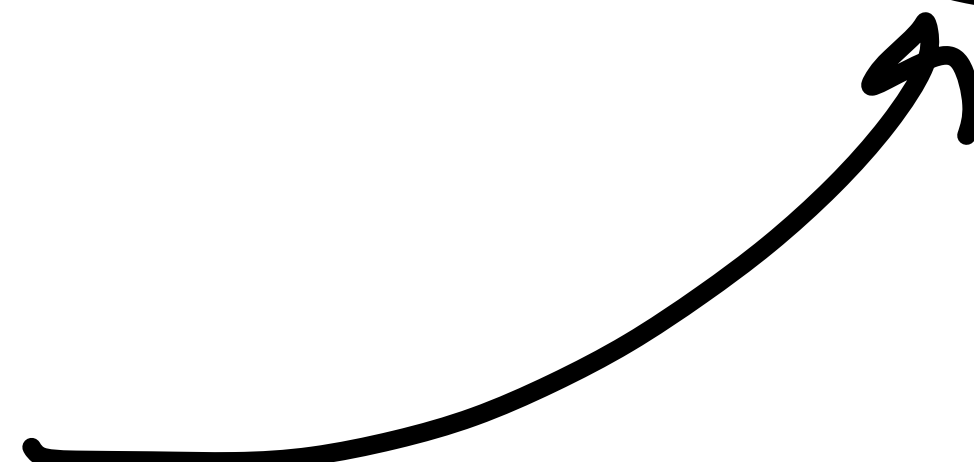
No test

30 min

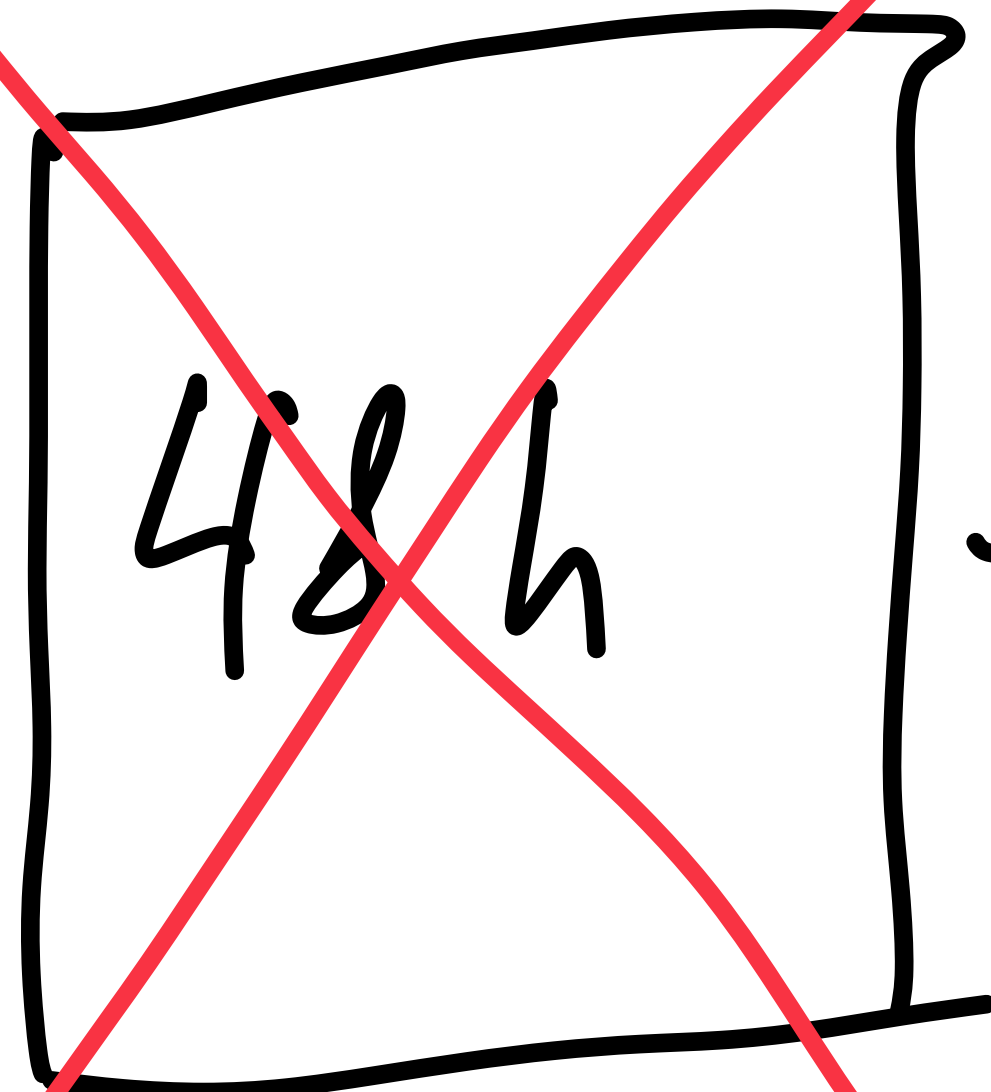
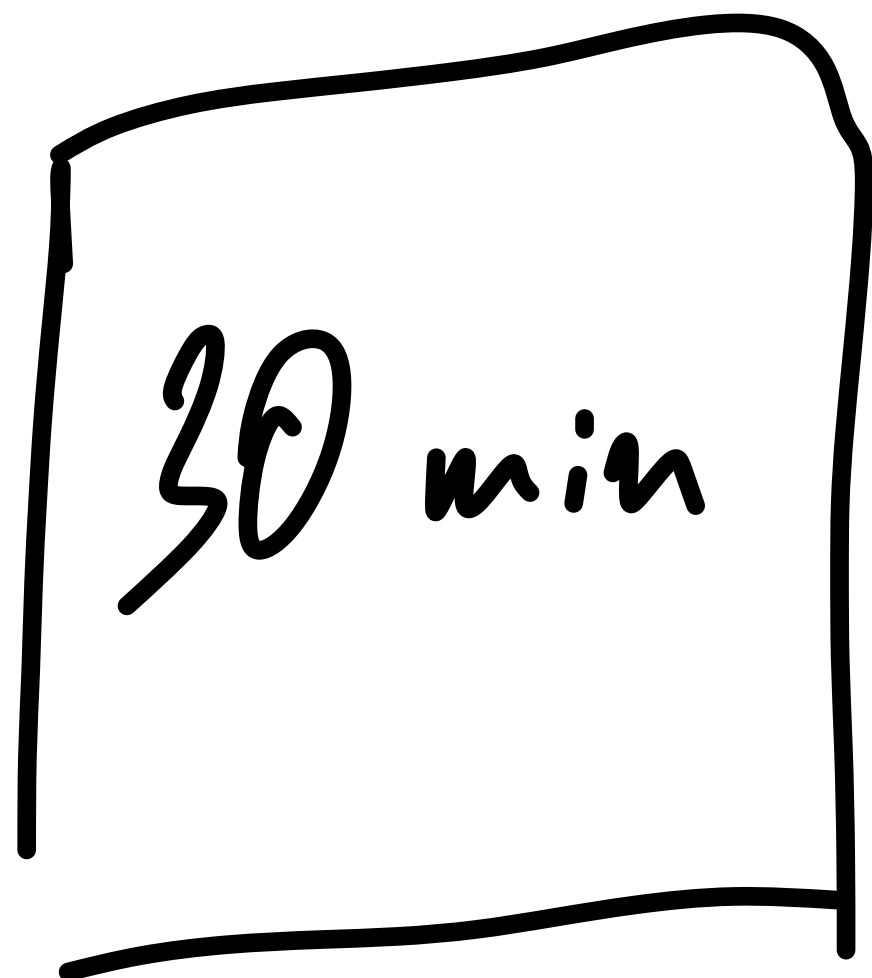


48 h

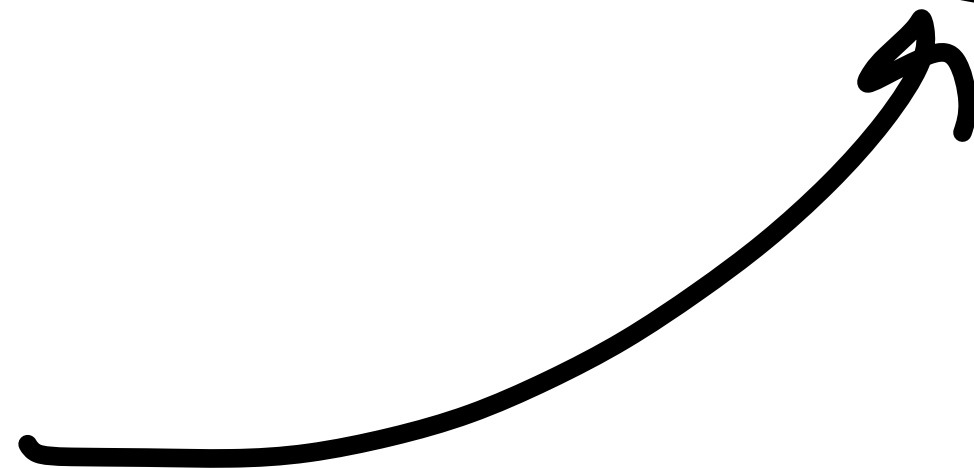
g



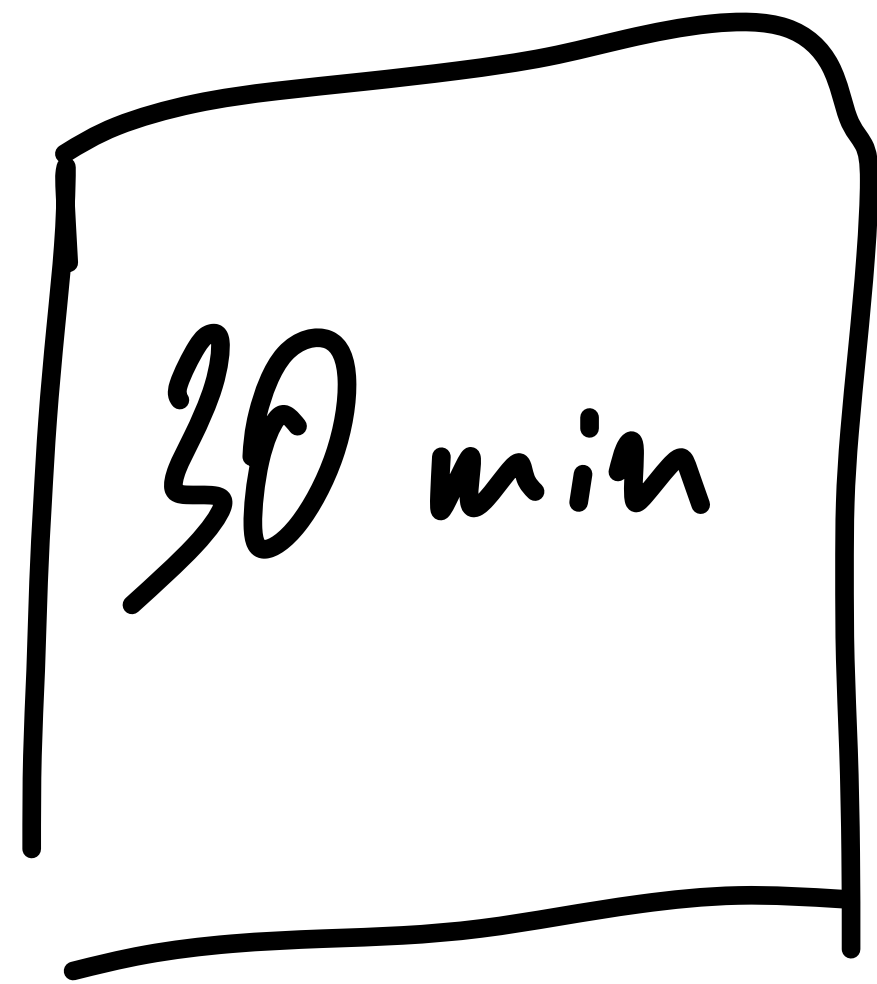
No test



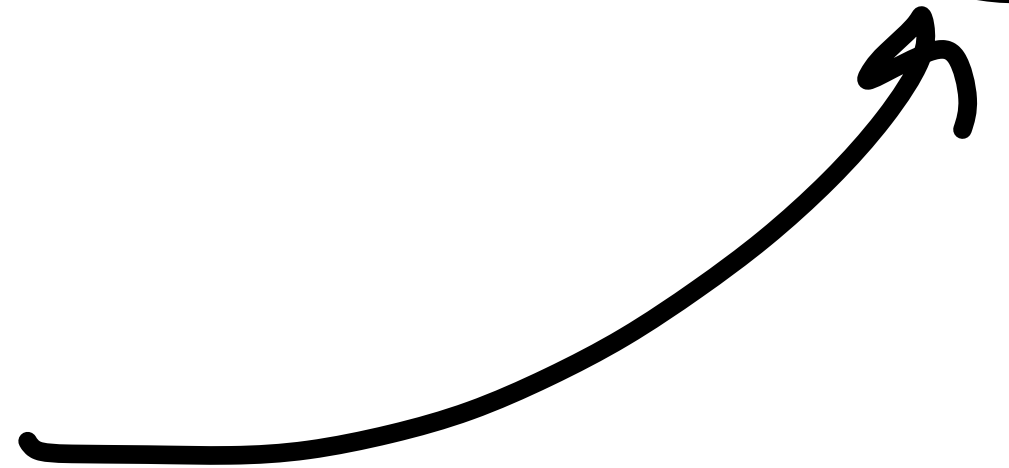
g



No test



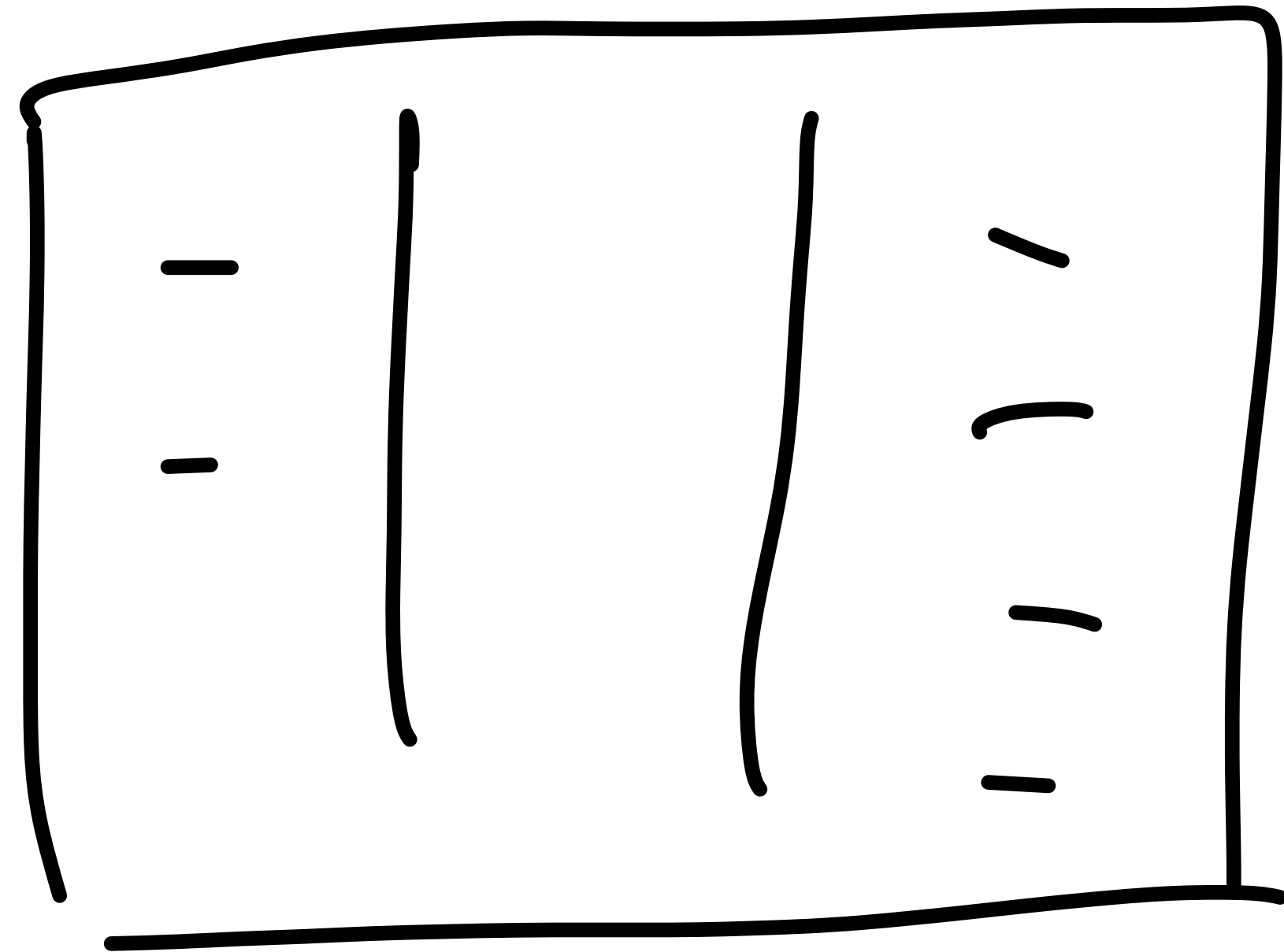
q



x 96

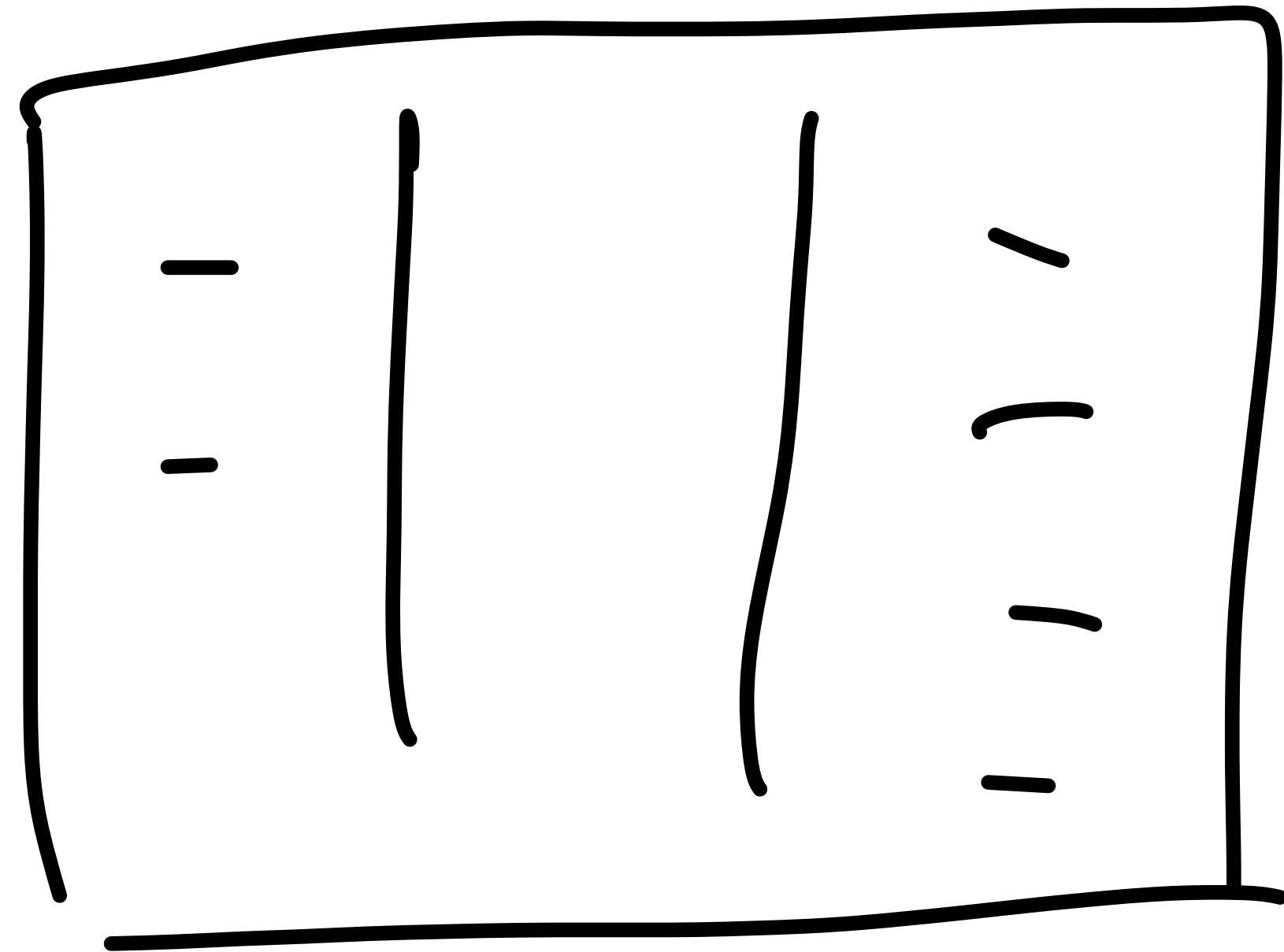
No Backlog

No Backlog

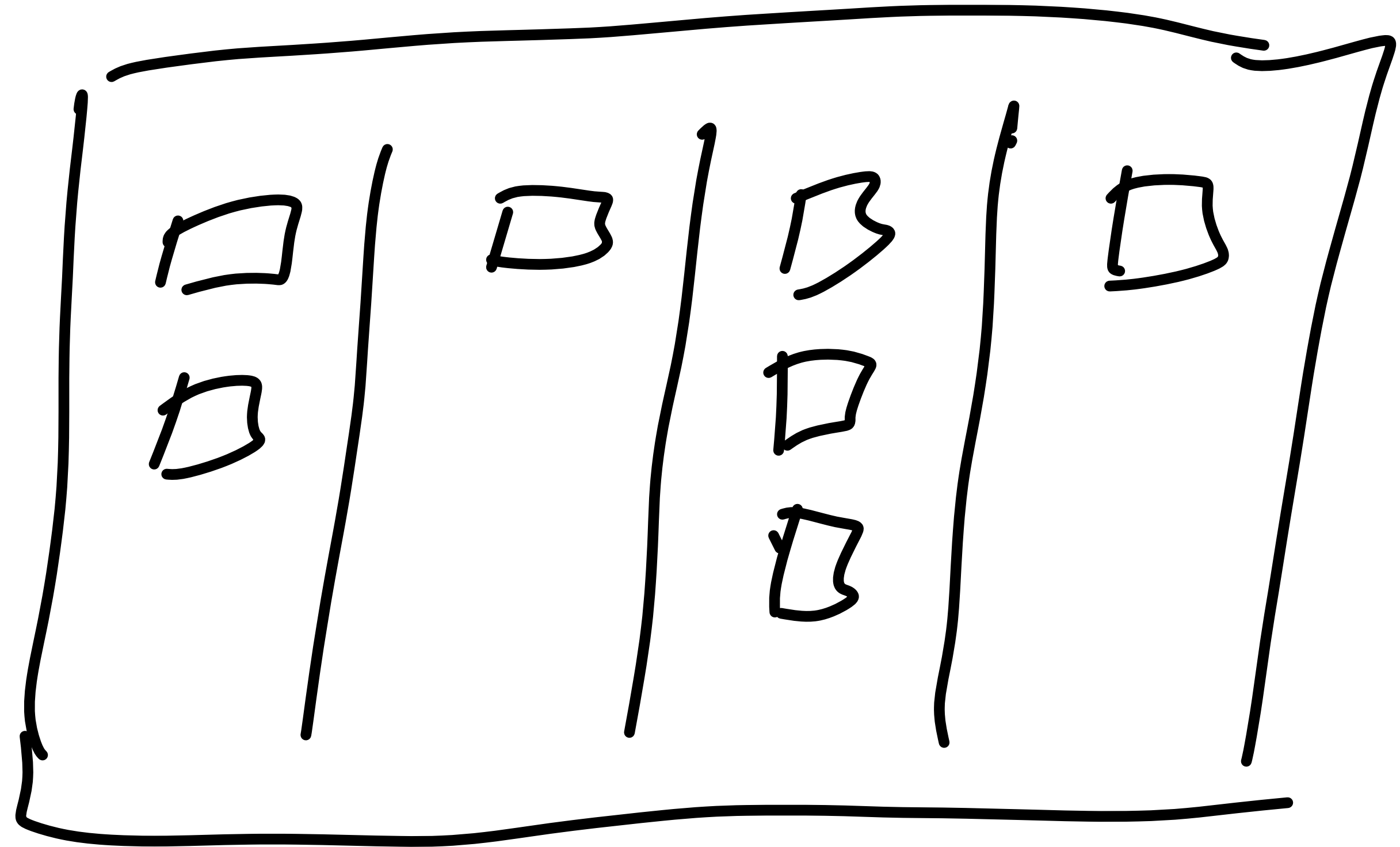


Product

No Backlog

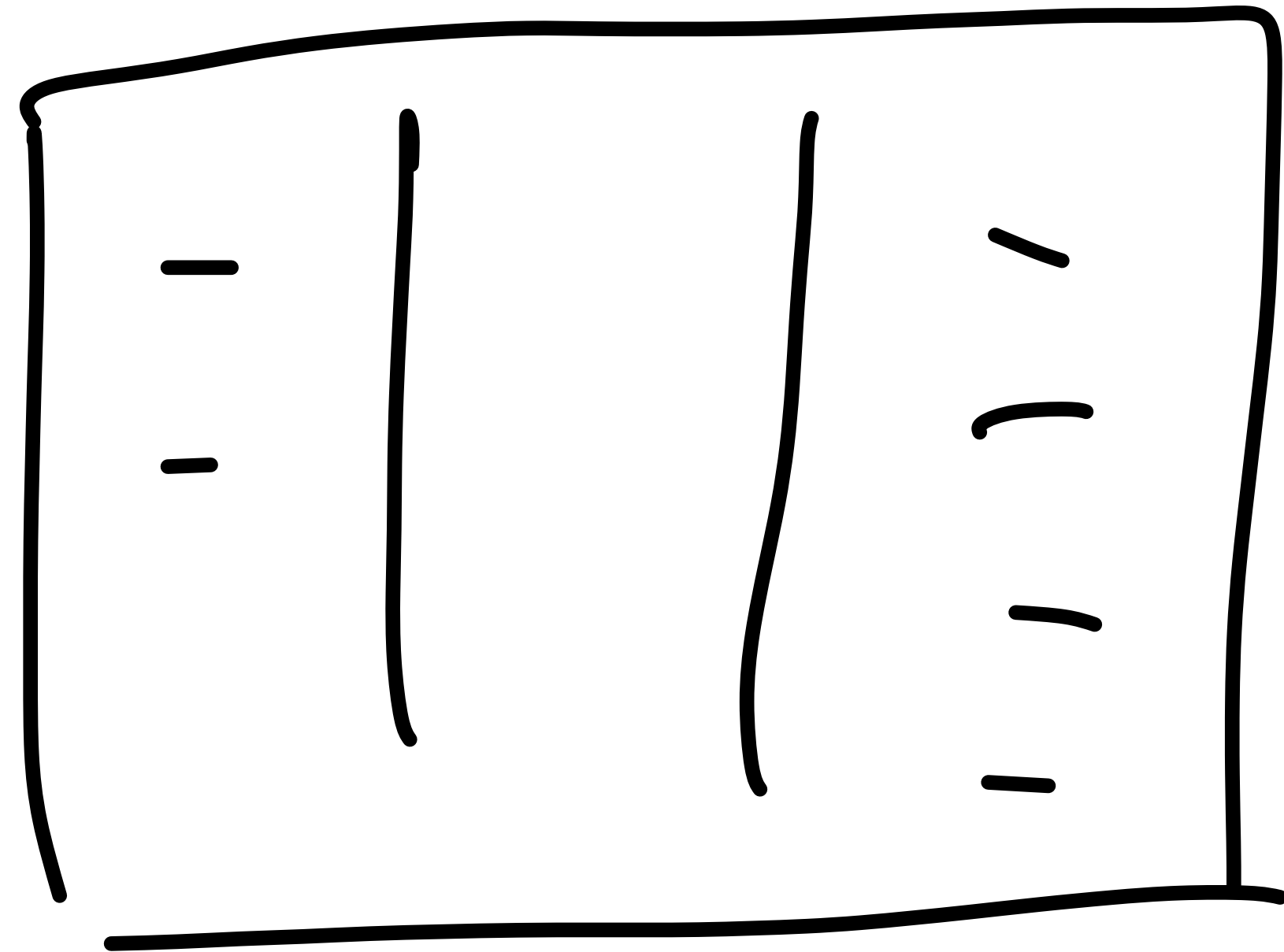


Product

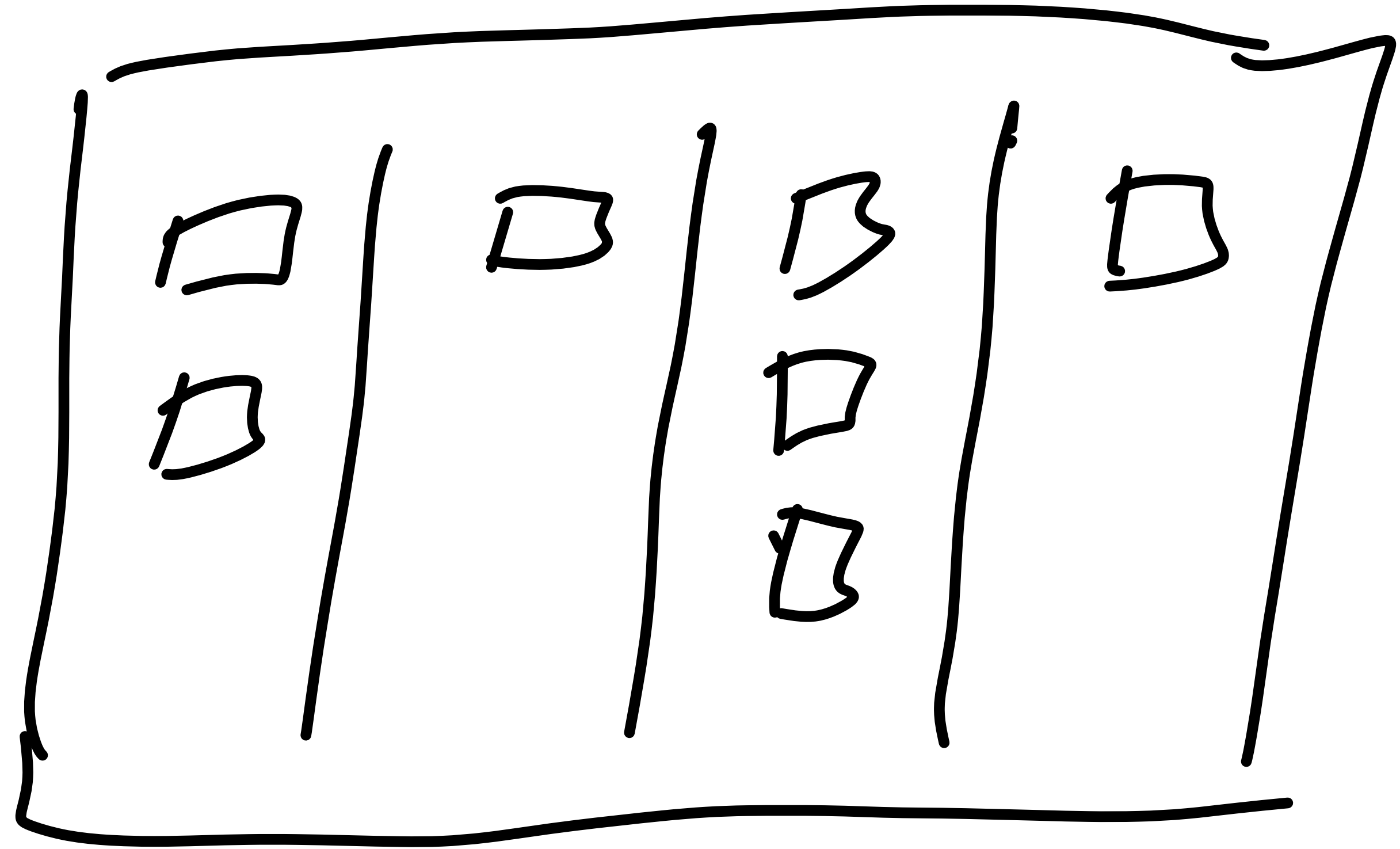


Dev

No Backlog



Product



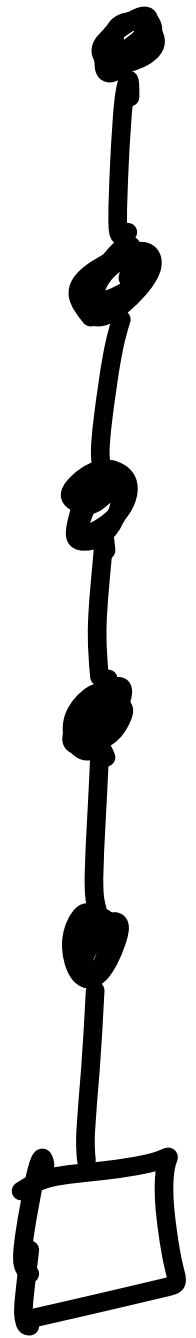
Dev

x 10

Test in Production

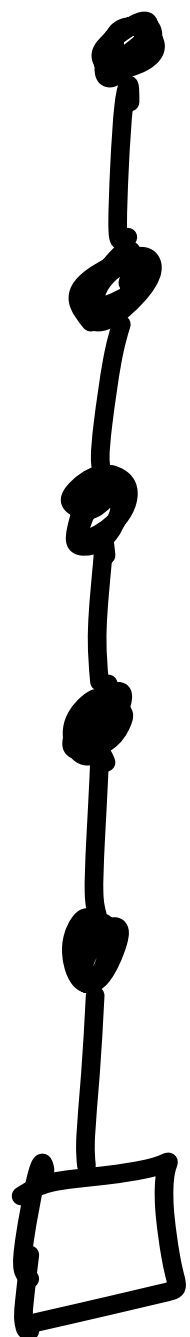
Test in Production

Test Case

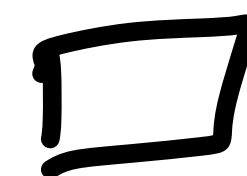


Test in Production

Test Case

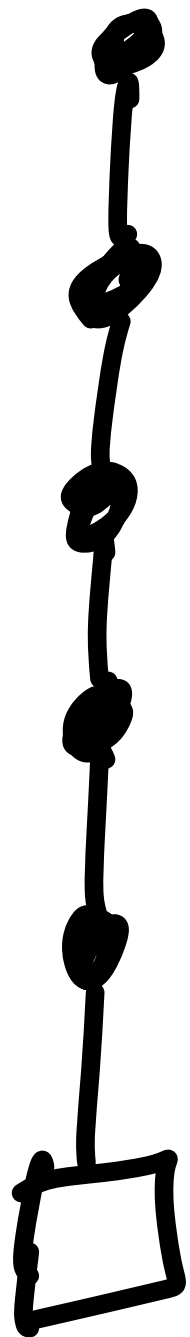


Exploratory Testing

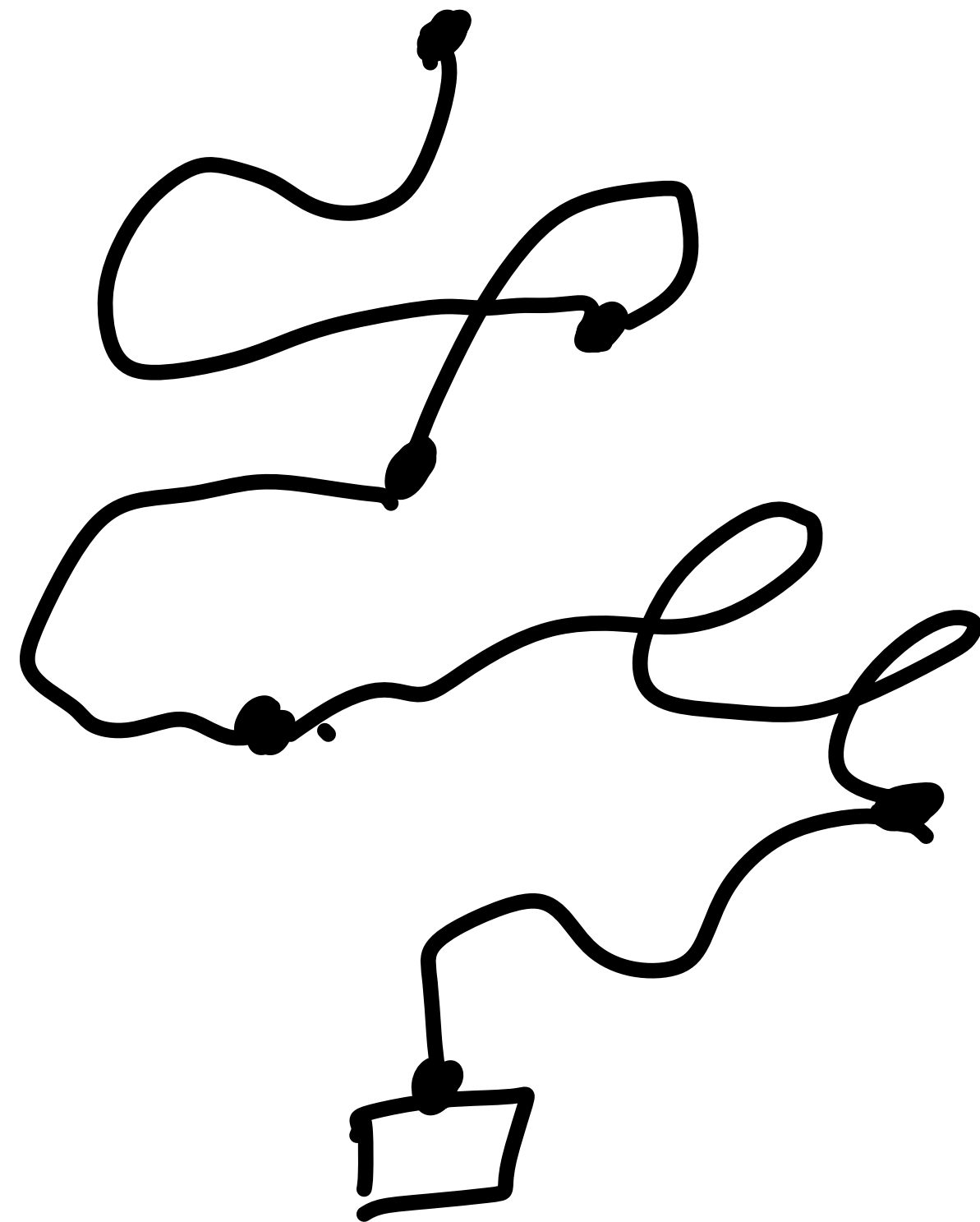


Test in Production

Test Case

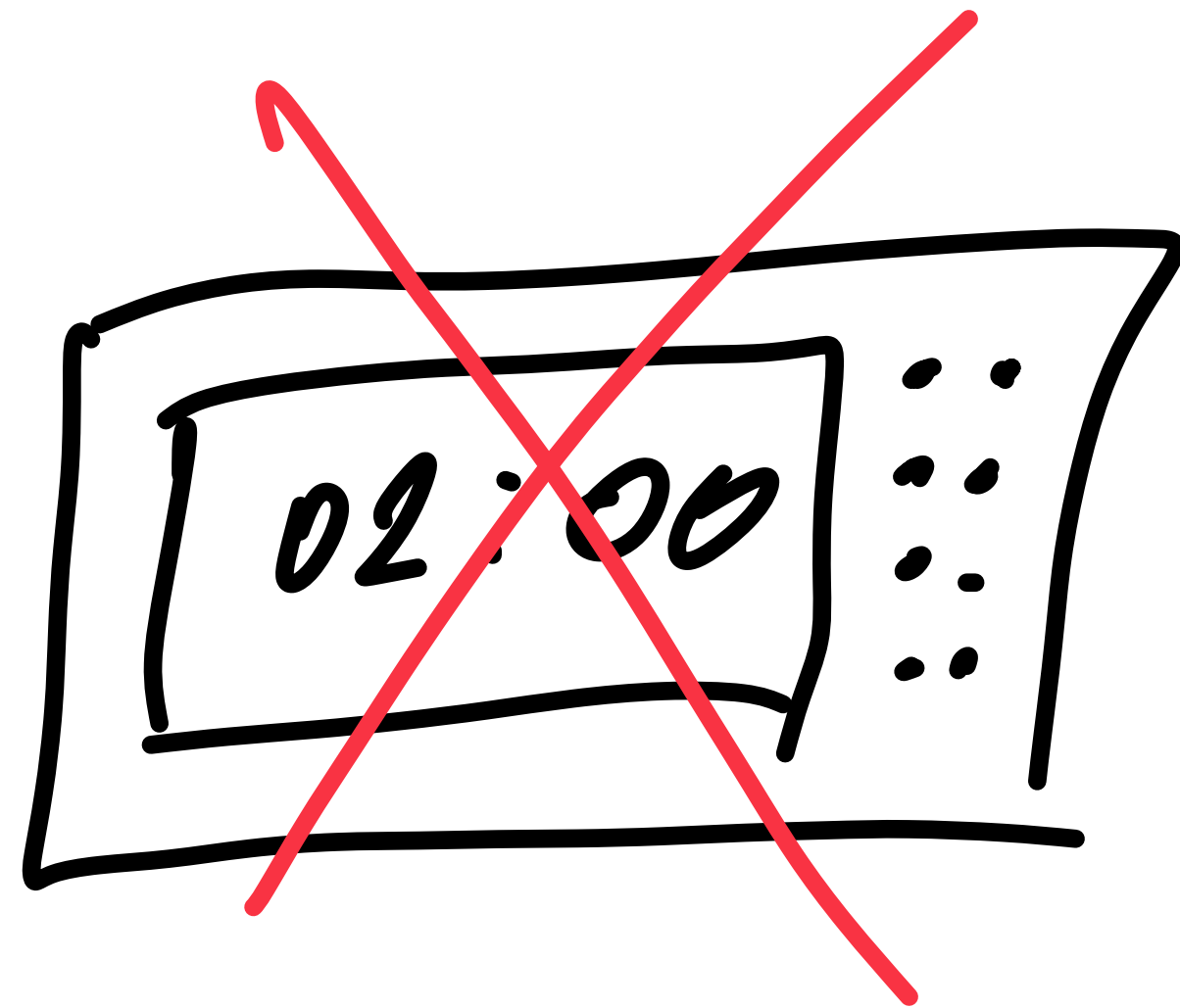
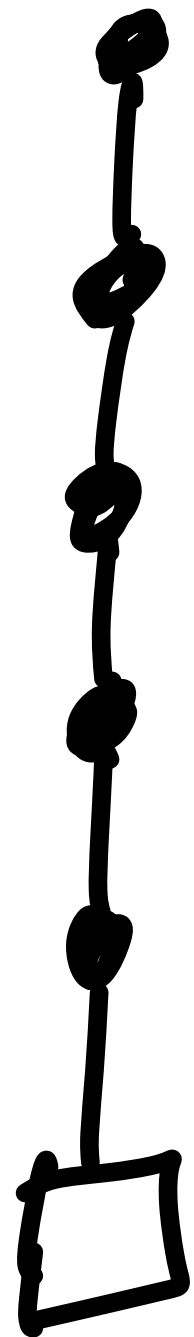


Exploratory Testing

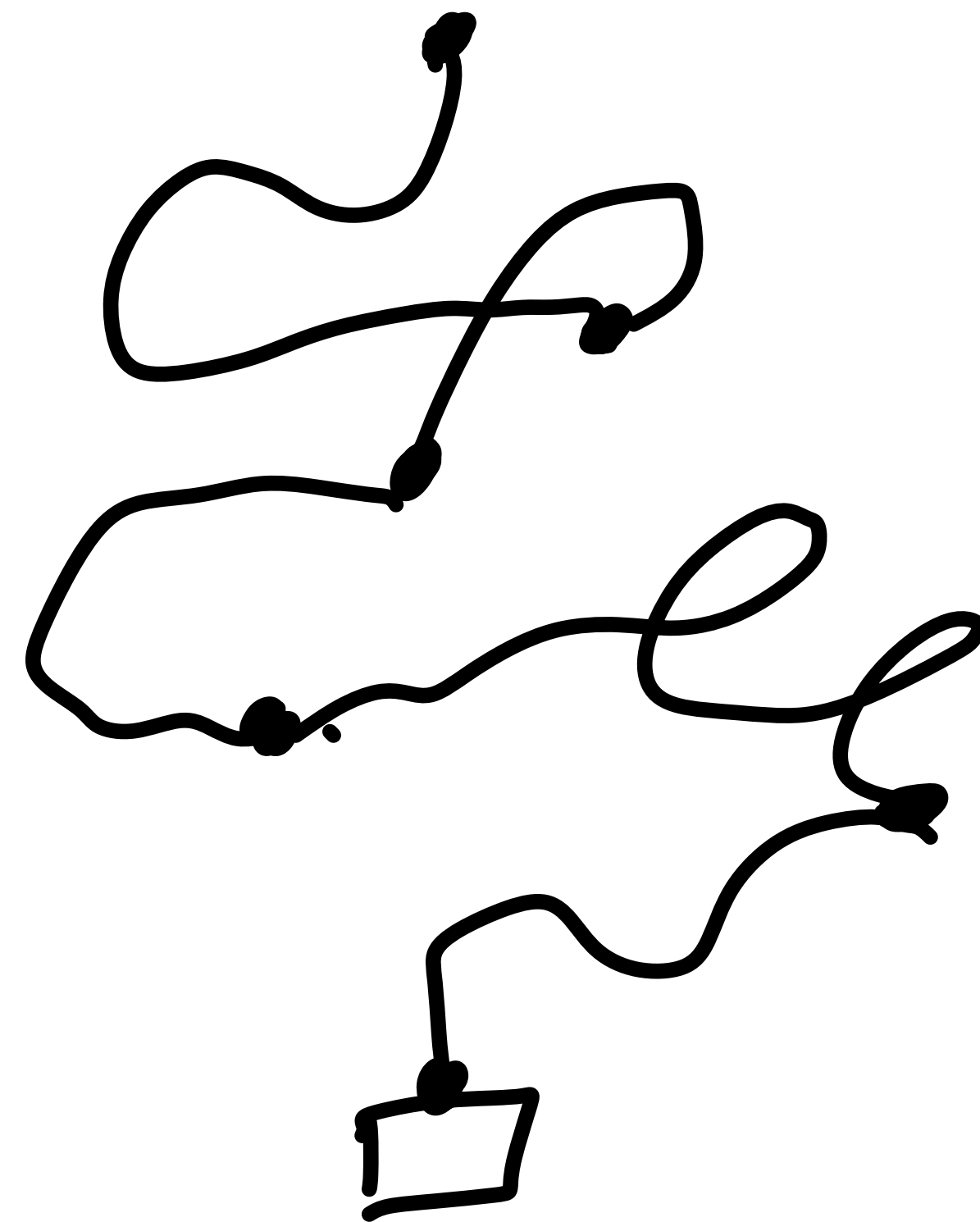


Test in Production

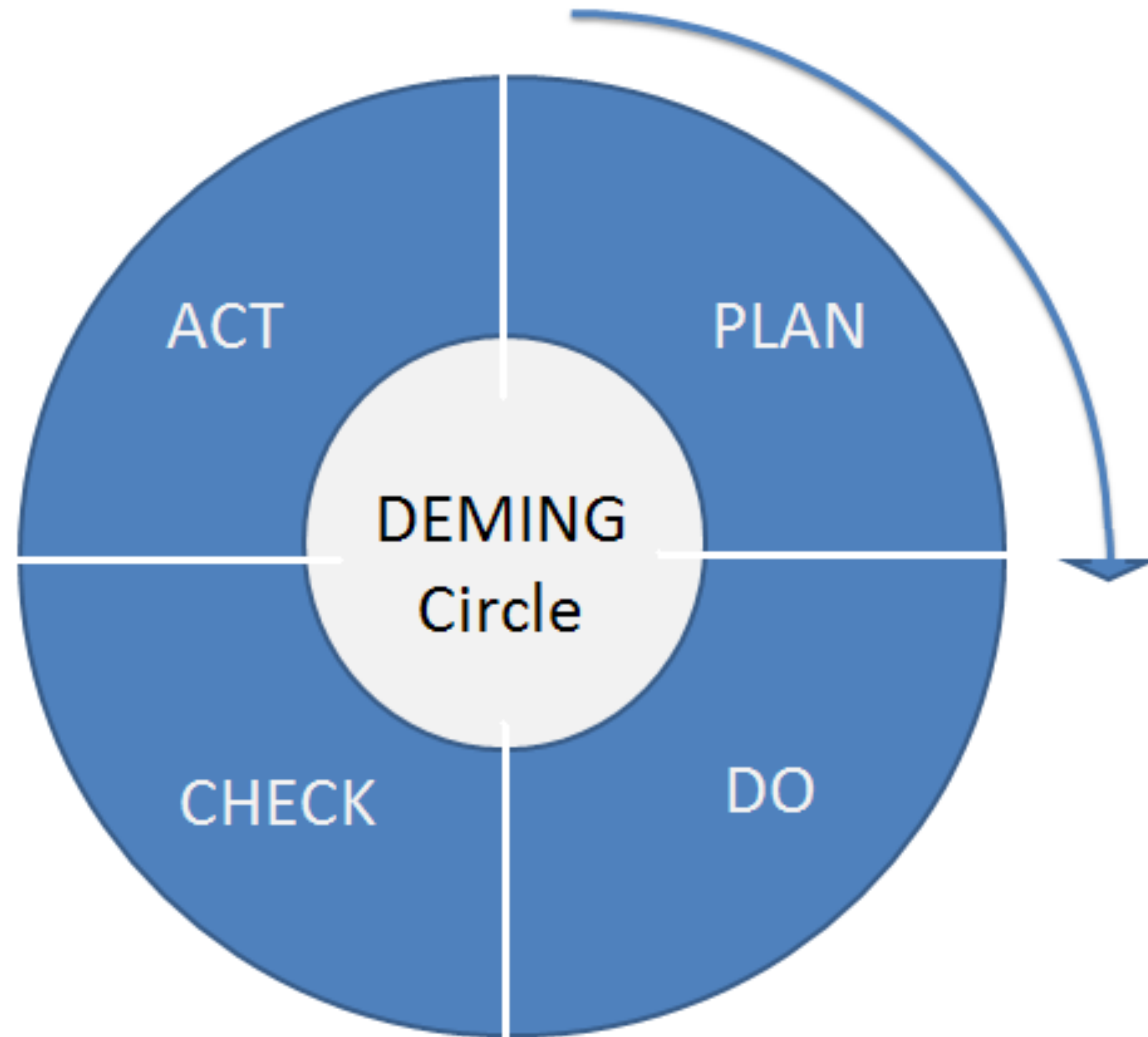
Test Case

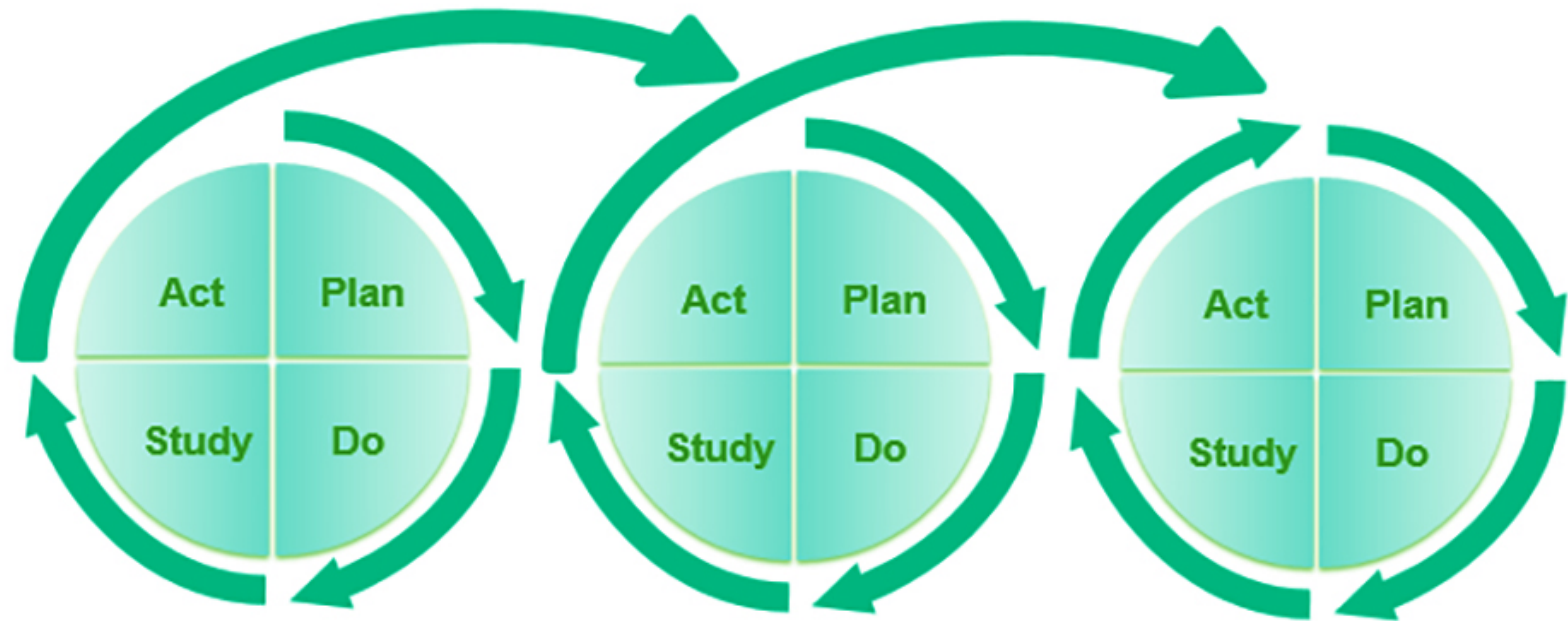


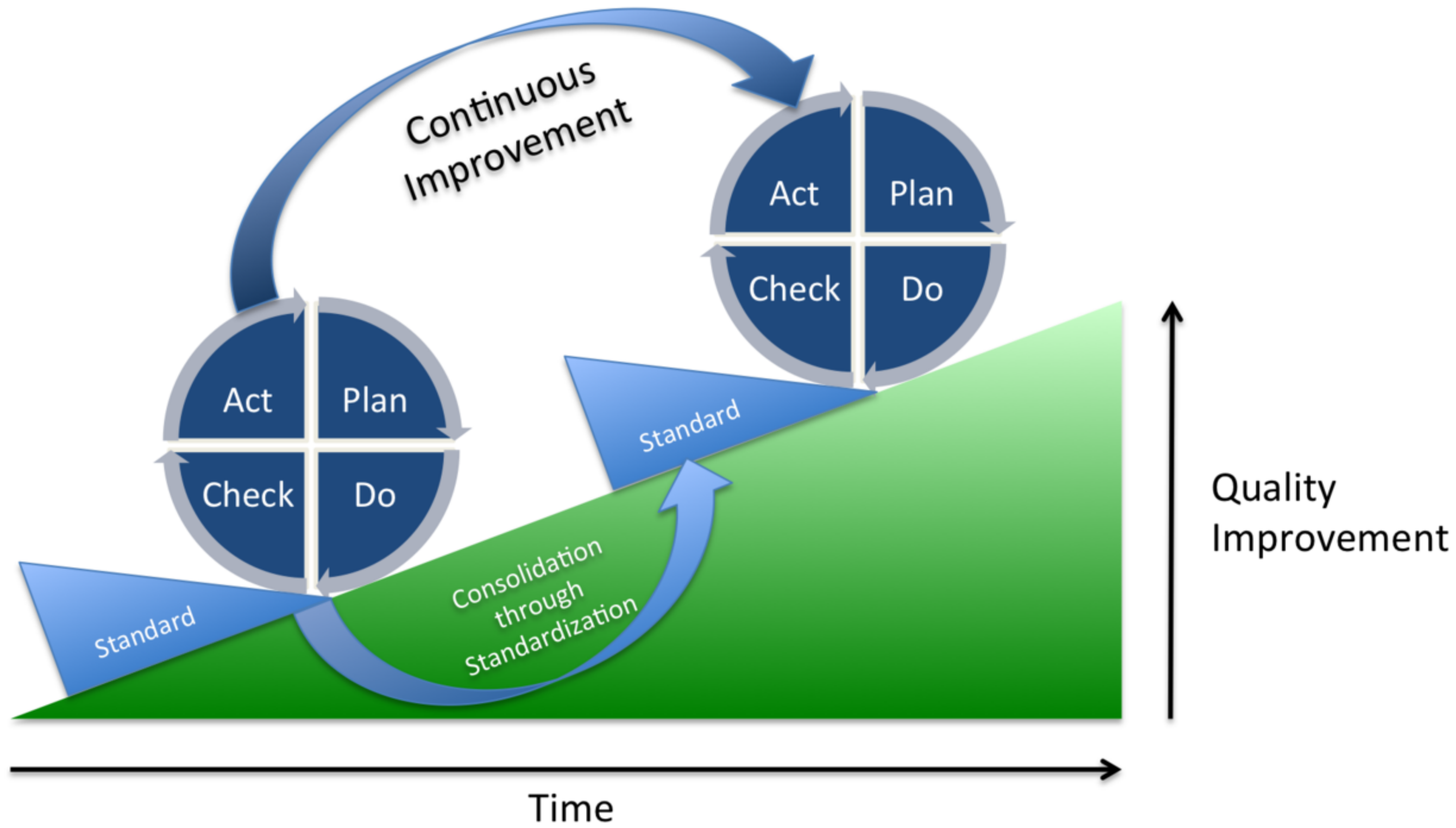
Exploratory Testing



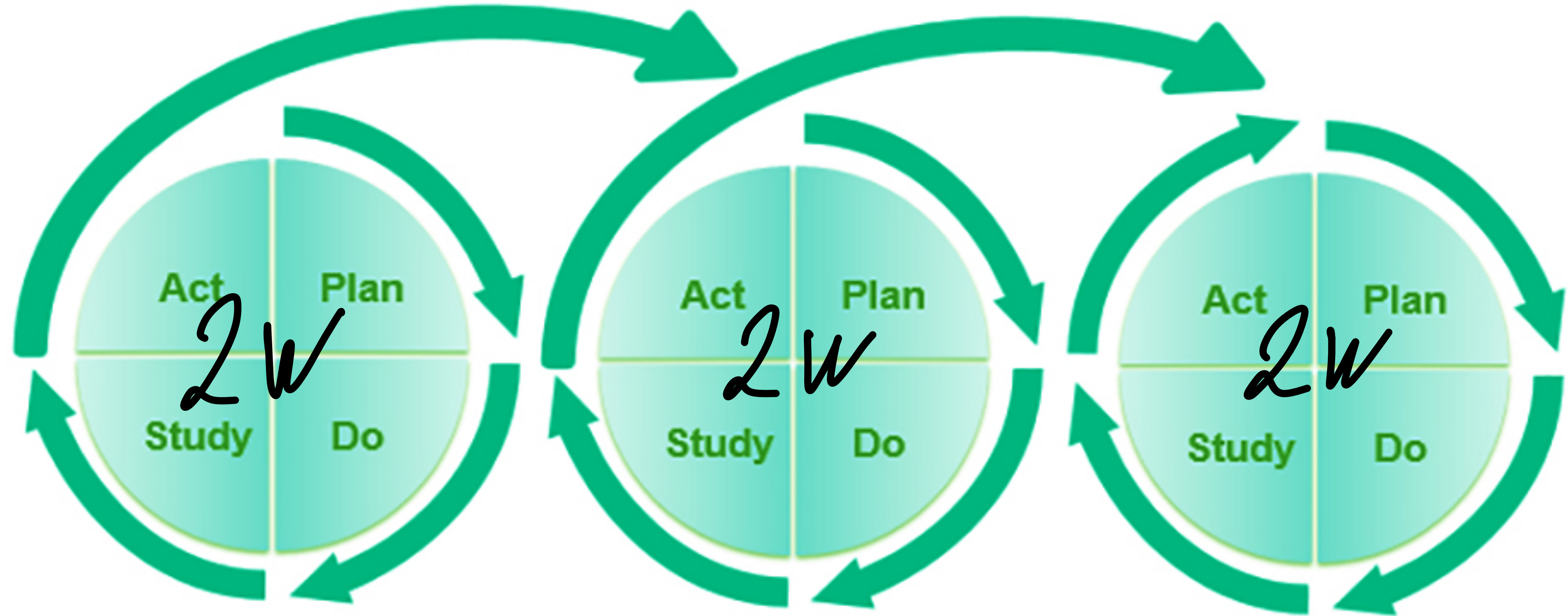
Почему эксперименты?



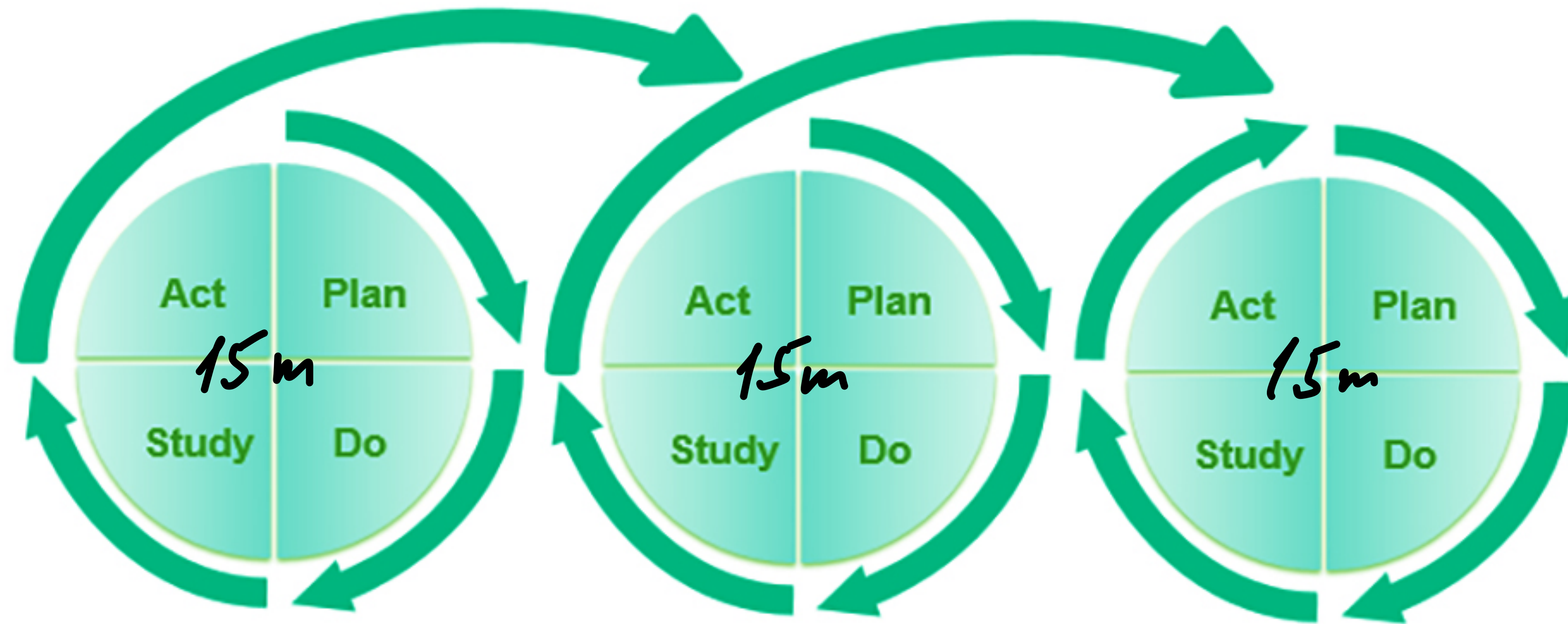


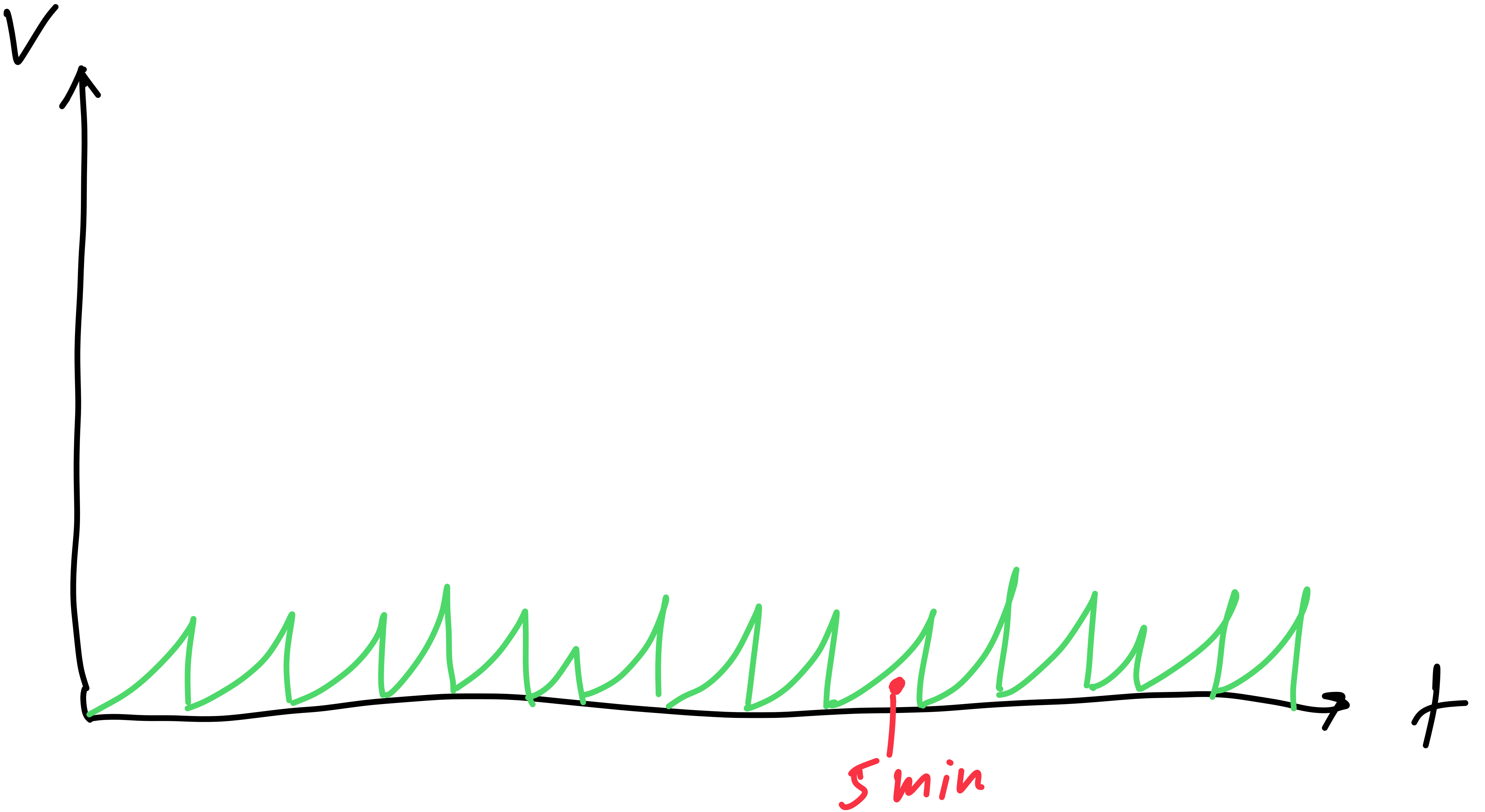


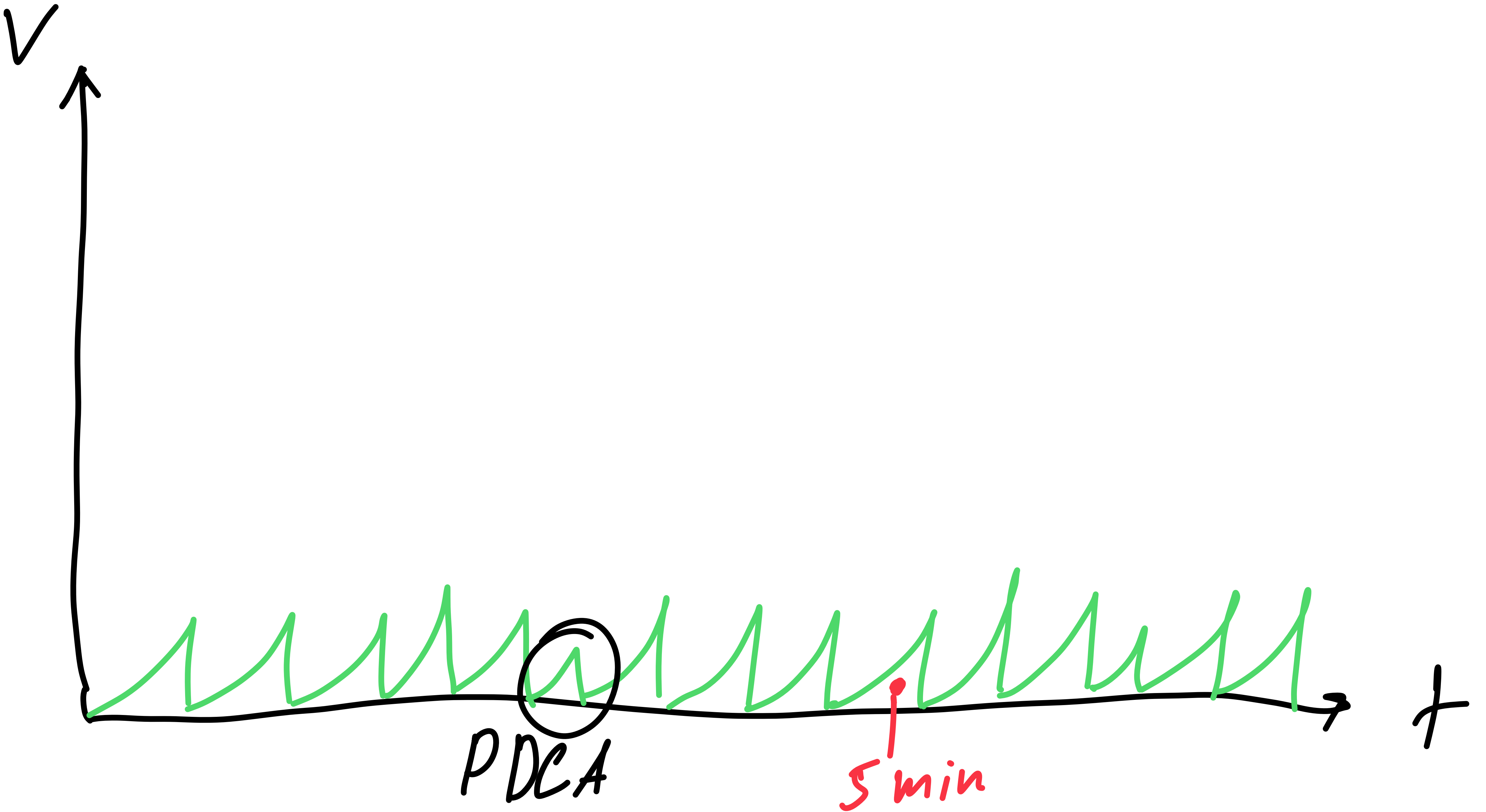
Sprint



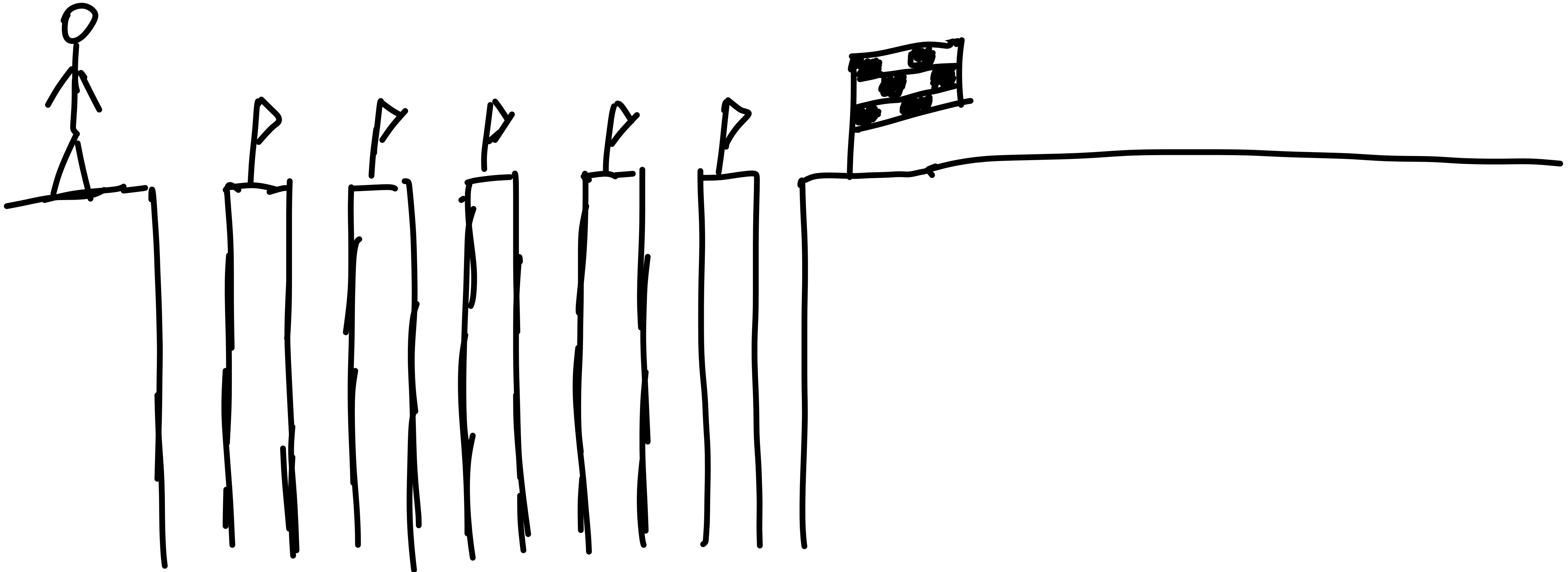
TDD



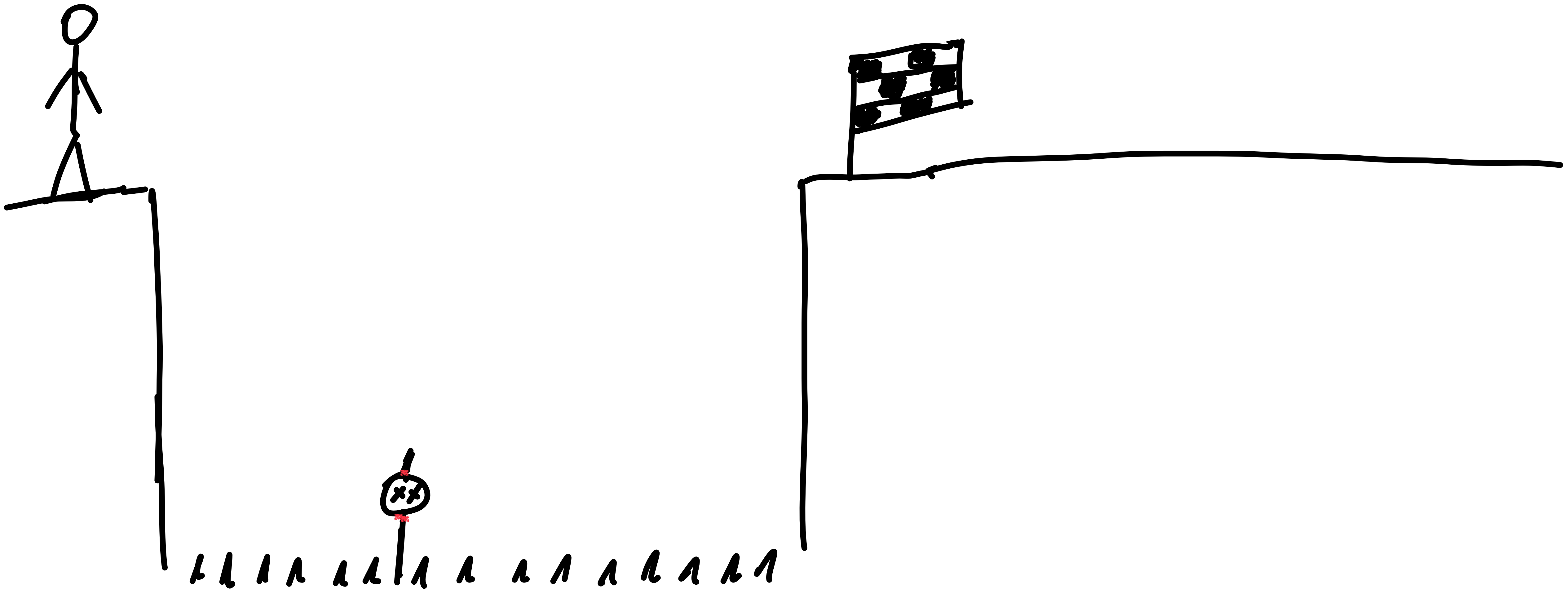




TDD

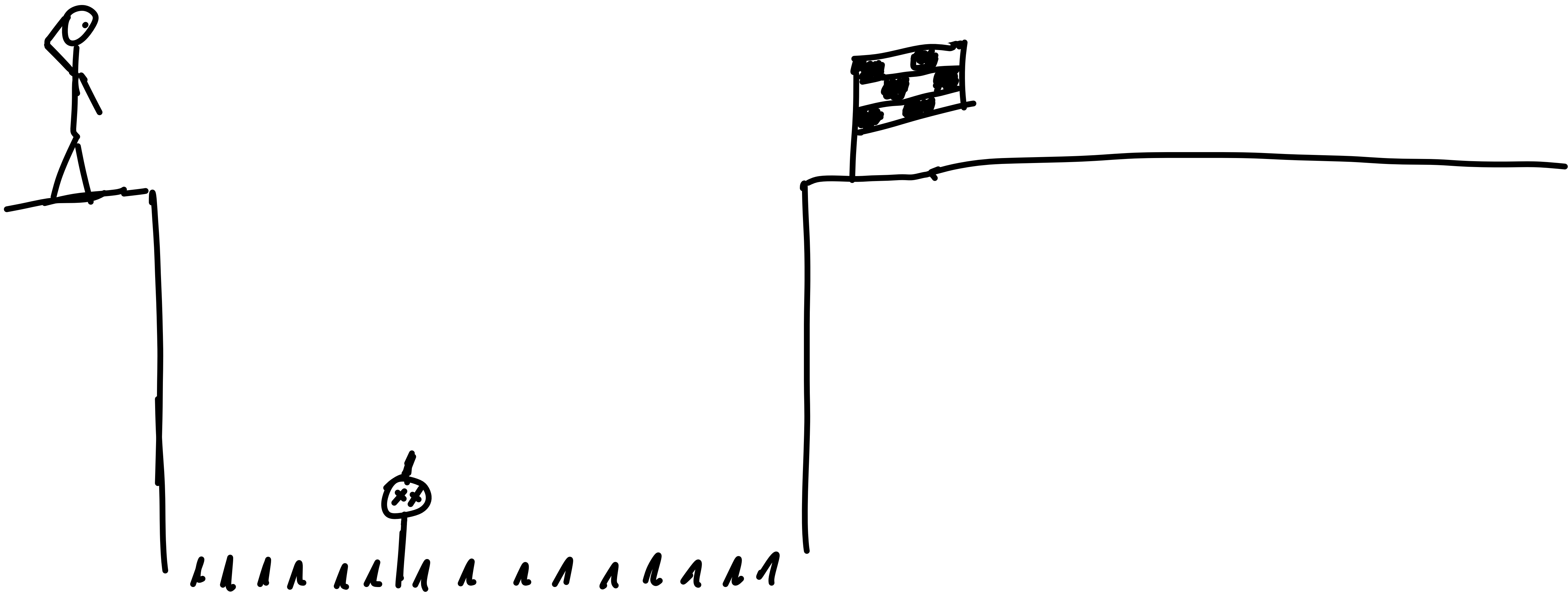


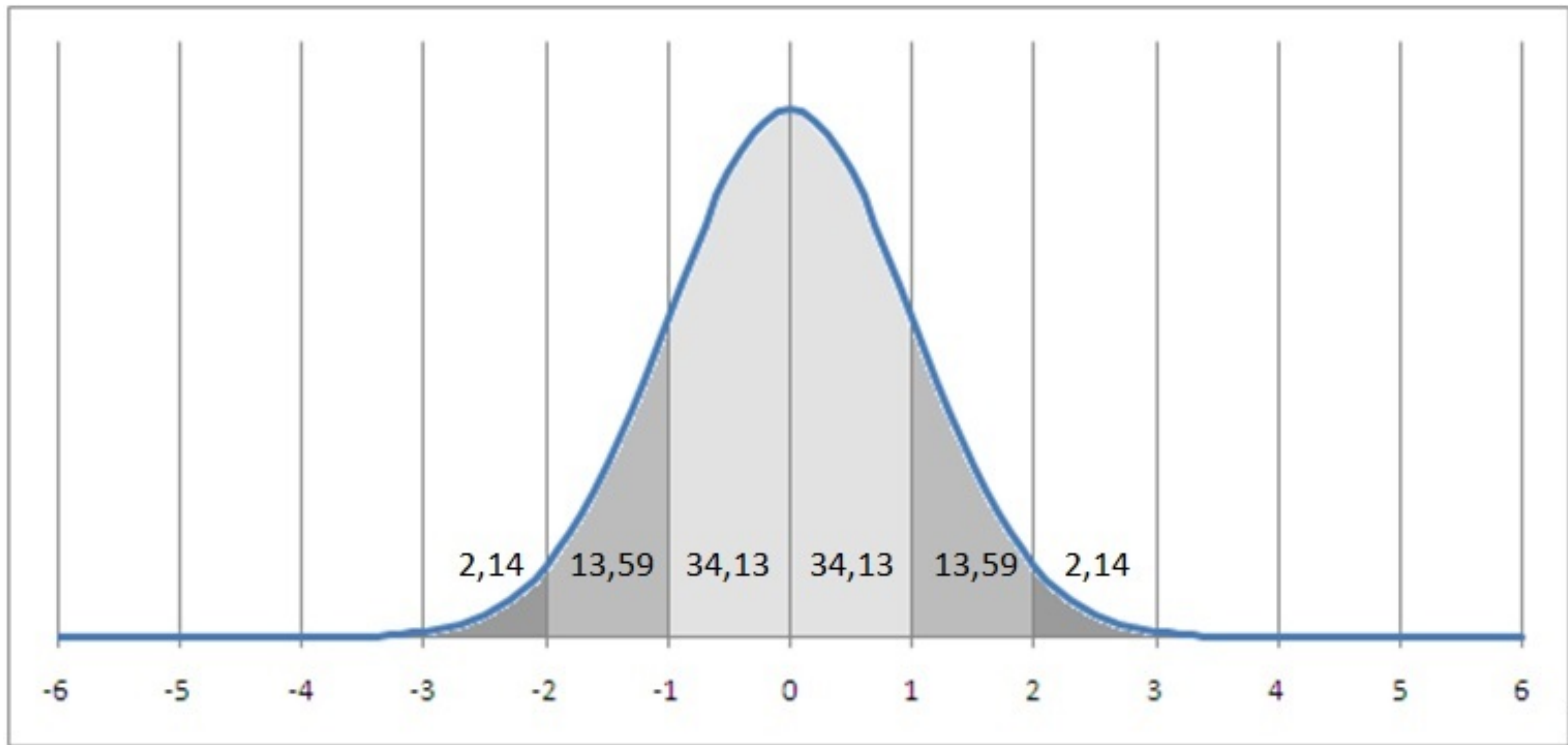
Sprint



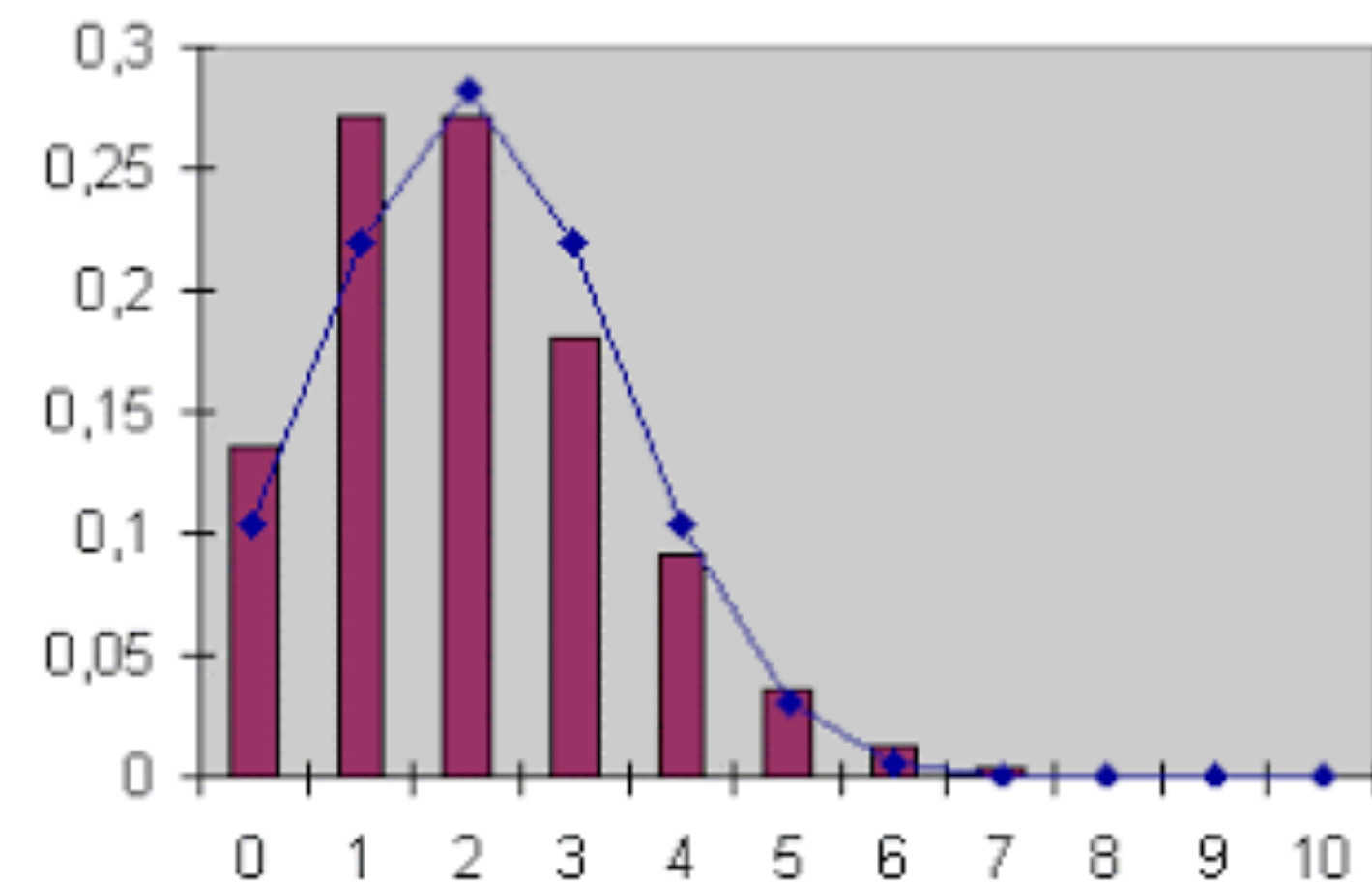
Sprint

???

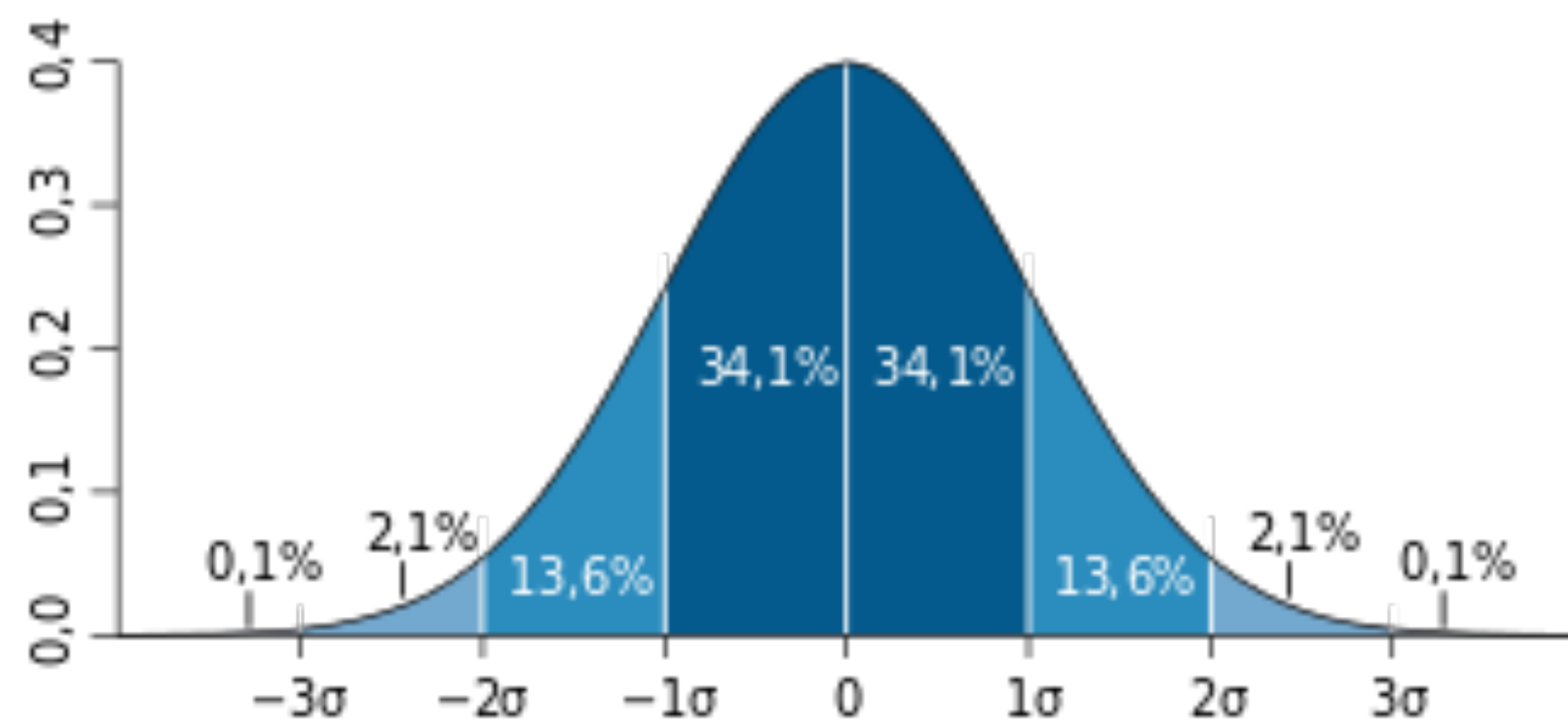
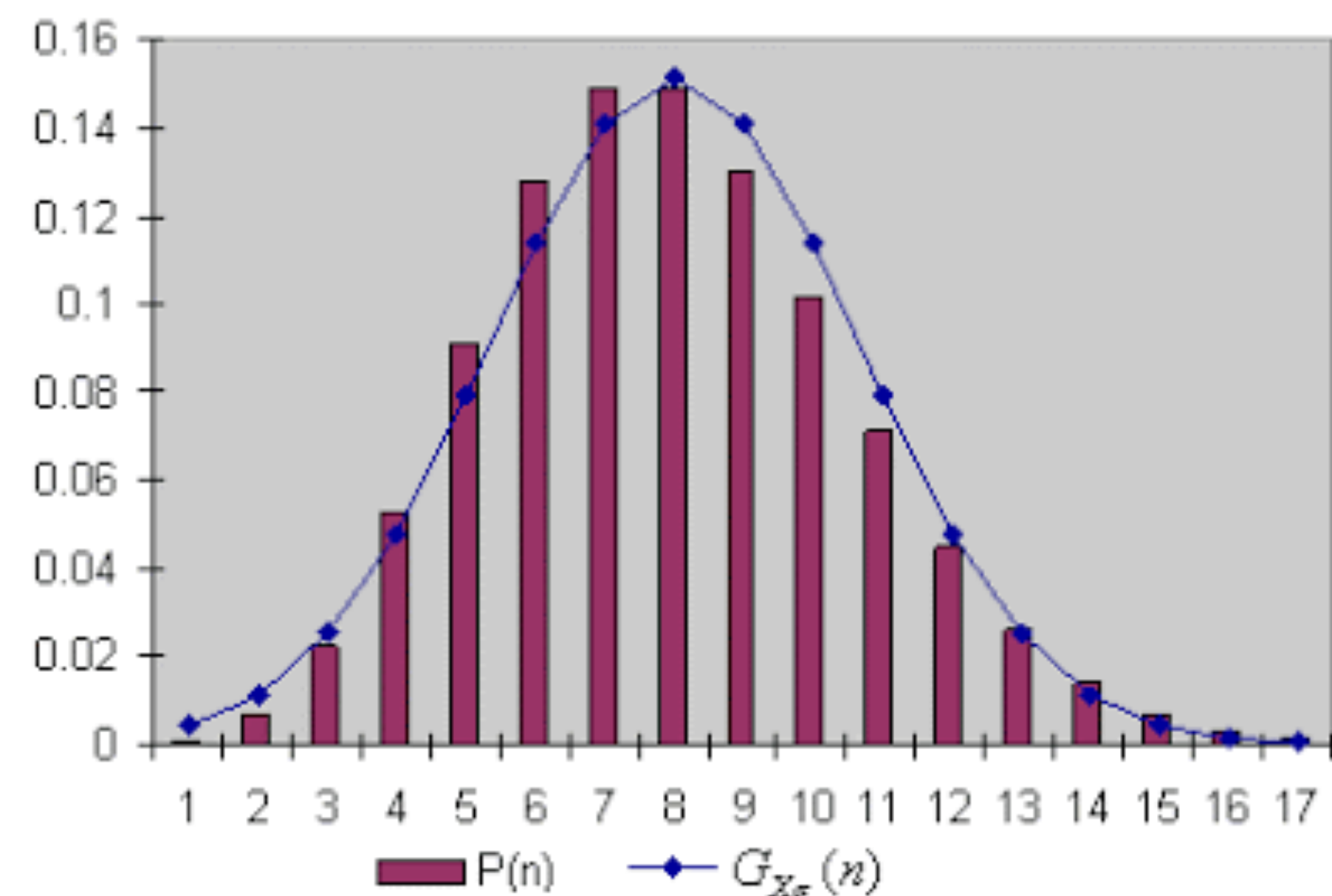




$\bar{n} = 2$

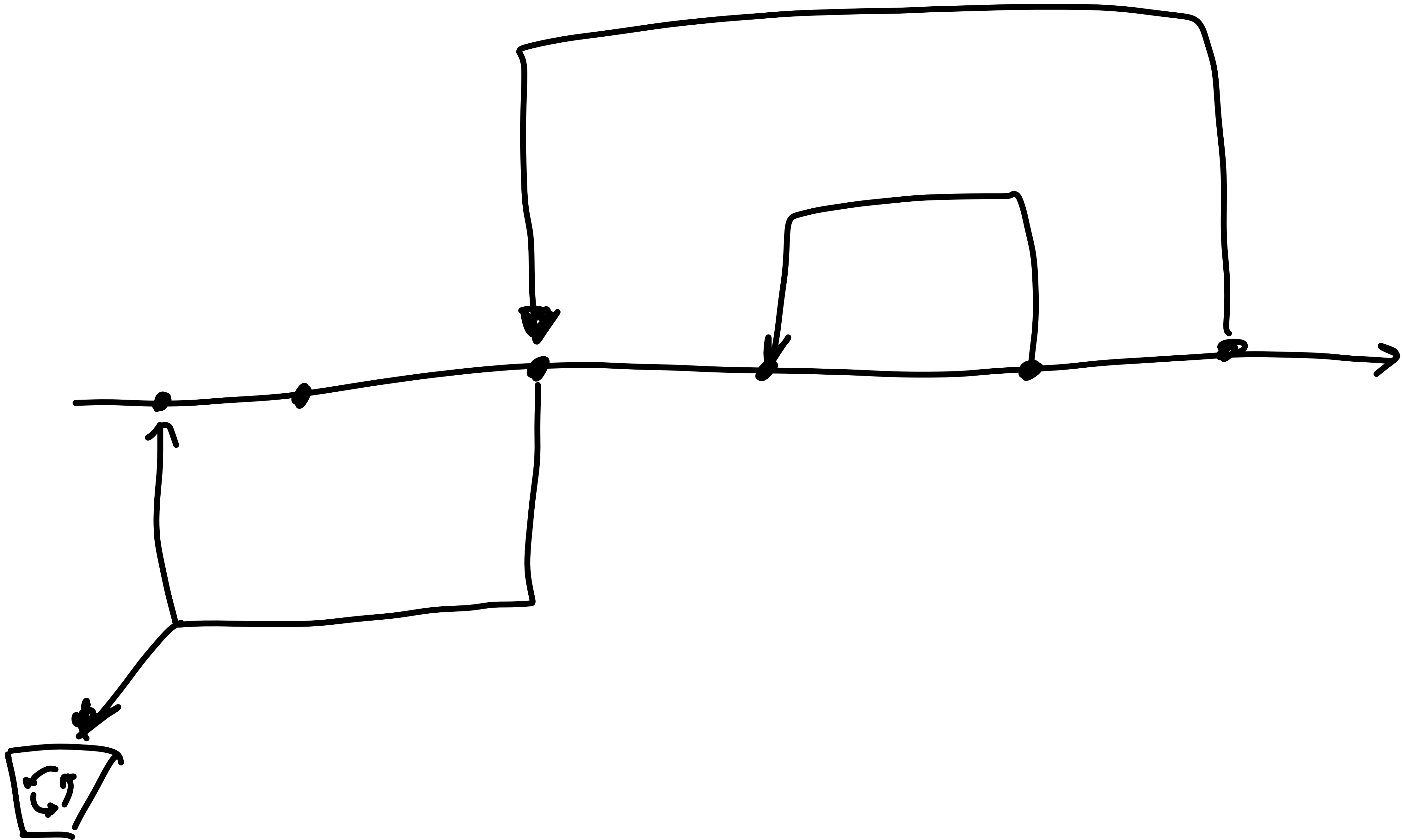


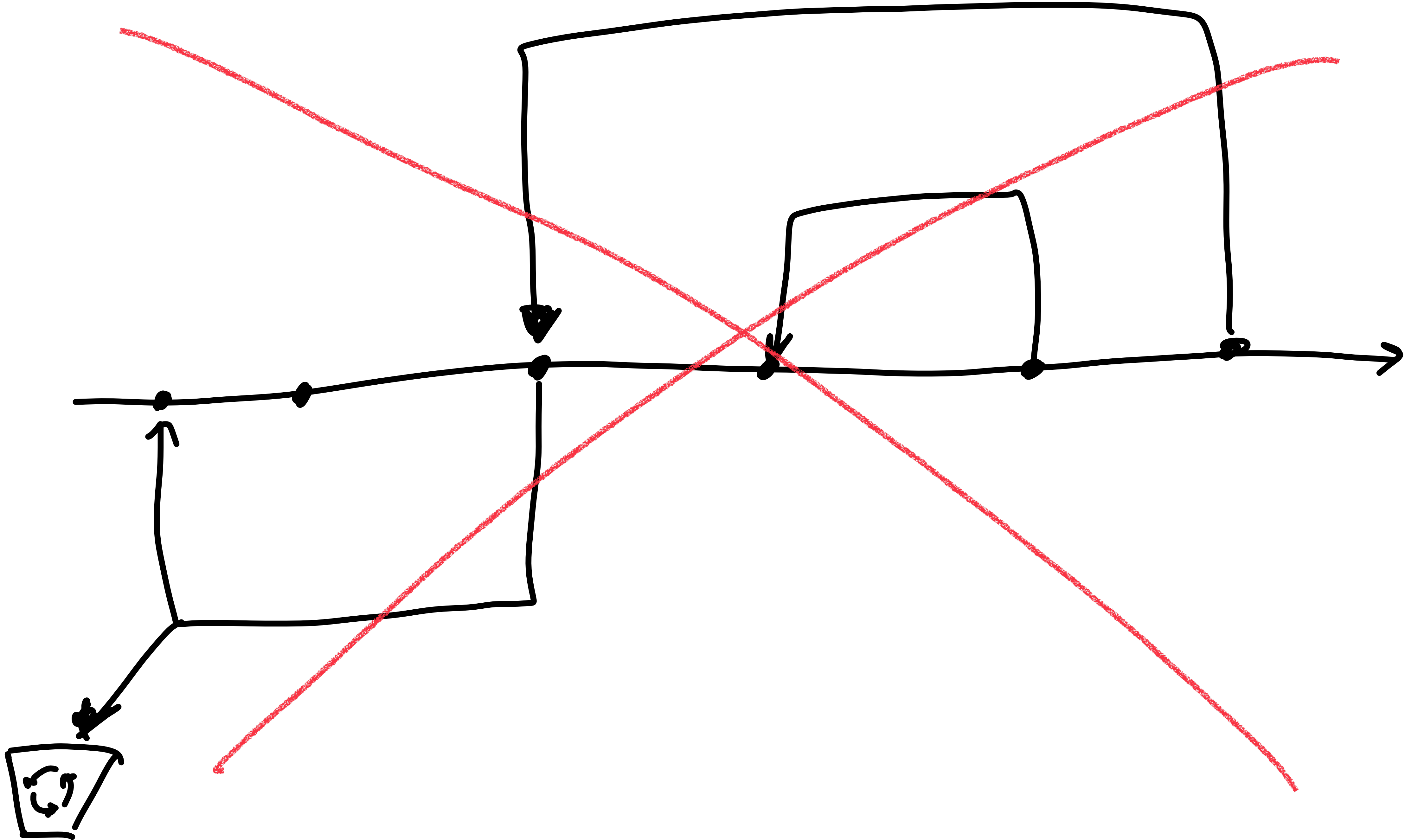
$\bar{n} = 7$

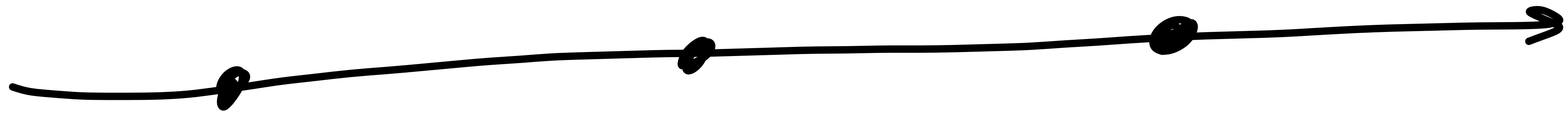


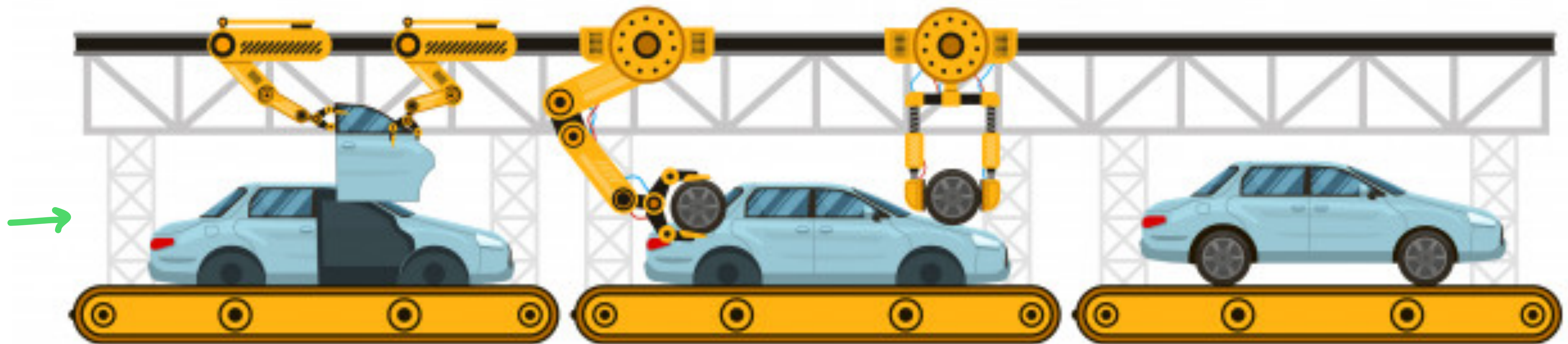
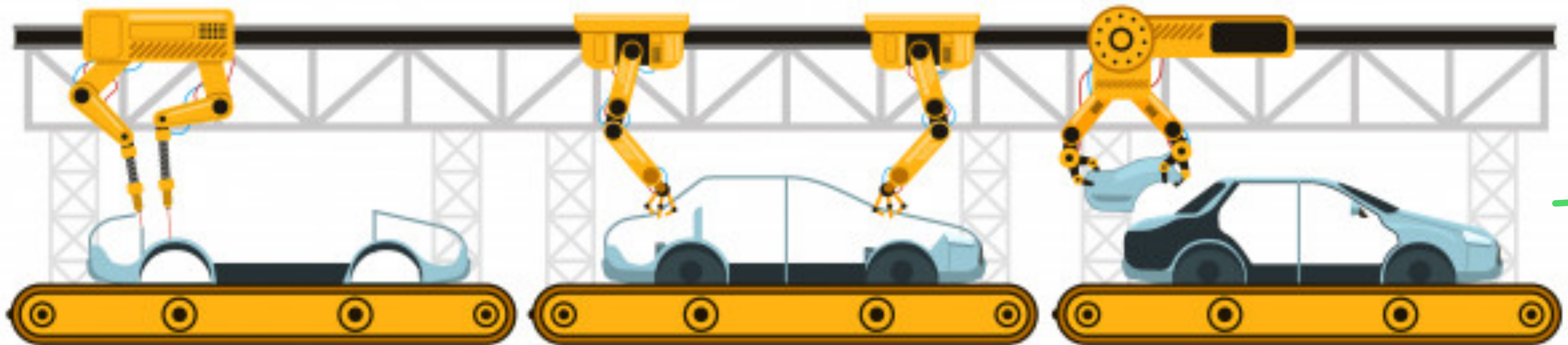
Важно

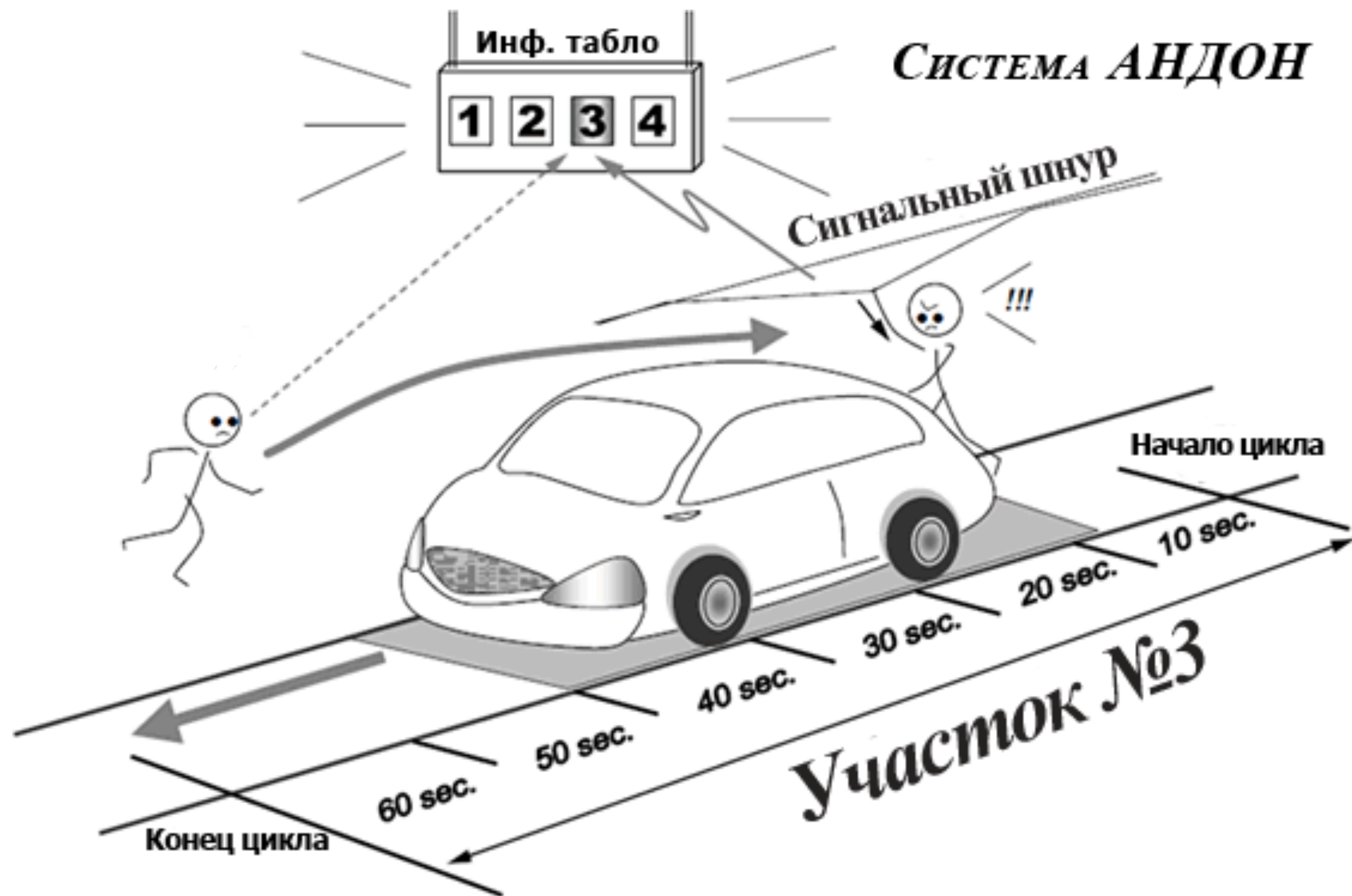
Качество достигается улучшением стабильного процесса











Обучение через инциденты

***Эволюция
руководствовалась одним
инструментом — ошибкой***



Первый и второй путь

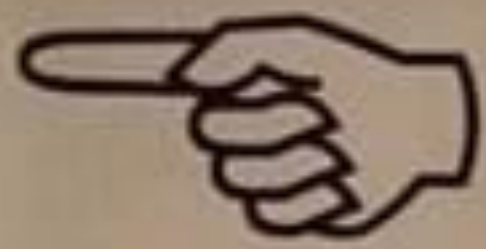
- Ускоряем и стабилизируем процесс разработки
- Боремся с вариацией
- Боремся с трением
- Экспериментируем
- Учимся на ошибках

Третий путь

- Развитие первых двух
- Борьба с энтропией
- Культура
- Обучение новому

Пора уничтожать отделы





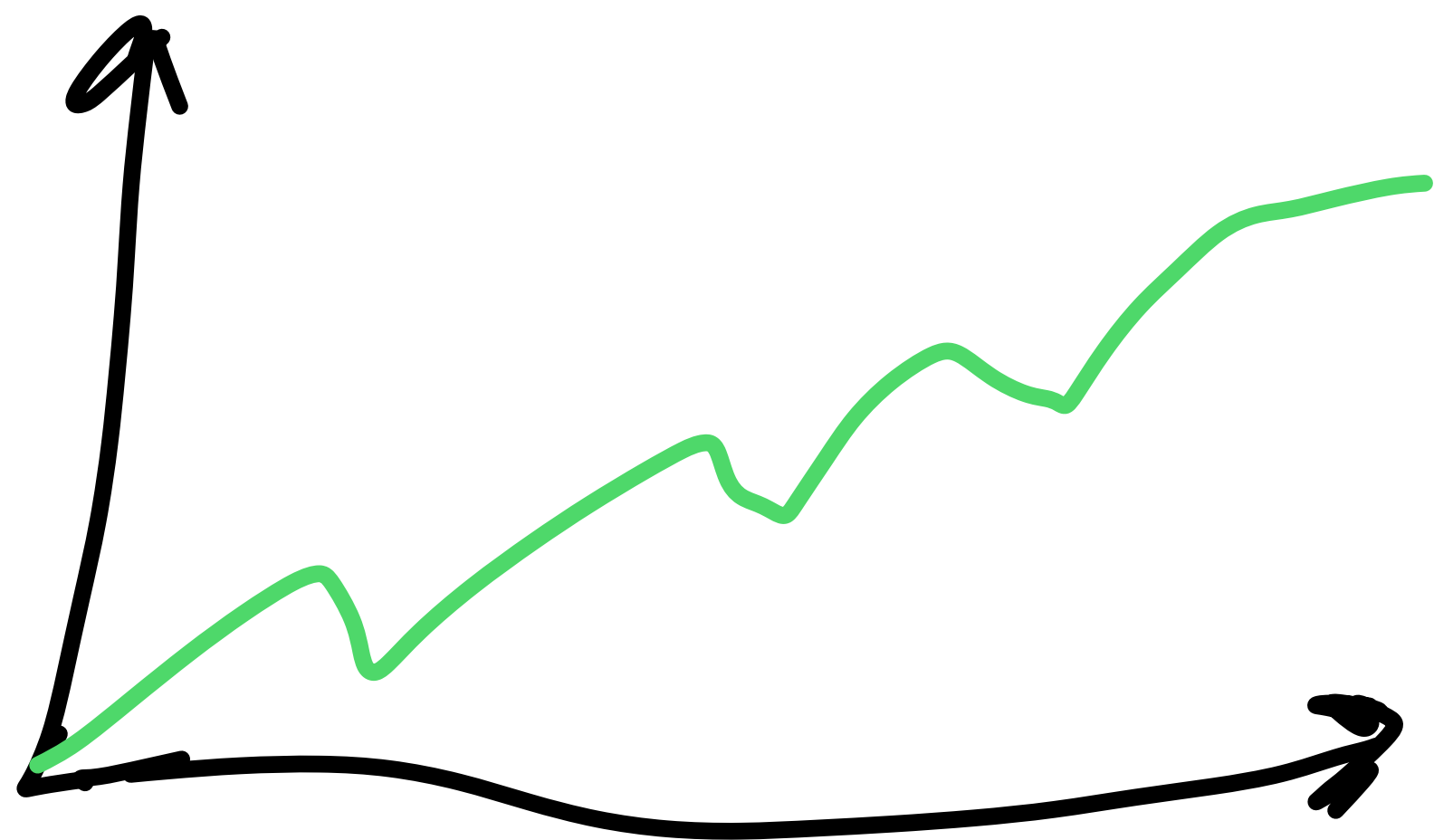
ТЕХОТДЕЛ

ЭТИХОТДЕЛ



$f(x)$

$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$

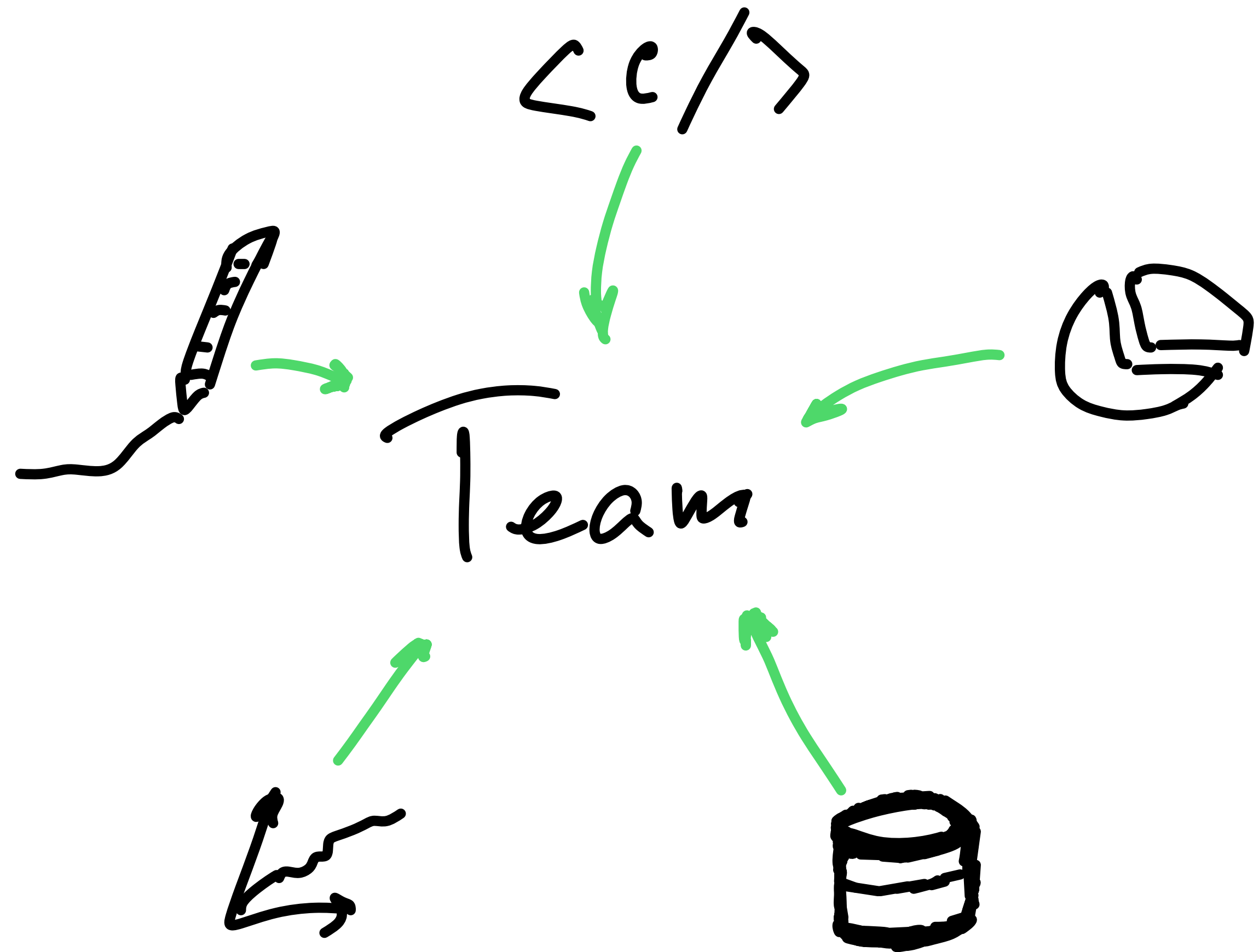


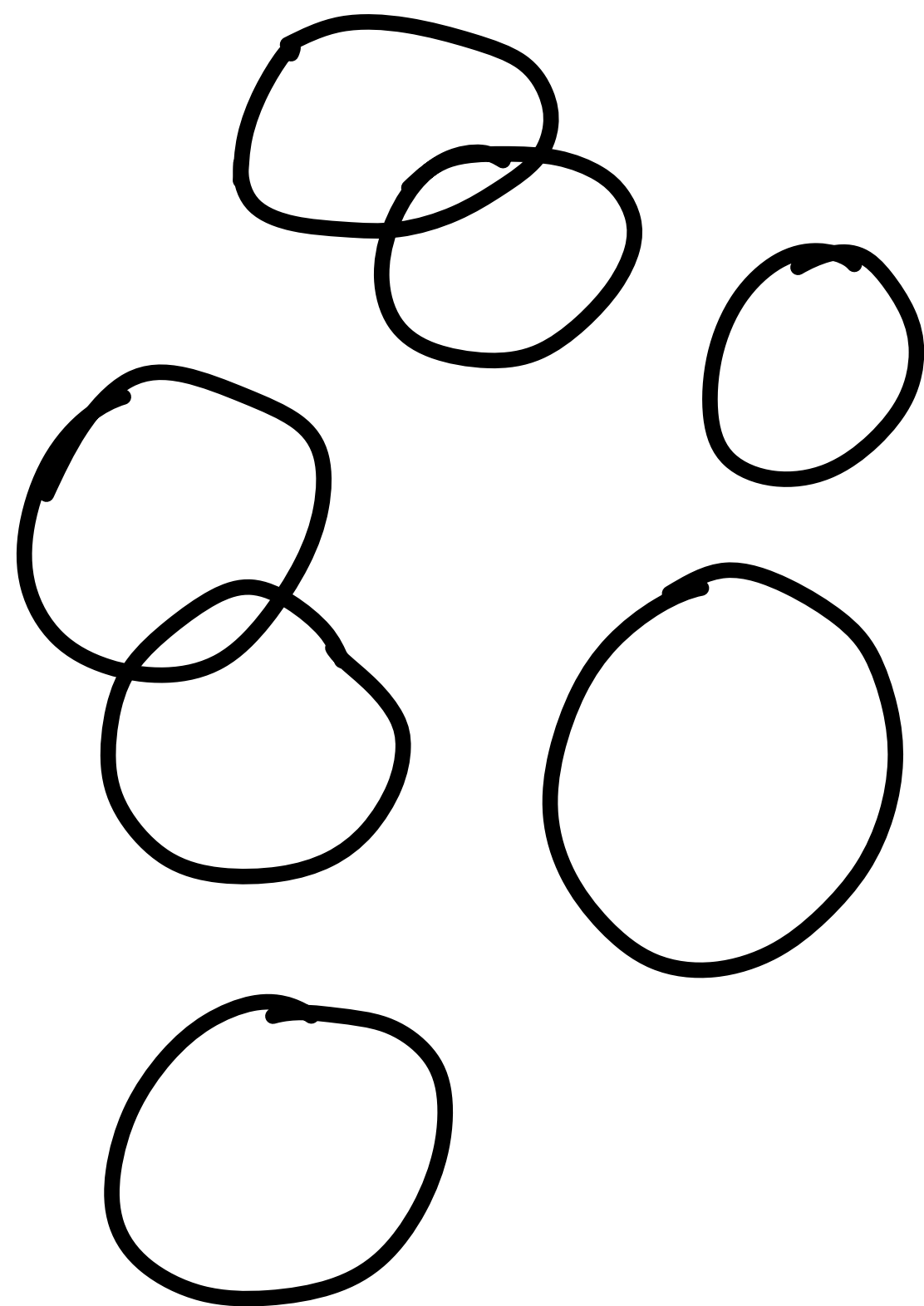


Vitaly D
@vdvision

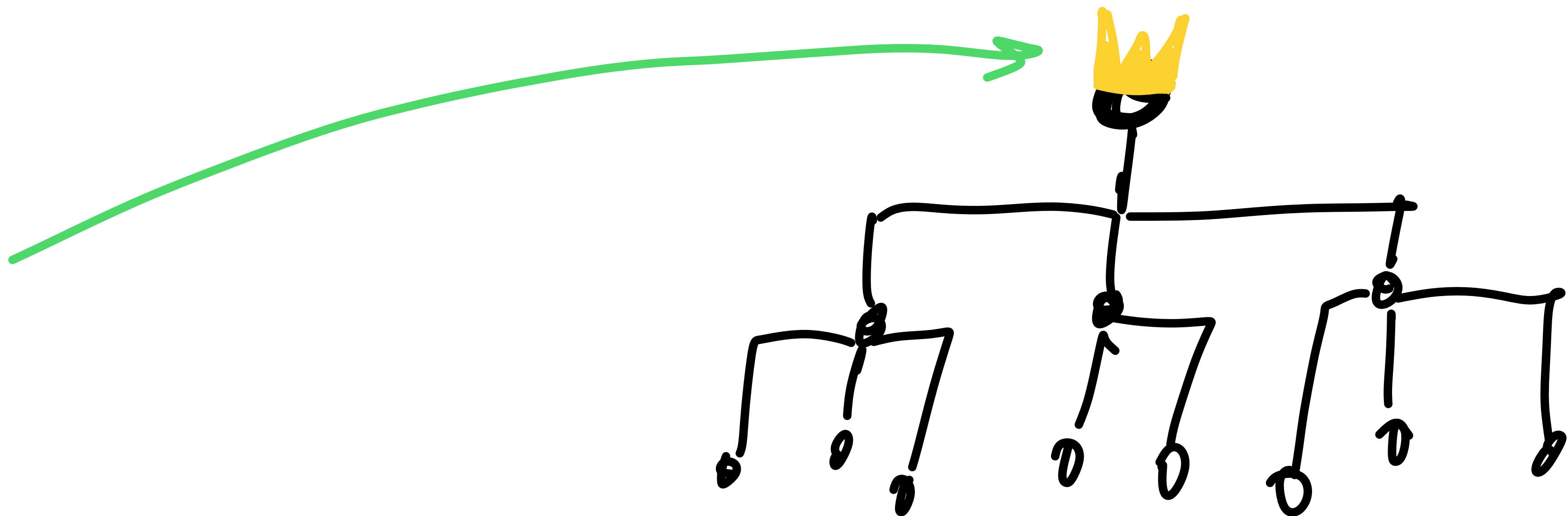
...

Nowadays there is no team, but a bunch of people called product team for some reason. Make no mistake: only people working at the same time, in the same space, on the same thing can produce value efficiently. That's what team really is.





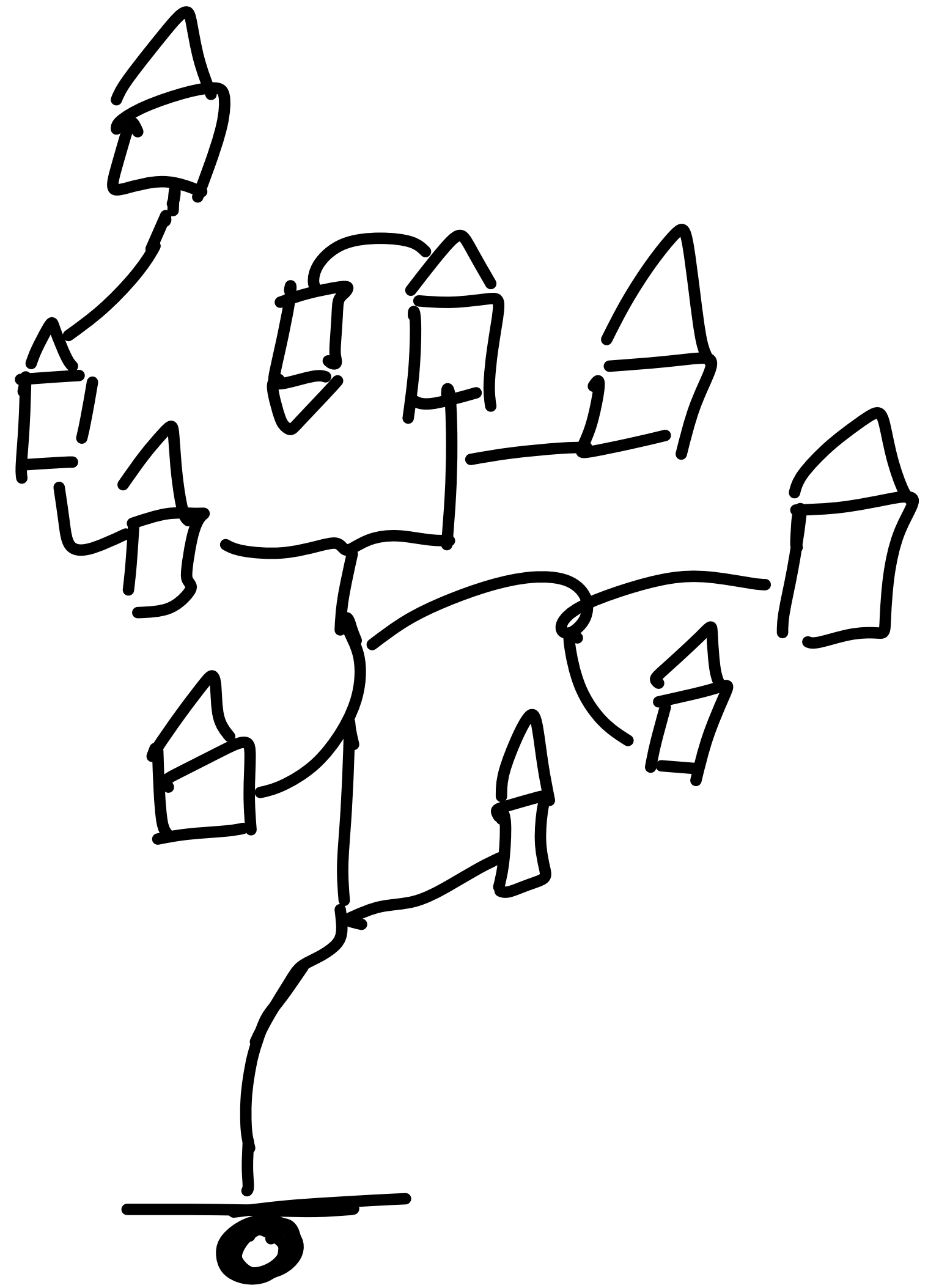
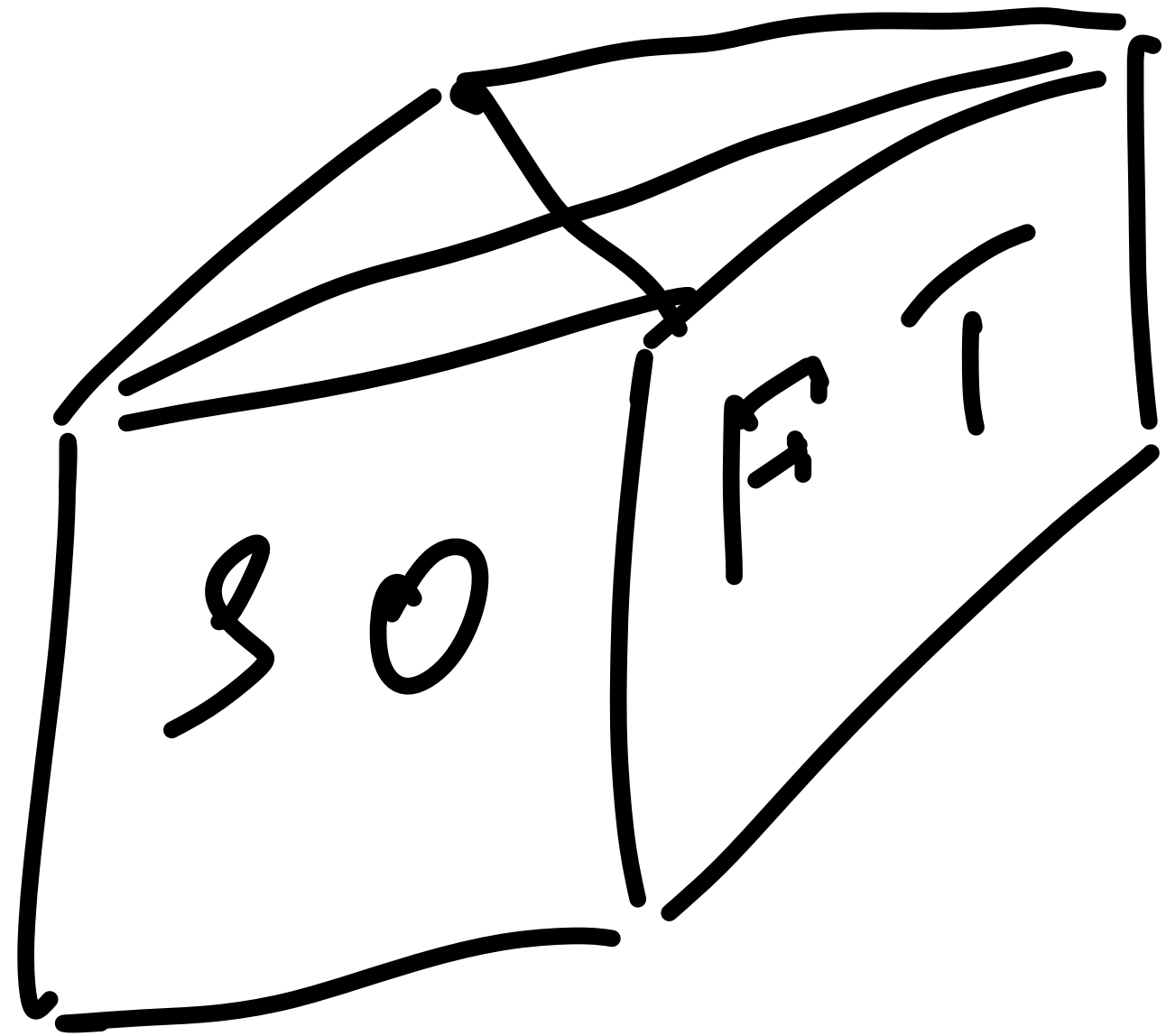
Teams



Deps.

А ПОТОМ...





Коробка

- Поставляется в банк
- Приемочное тестирование на стороне заказчика
- Эксплуатация на стороне заказчика
- Разработка пушит в мастер, но хочет отказаться от этого
- Аудитория - сотрудники банка
- Скрам и планирование

Коробка

- Пушить в мастер — хорошо
- Переход на авто-тесты
- Внутренний тестовый стенд
- Перевести сотрудников к себе

SCRUM

SCRUM

- Daily
- Retro
- Sprints
- Sprint review
- Planning
- Backlog refinement

Kanban

- Daily в стиле kanban
- Retro -> service review
- ~~Sprints~~
- ~~Sprint review~~
- ~~Planning~~
- Backlog refinement по запросу (собрание по пополнению)

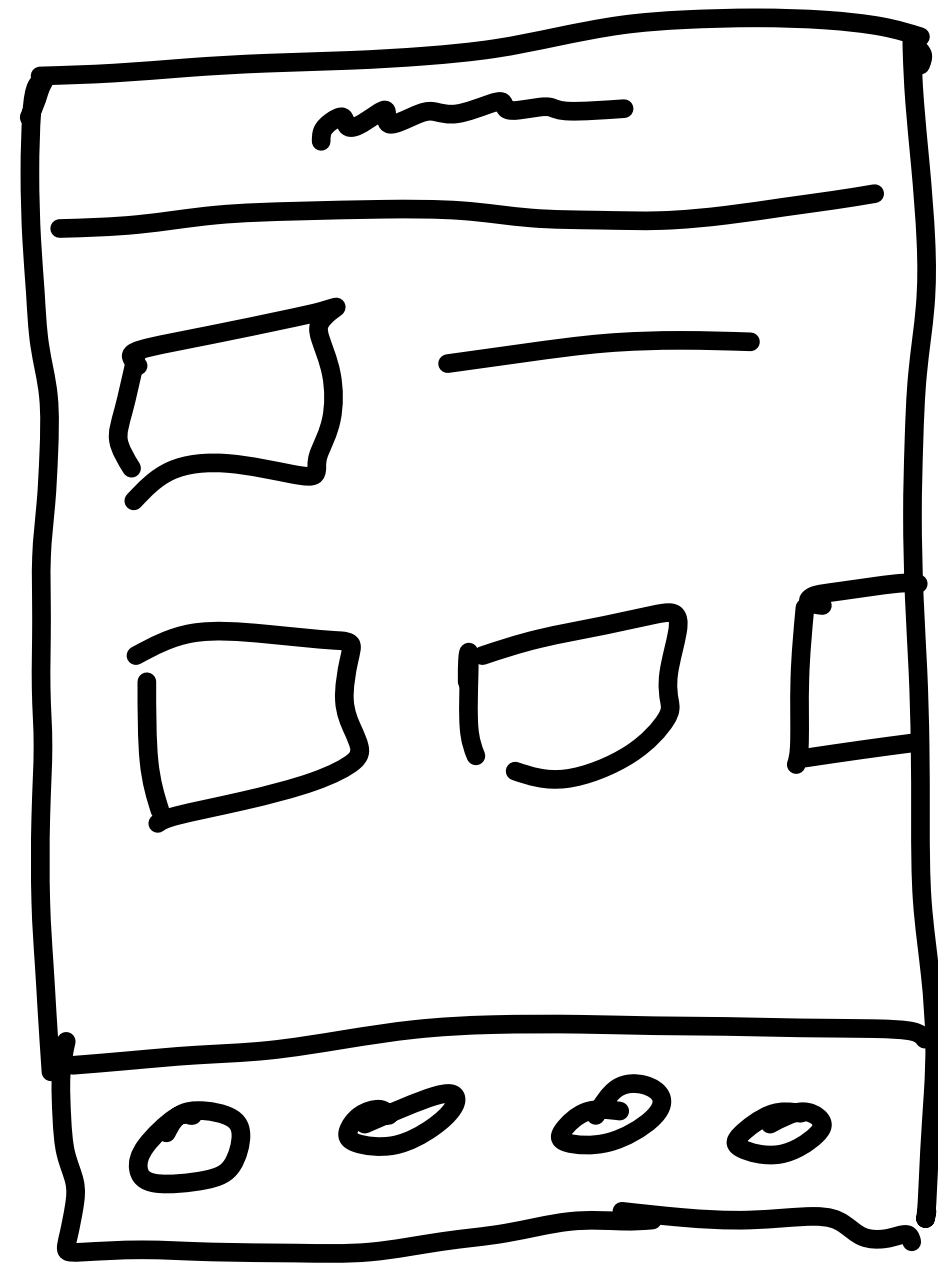
Стартап

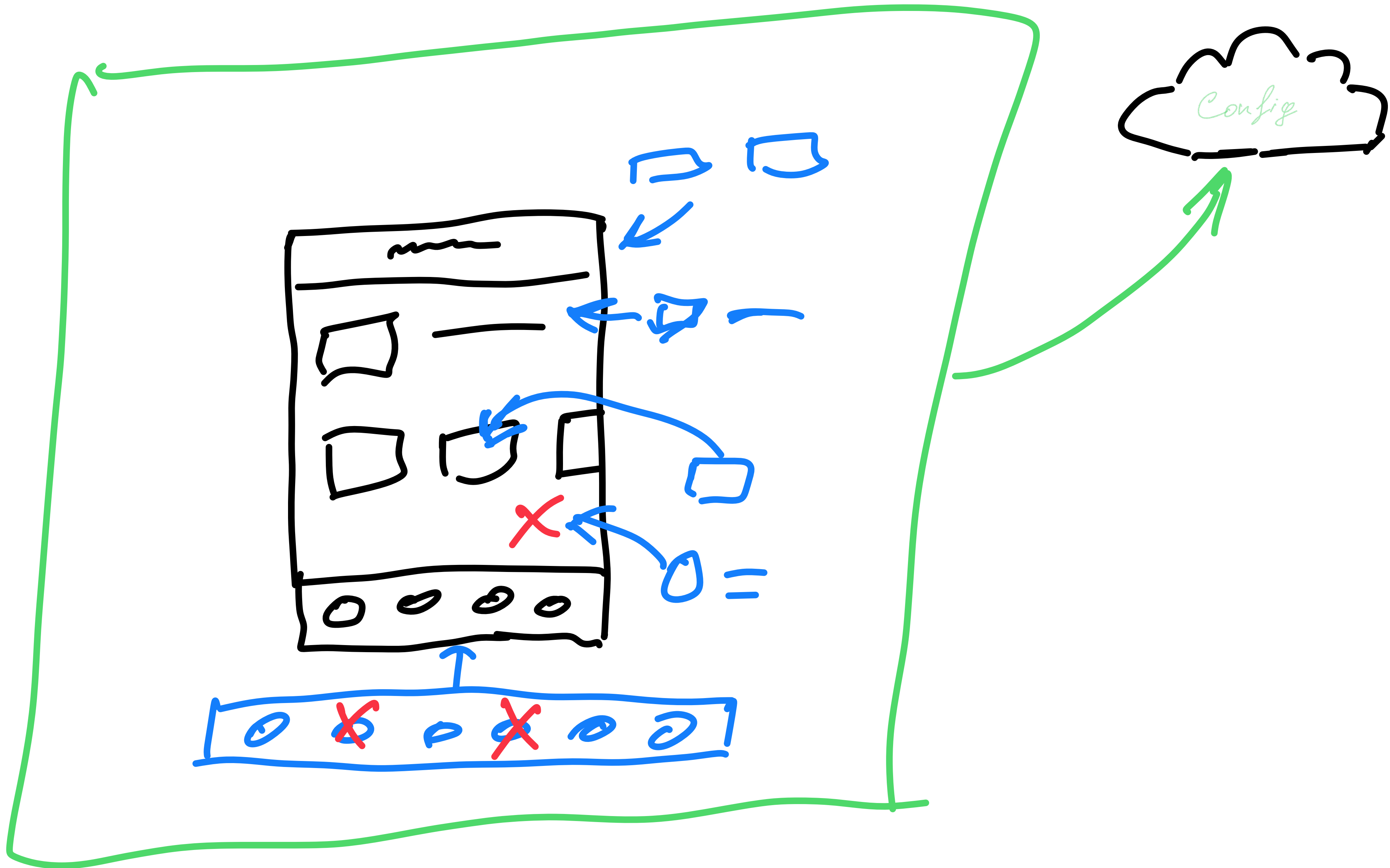
- Ручной деплой на `git pull`
- Ручное обновление зависимостей (`composer` все ломает)
- Недоскрам
- Сильно связанный монолит
- Не соблюдена детерминированность
- Нет веб версии, только мобильные приложения

**Как повысить MTTR на
мобилках?**

Важно

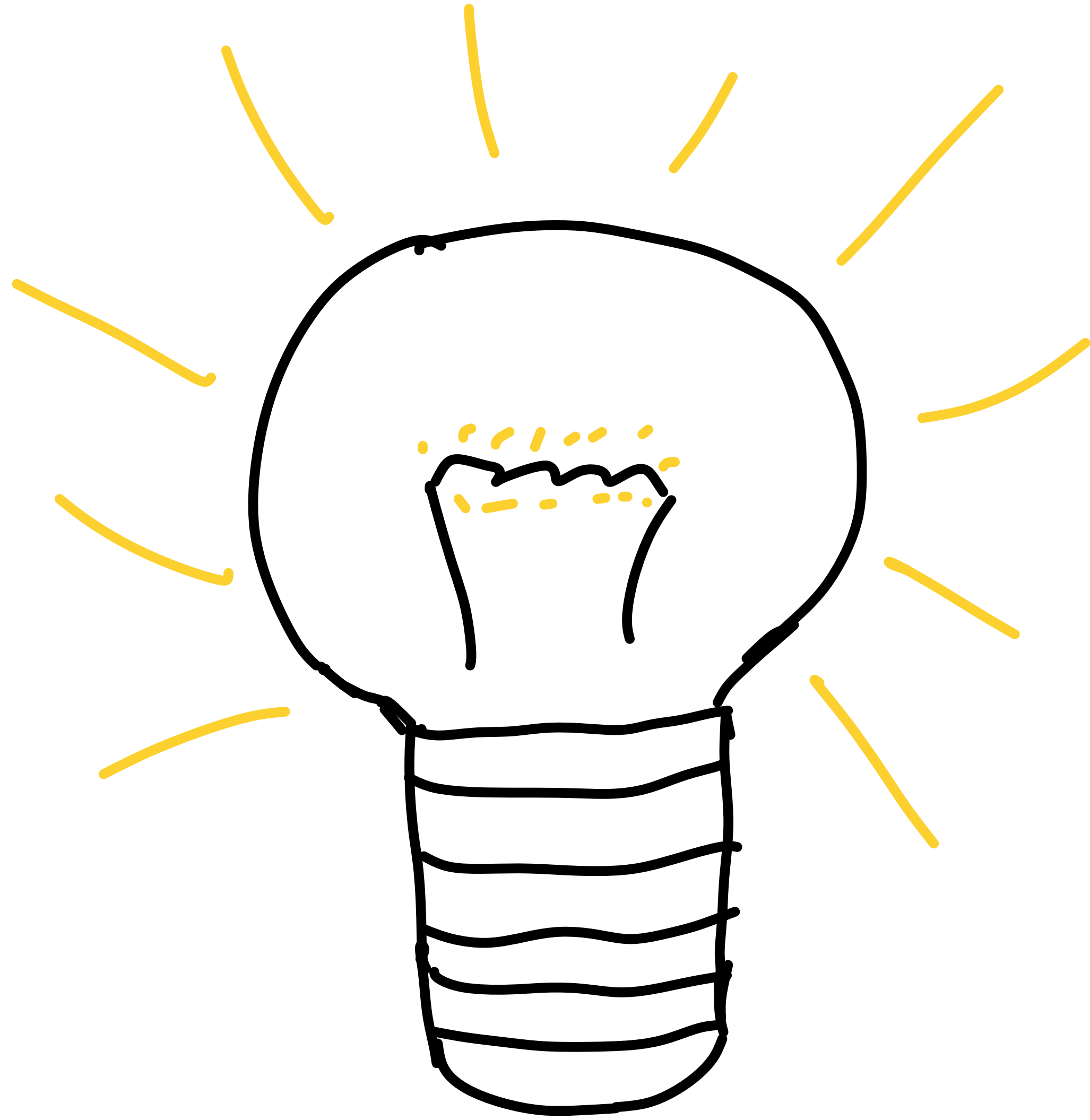
Создавайте системы предоставляющие пользовательский опыт, а
не системы являющиеся им





Стартап

- Облако, k8s, gitlab
- Микросервисы на Kotlin
- Kanban method
- Feature Flags



Analytics

□ □ □
□ □ □

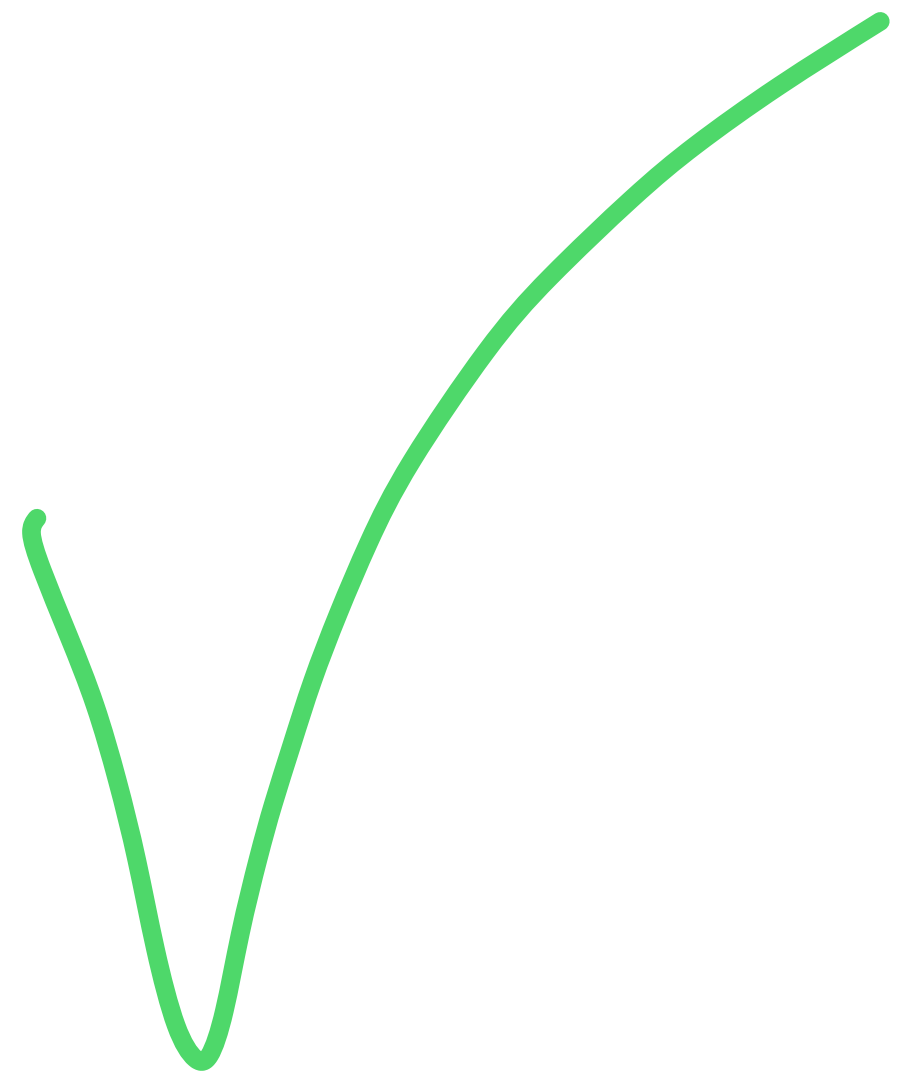
Dev

□ □
□ □
□

Test

□ □
□ □
□ □
□ □
□ □

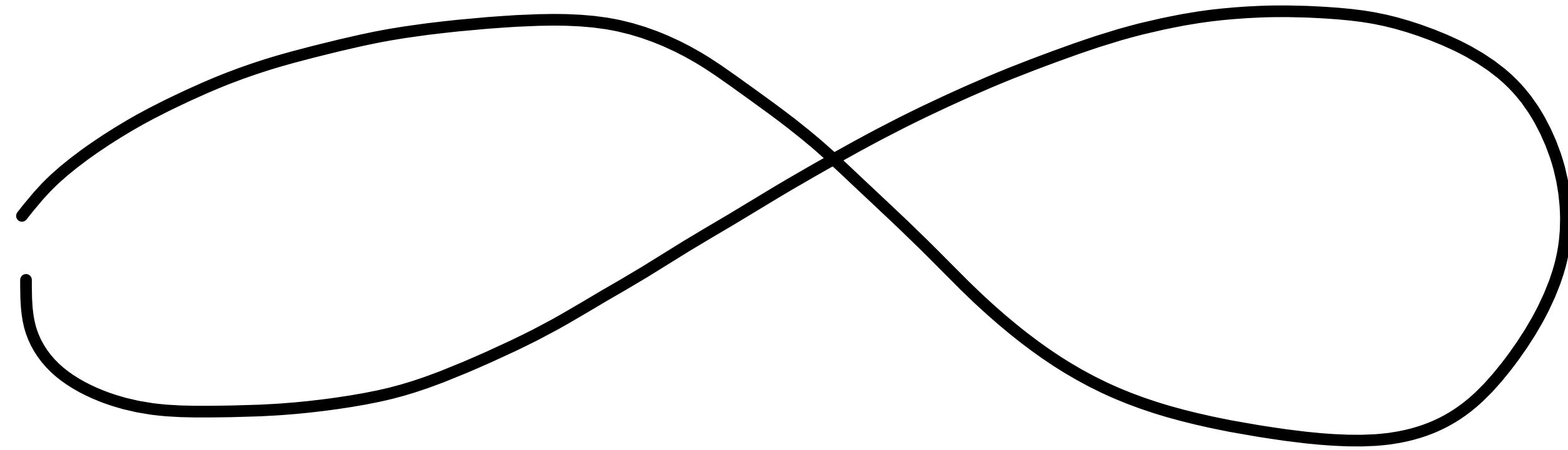
Done



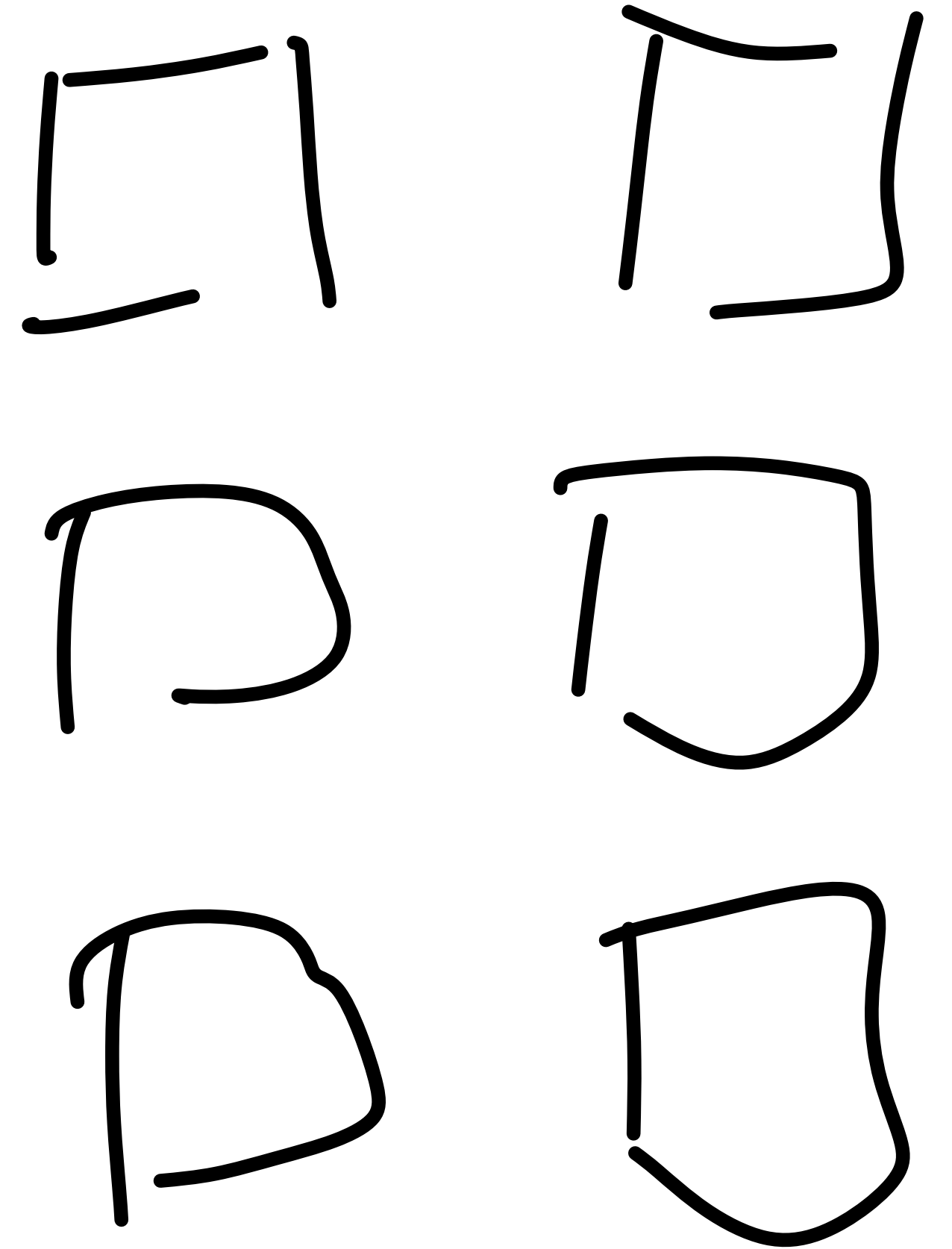
Ready for Dev

Dev

Ready for Dev



Dev



Закон Литтла

$$\left[\begin{array}{l} \text{Время} \\ \text{в системе} \end{array} \right] = \frac{\left[\begin{array}{l} \text{Число заявок} \\ \text{в системе} \end{array} \right]}{\left[\begin{array}{l} \text{Пропускная} \\ \text{Способность} \end{array} \right]}$$

Analytics

□ □ □
□ □ □

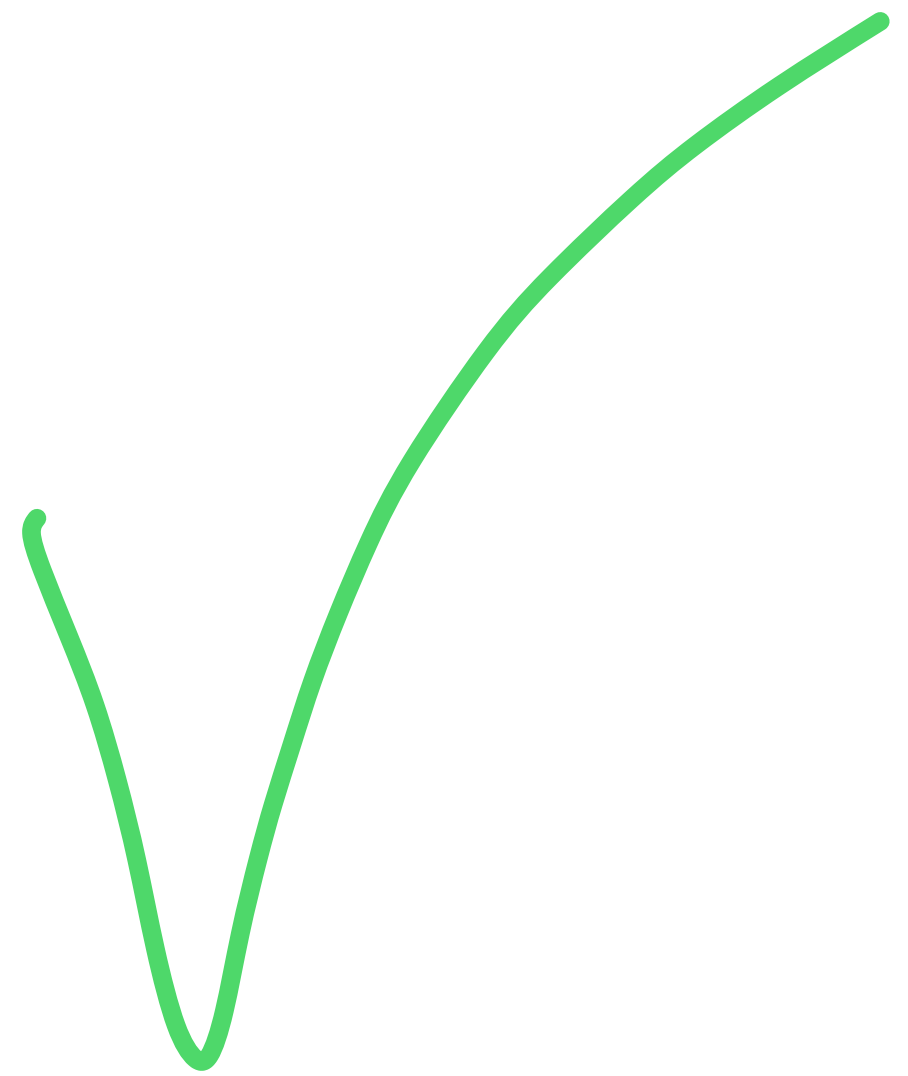
Dev

□ □
□ □
□

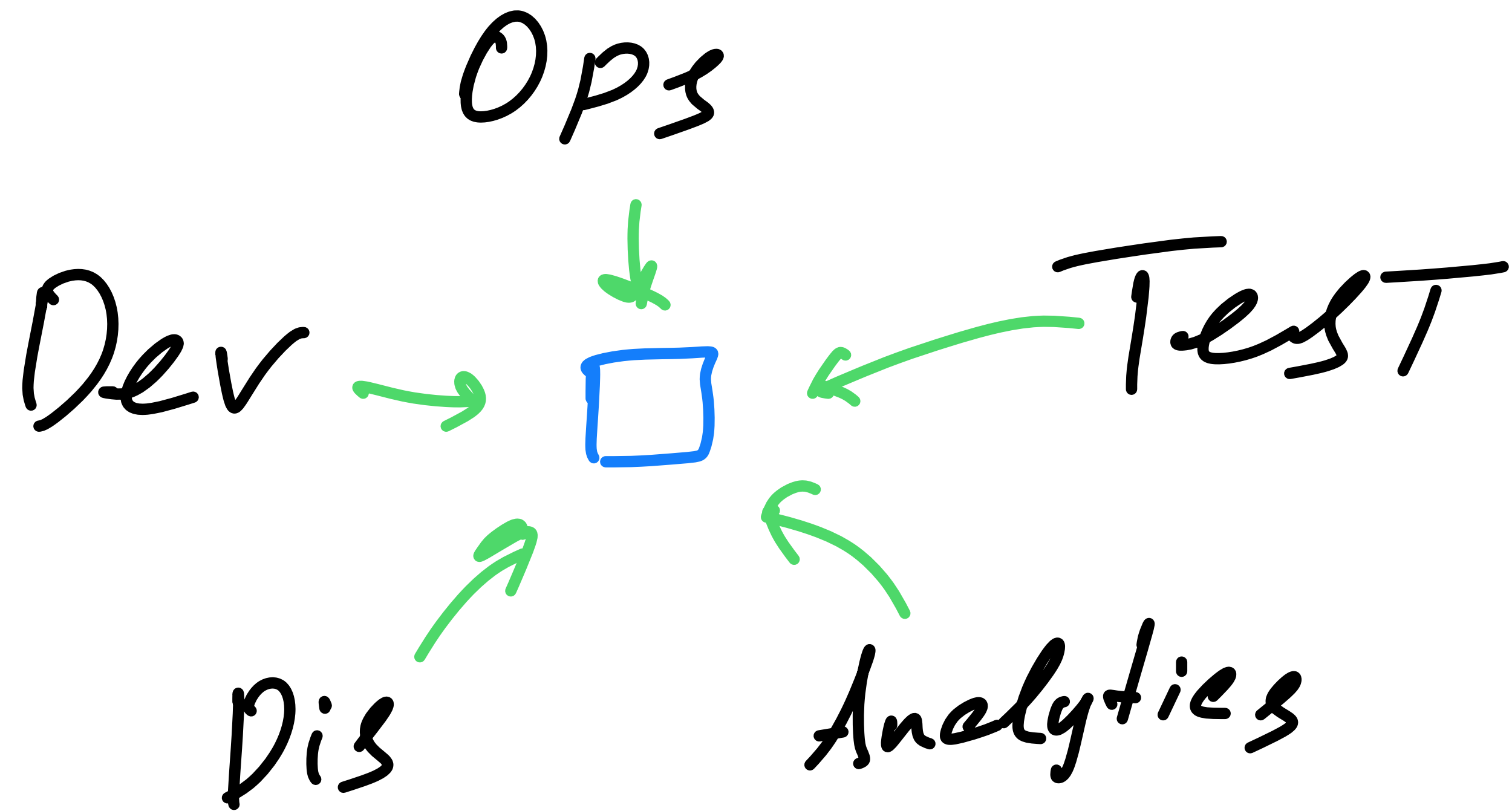
Test

□ □
□ □
□ □
□ □
□ □

Done



Spec Ops



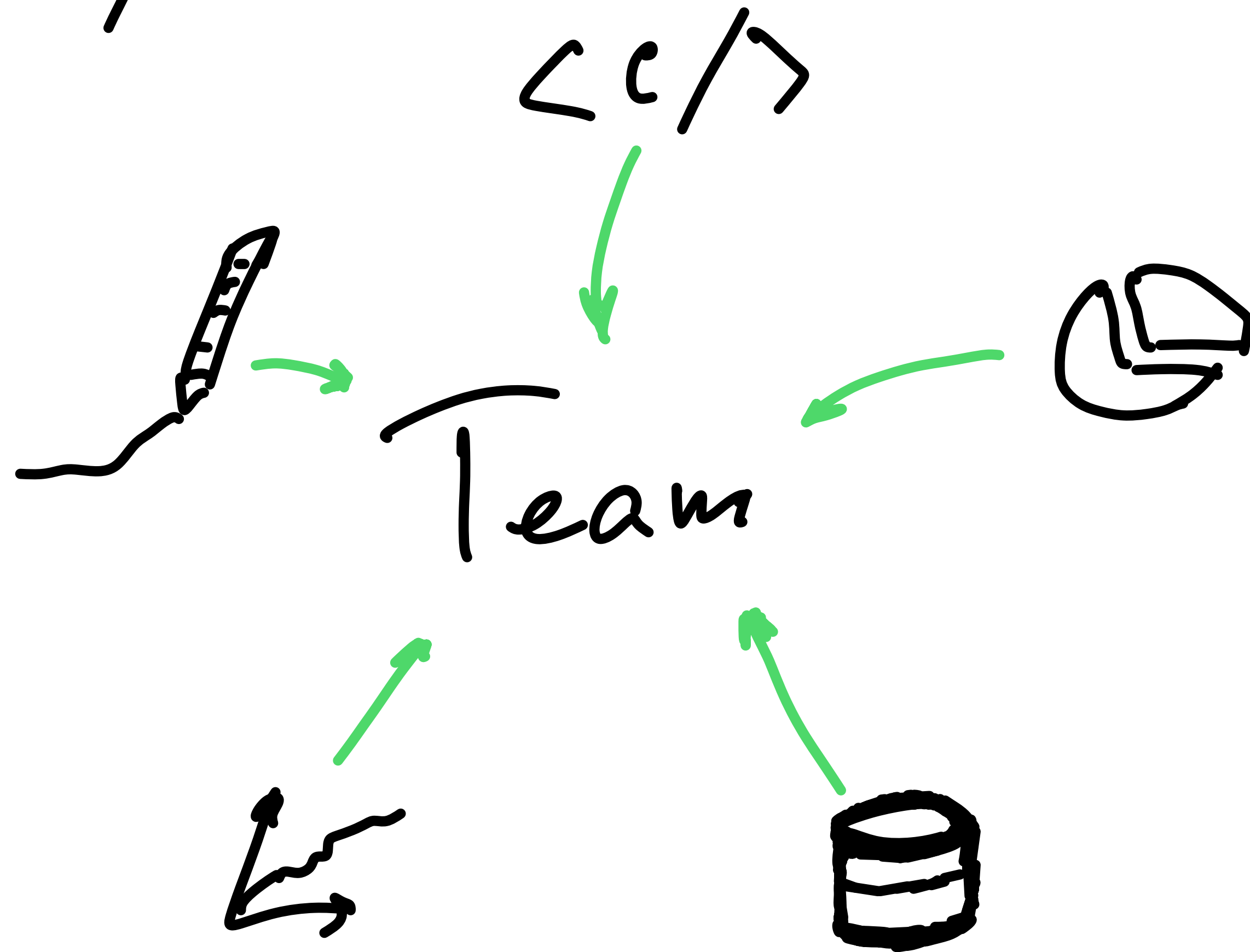
Spec Ops



Vitaly D
@vdvision

...

Nowadays there is no team, but a bunch of people called product team for some reason. Make no mistake: only people working at the same time, in the same space, on the same thing can produce value efficiently. That's what team really is.



Важно

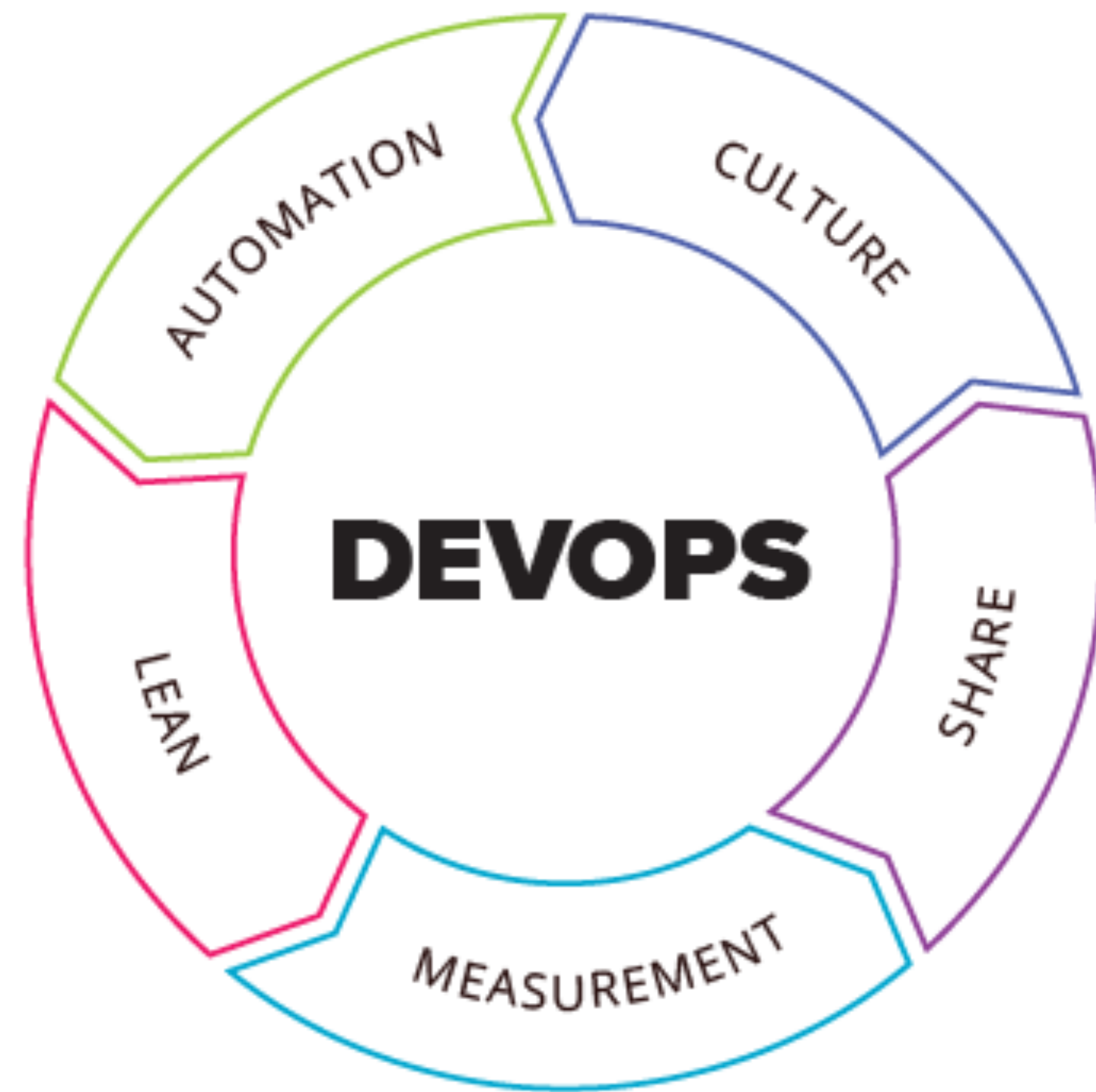
- Этапы проверок и согласований только увеличивают риски будущих сбоев
- Качество достигается улучшением стабильного процесса
- Создавайте системы предоставляющие пользовательский опыт, а не являющиеся им



DevOps

DevOps everywhere

imgflip.com



Будьте DevOps