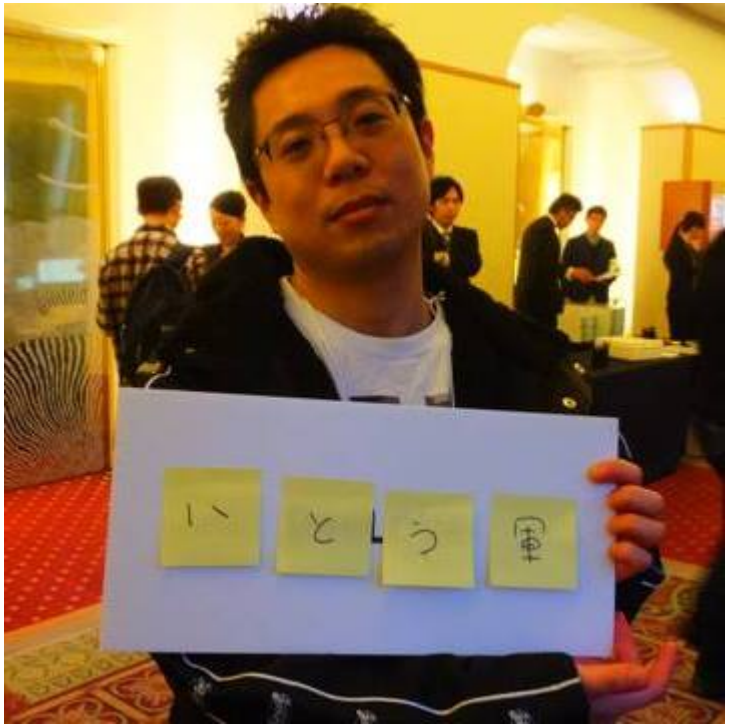
Technology-Driven Development: Using Automation and Development Techniques to Grow an Agile Culture

Jul/29/2014
Hiroyuki Ito
Development Process Optimization Department, Rakuten, Inc.
http://www.rakuten.co.jp/

楽Ⓡ天 ⓇRakuten

About me

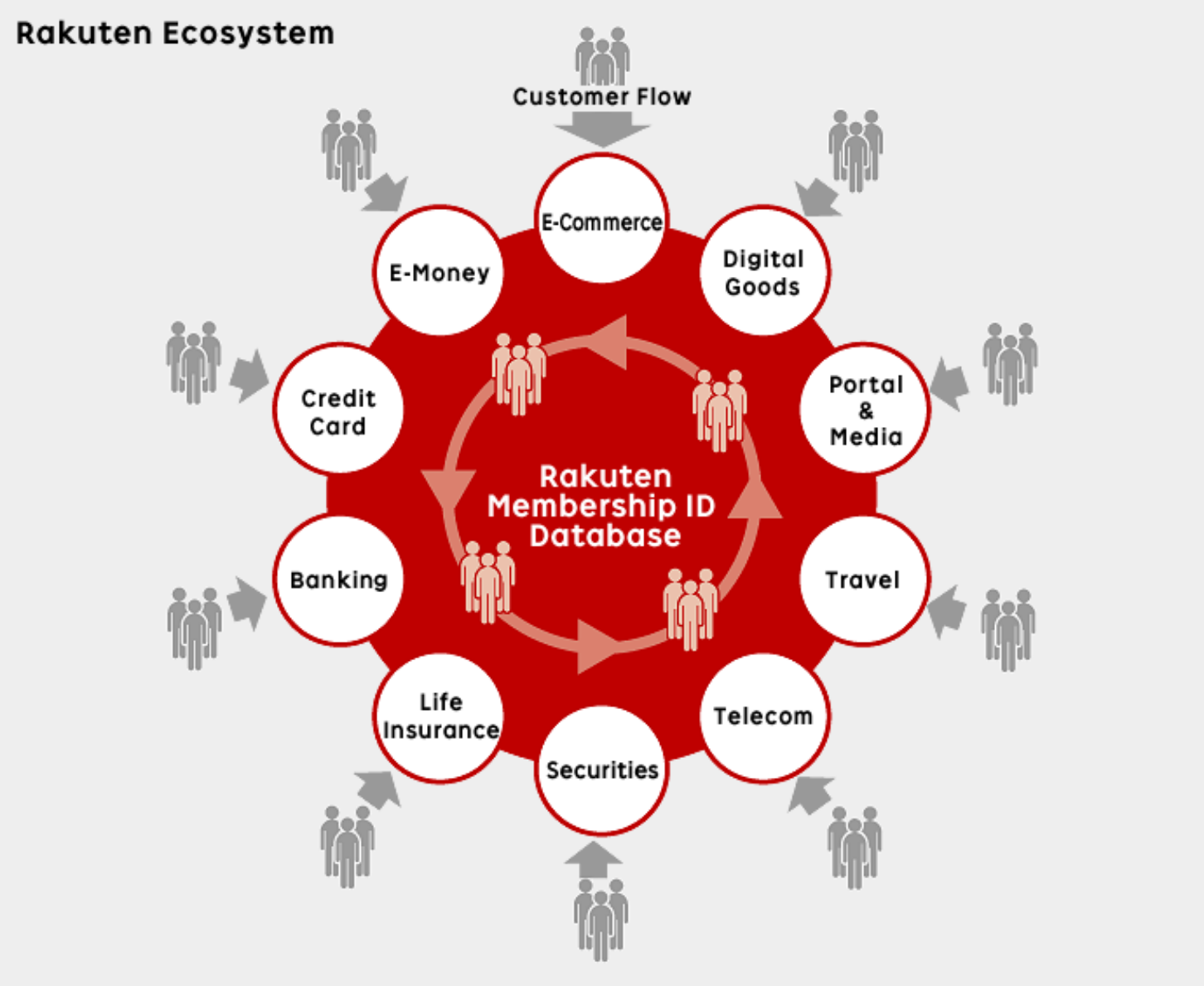


Hiroyuki Ito (The Hiro)

@hageyahhoo

Test-Driven Development Group

楽天 Rakuten



http://global.rakuten.com/corp/about/strength/business_model.html

楽天 Rakuten

It’s my 3rd time to be here!

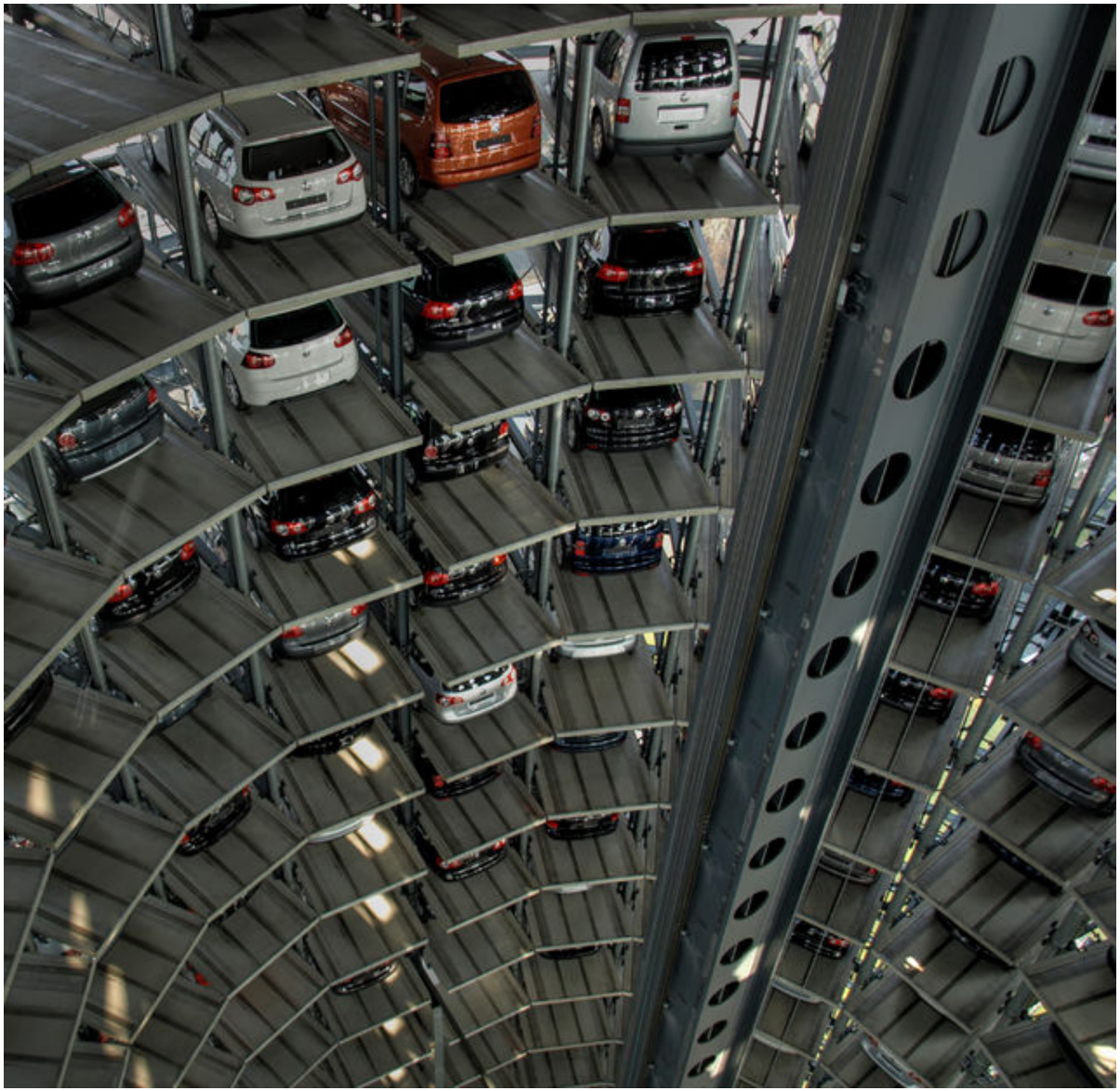
Agile2014 : as a Speaker

楽天 Rakuten

This session's theme

Technology-Driven Development

Additional possibilities of automation



“TDD” stands for three purposes

Efficiency

Learning

Collaboration

By three approaches

CI/CD

TDD

BDD

楽天 Rakuten

Three approaches by

CI/CD

TestFlight

iOS Beta Testing On The Fly

TDD

BDD

楽天 Rakuten

Agenda

1. Conditions and Challenges
2. CI/CD
3. TDD
4. BDD
5. Results, Problems, Possibility and Future
6. Conclusions

楽天 Rakuten

1. Conditions and Challenges

2. CI/CD

3. TDD

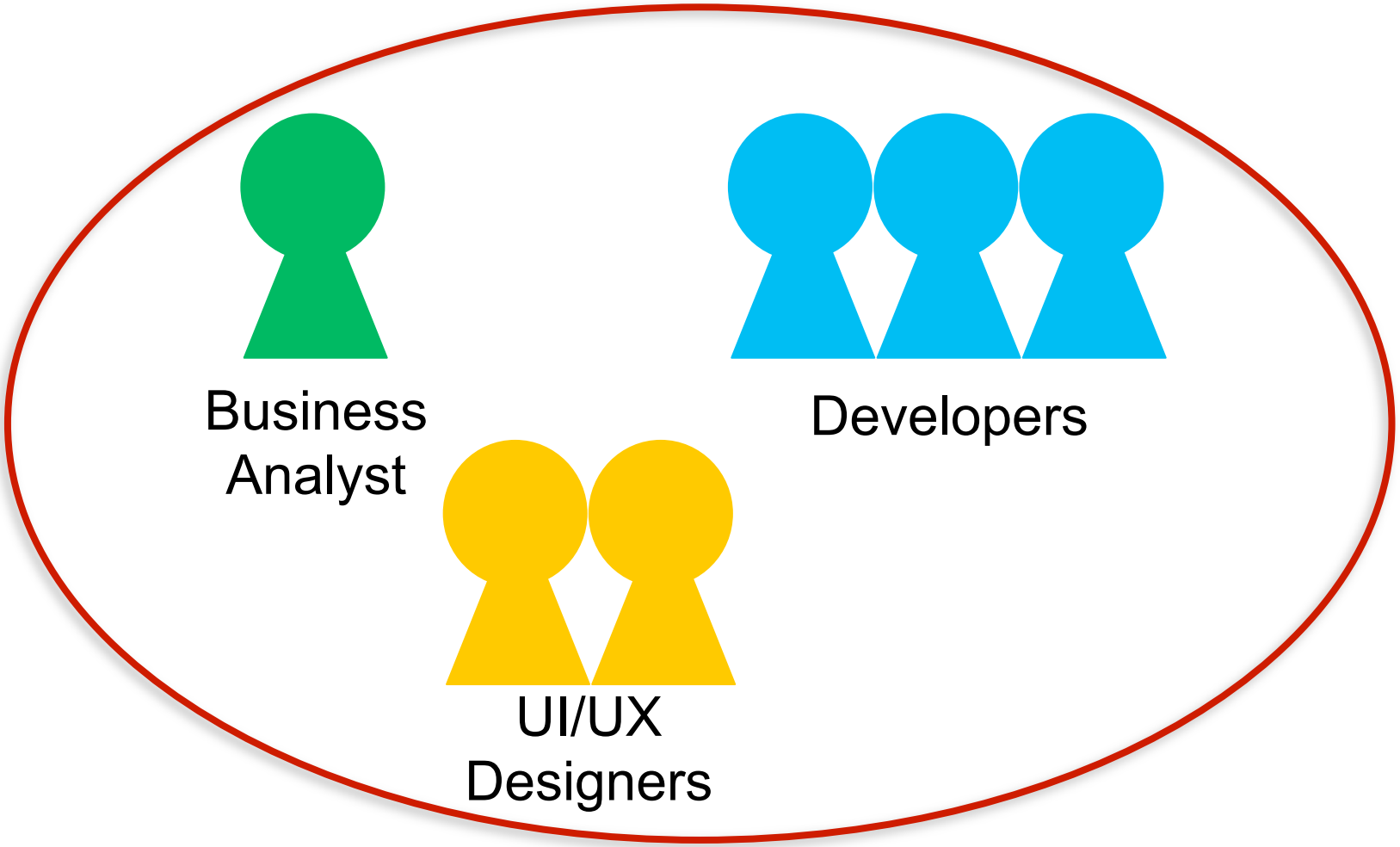
4. BDD

5. Results, Problems, Possibility and Future

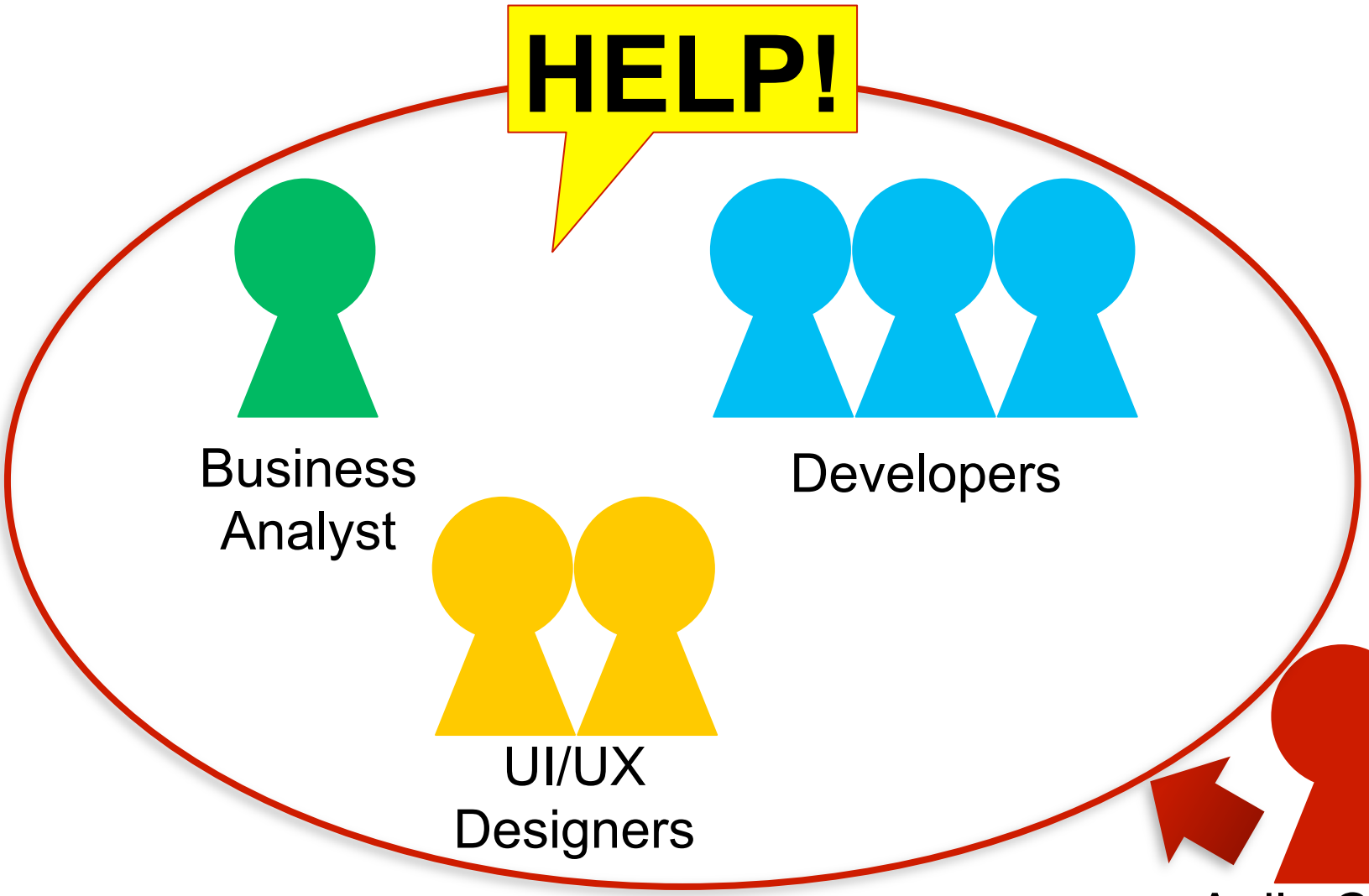
6. Conclusions

楽天 Rakuten

At the end of April 2013



At the end of April 2013



Our target application is

Conditions and Challenges

楽R天 RRakuten

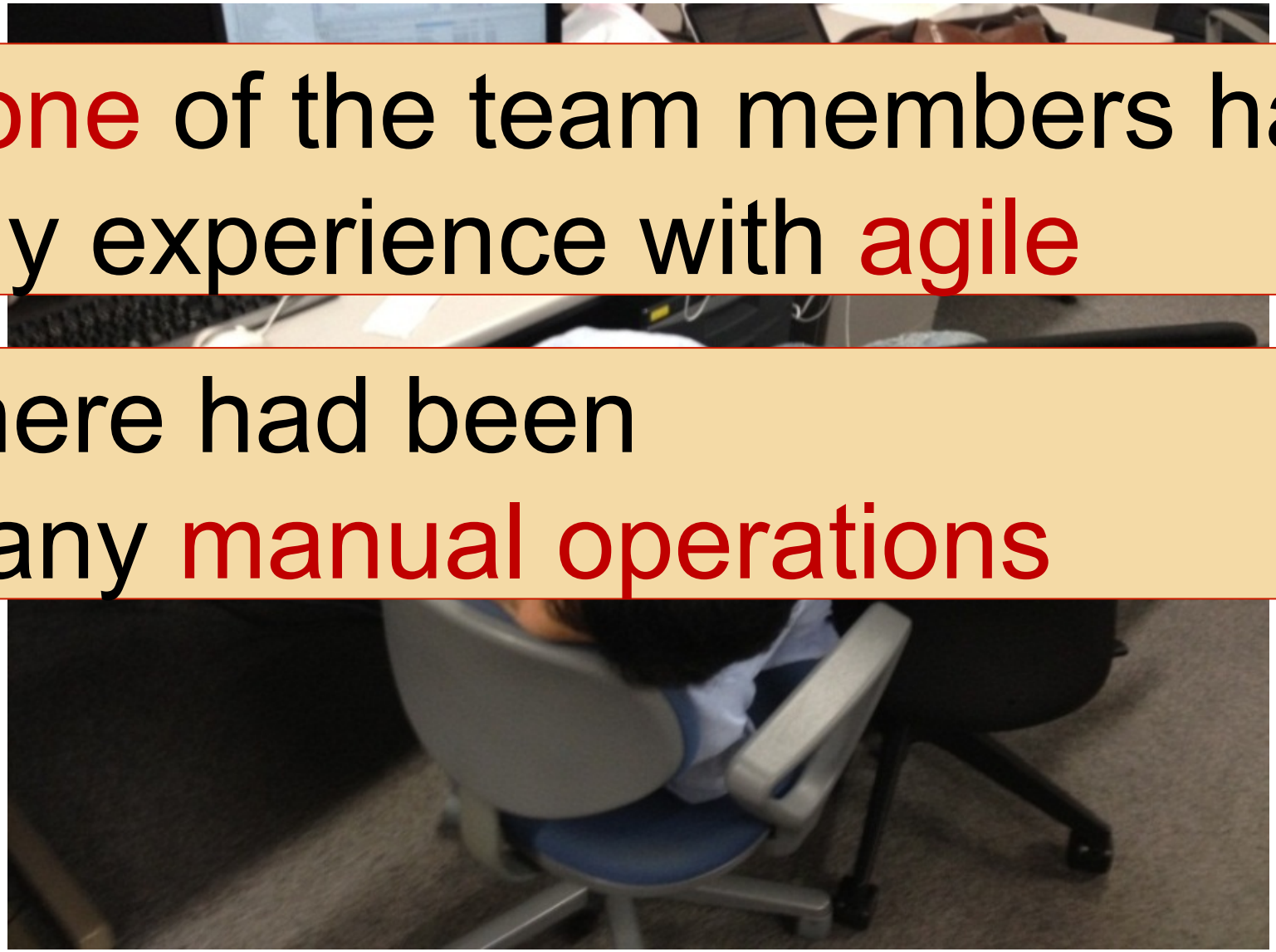
Conditions and Challenges

None of the team members had any experience with agile

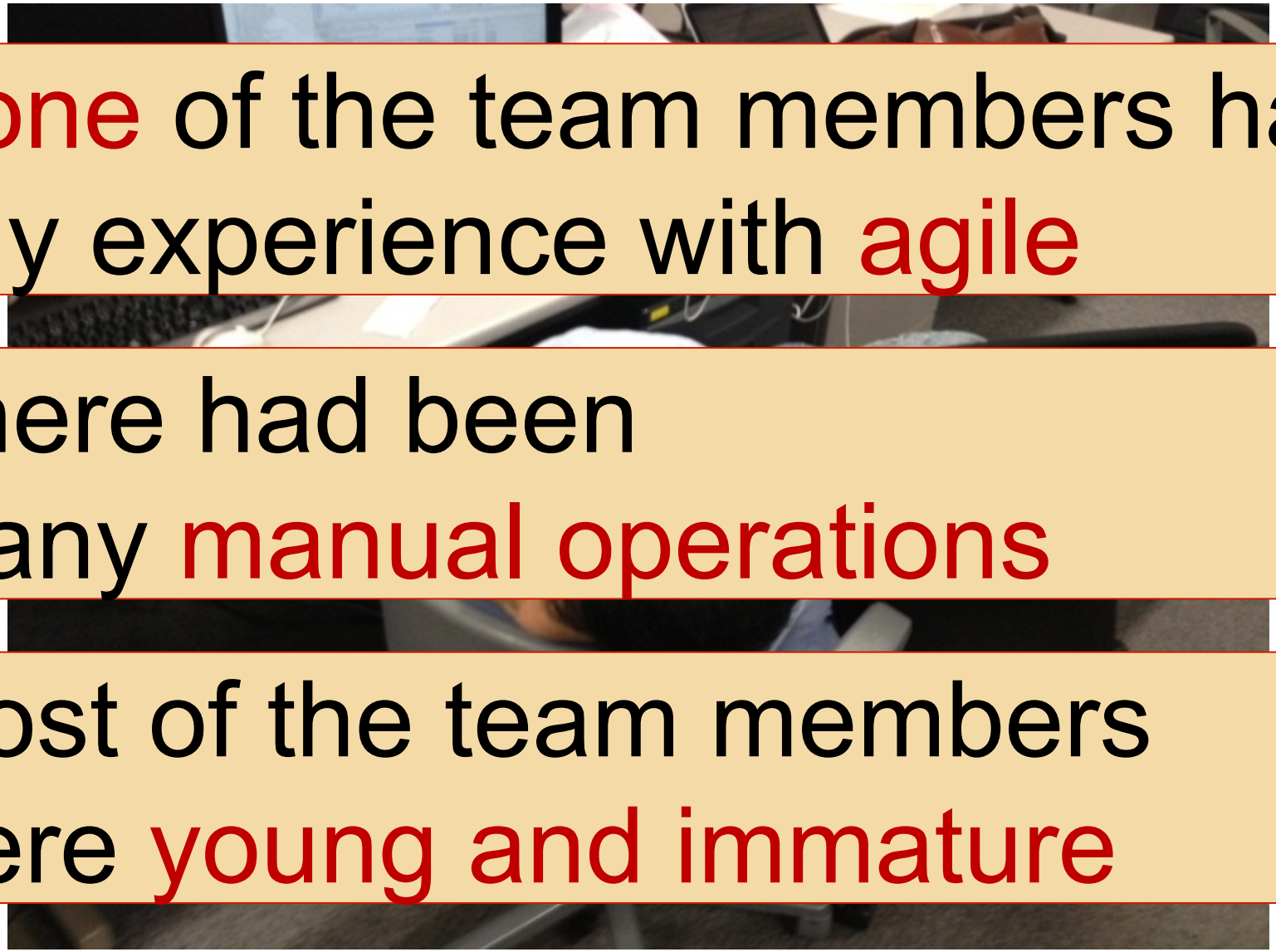
Conditions and Challenges

None of the team members had any experience with agile

There had been many manual operations



Conditions and Challenges



None of the team members had any experience with agile

There had been many manual operations

Most of the team members were young and immature

What do you think?

I was so much excited!

WHY?

I can
achieve **anything**
through such a
challenging project!

Three approaches

CI/CD

TDD

BDD

1. Conditions and Challenges

2. CI/CD

3. TDD

4. BDD

5. Results, Problems, Possibility and Future

6. Conclusions

Challenges

Low performance

- So many manual tasks

Going in circles

- No clear vision and no requirements
- No timely progress information

Before CI/CD

13.5 hours/week

- Change requests : 3 times/week
- Regression testing : **4 hours**/change
  - Need to retry if we find bugs…
- Install applications : **0.5 hour**/change
  - 5-minite work for 6 persons

The Implementation of CI/CD in our project

The Implementation of CI/CD in our project

My PC

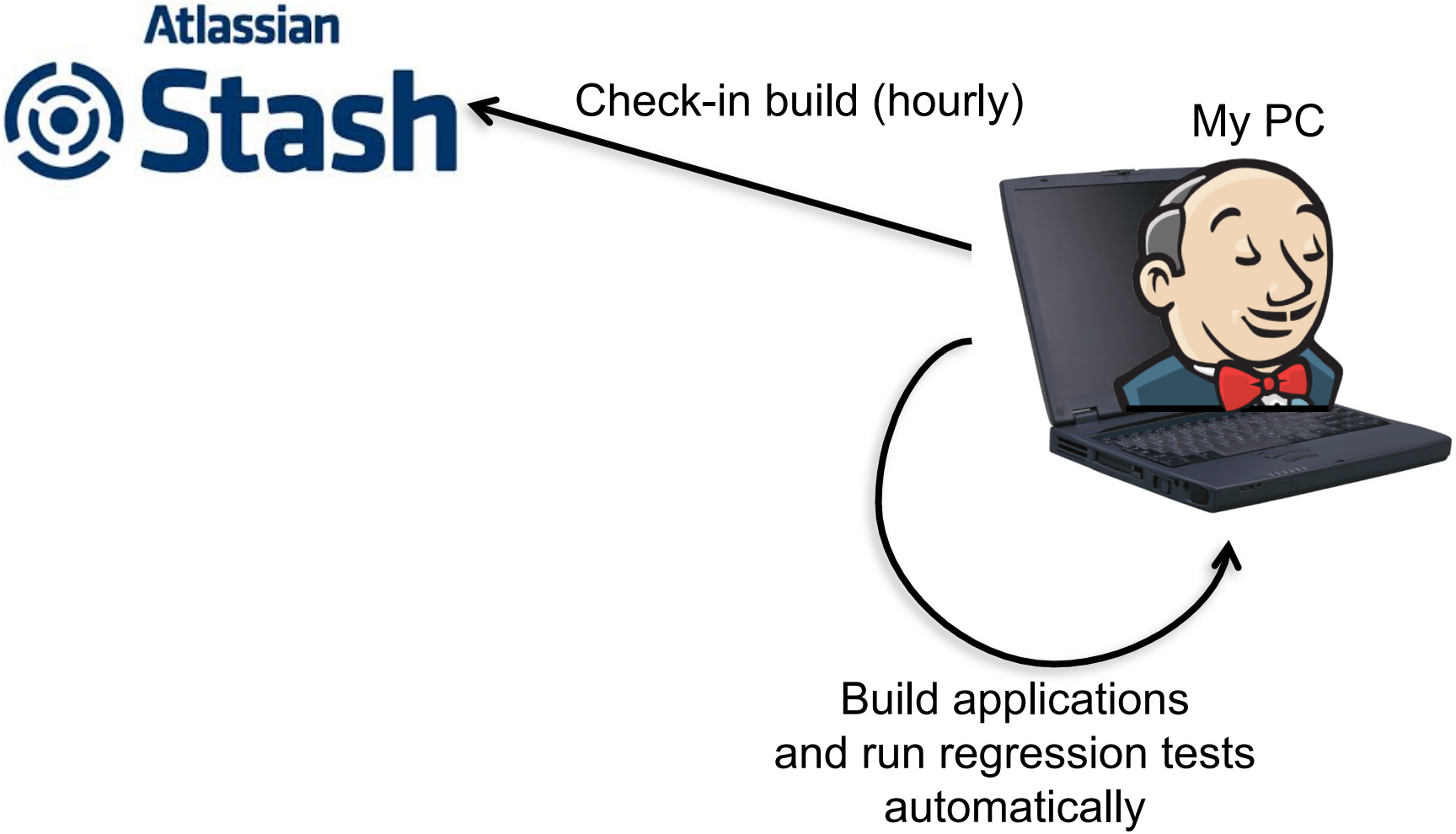
The Implementation of CI/CD in our project

Atlassian Stash

Check-in build (hourly)

My PC

The Implementation of CI/CD in our project



The Implementation of CI/CD in our project

Atlassian Stash

Check-in build (hourly)

My PC

Deliver to
all team members
automatically

TestFlight
iOS Beta Testing On The Fly

Build applications
and run regression tests
automatically

楽天 Rakuten

The Implementation of CI/CD in our project

Atlassian Stash

Check-in build (hourly)

My PC

Deliver to
all team members
automatically

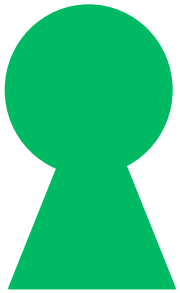
TestFlight
iOS Beta Testing On The Fly

楽天 Rakuten

Build applications
and run regression tests
automatically

We demonstrate latest application
to the business analyst and managers
in every daily scrum

Shared understanding by the working software



Business Analyst

Developers

Know about the progress

Get fast feedback

UI/UX Designers

楽天 Rakuten

After CI/CD

15 minutes/week

- Change requests : 3 times/week
- Regression testing : **3 minutes**/change
- Install applications : **2 minutes**/change

After CI/CD

15 minutes/wee

- Cl
- **13 hours!**
- /change
- pplications : **2 minutes**/change

楽R天 RRakuten

1. Conditions and Challenges

2. CI/CD

3. TDD

4. BDD

5. Results, Problems, Possibility and Future

6. Conclusions

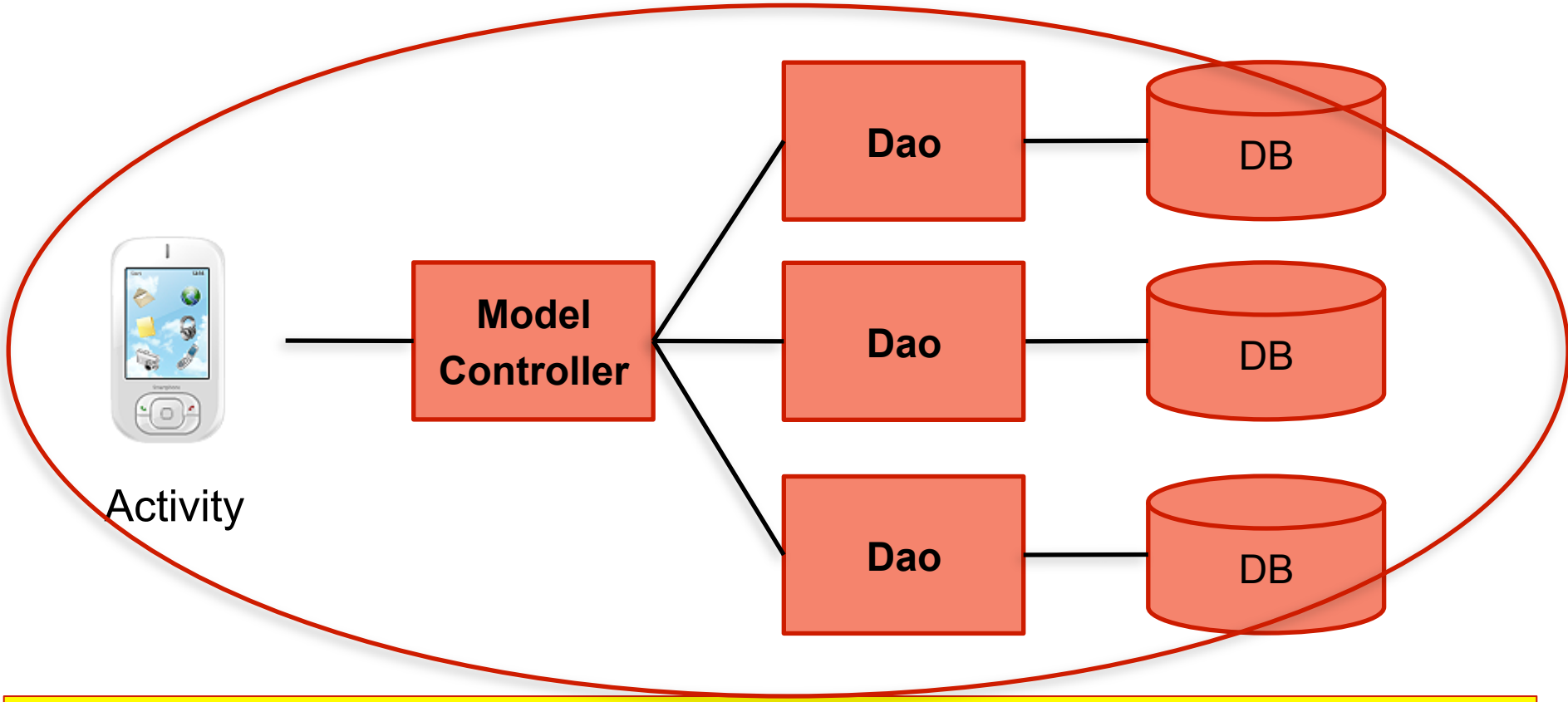
楽天 Rakuten

Challenges

Lack of skill and knowledge of Android

- the architecture of Android
- how to develop the Android application
- how to access the database on the device
- how to implement the UI

Before TDD

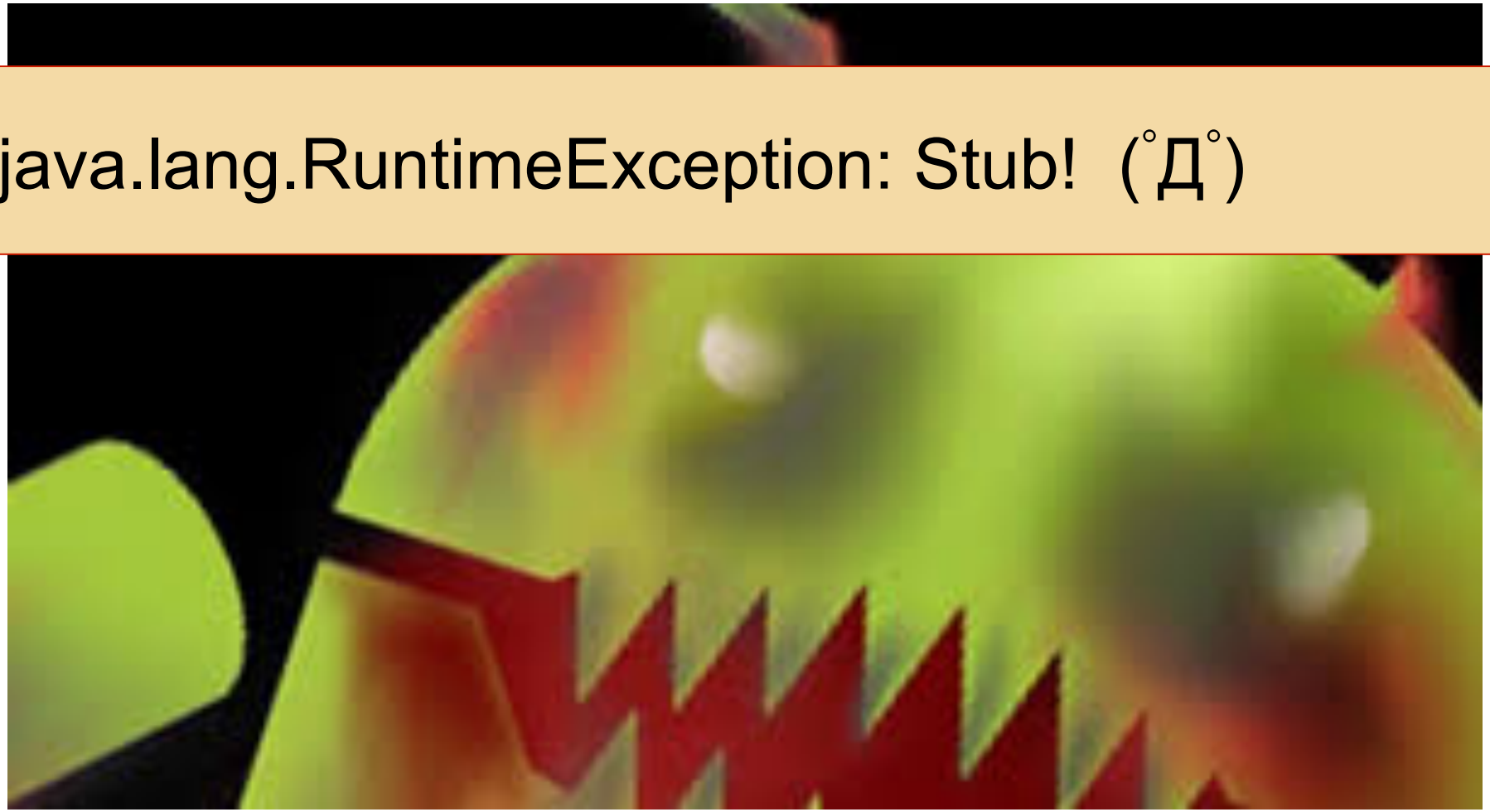


- Could not test after we implemented **all components (Debug Later Programming)**
- It took **five days** to implement one activity set

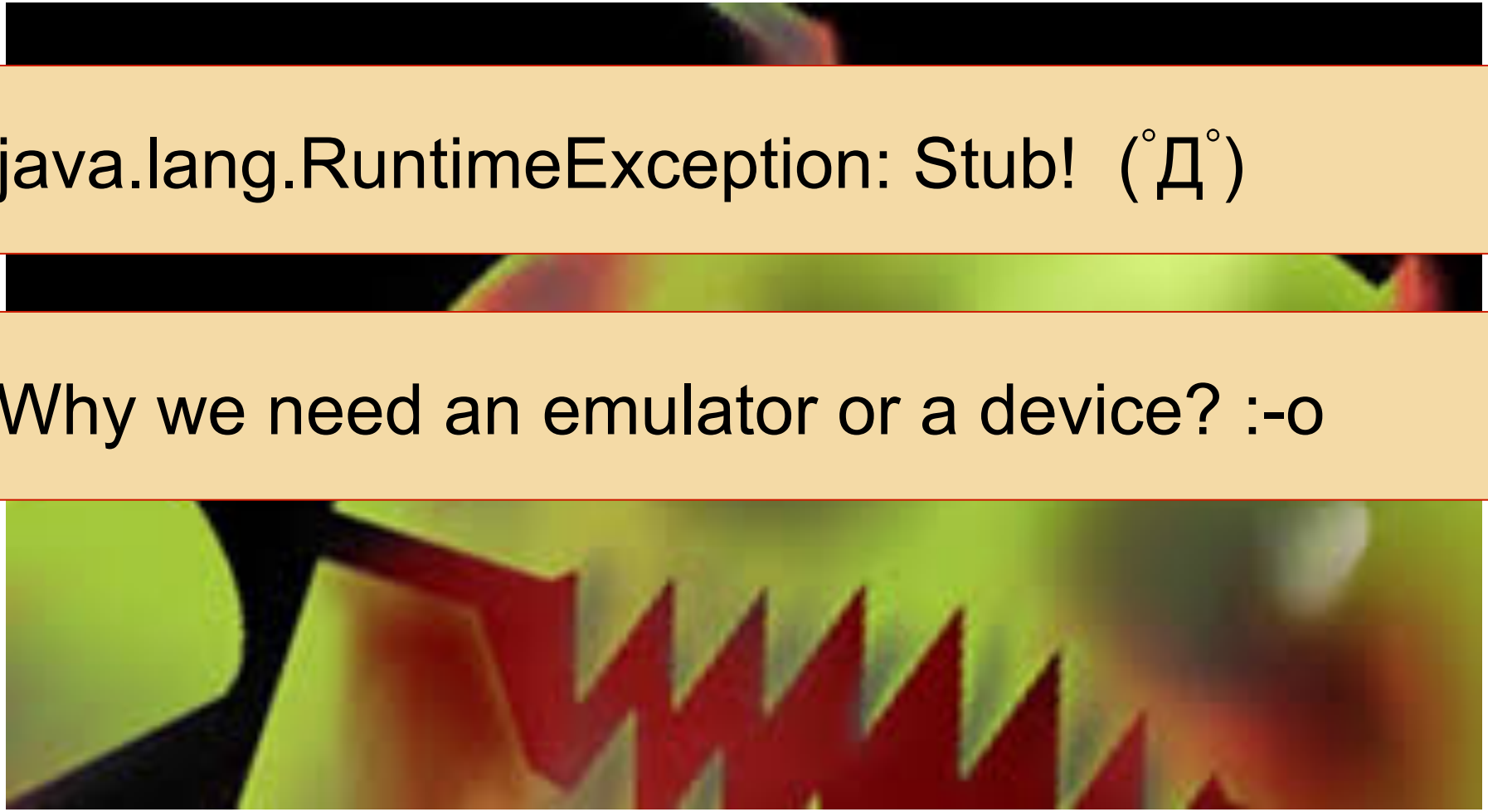
Too difficult to use Android JUnit 😐

Too difficult to use Android JUnit 😐

java.lang.RuntimeException: Stub! (゜Д゜)



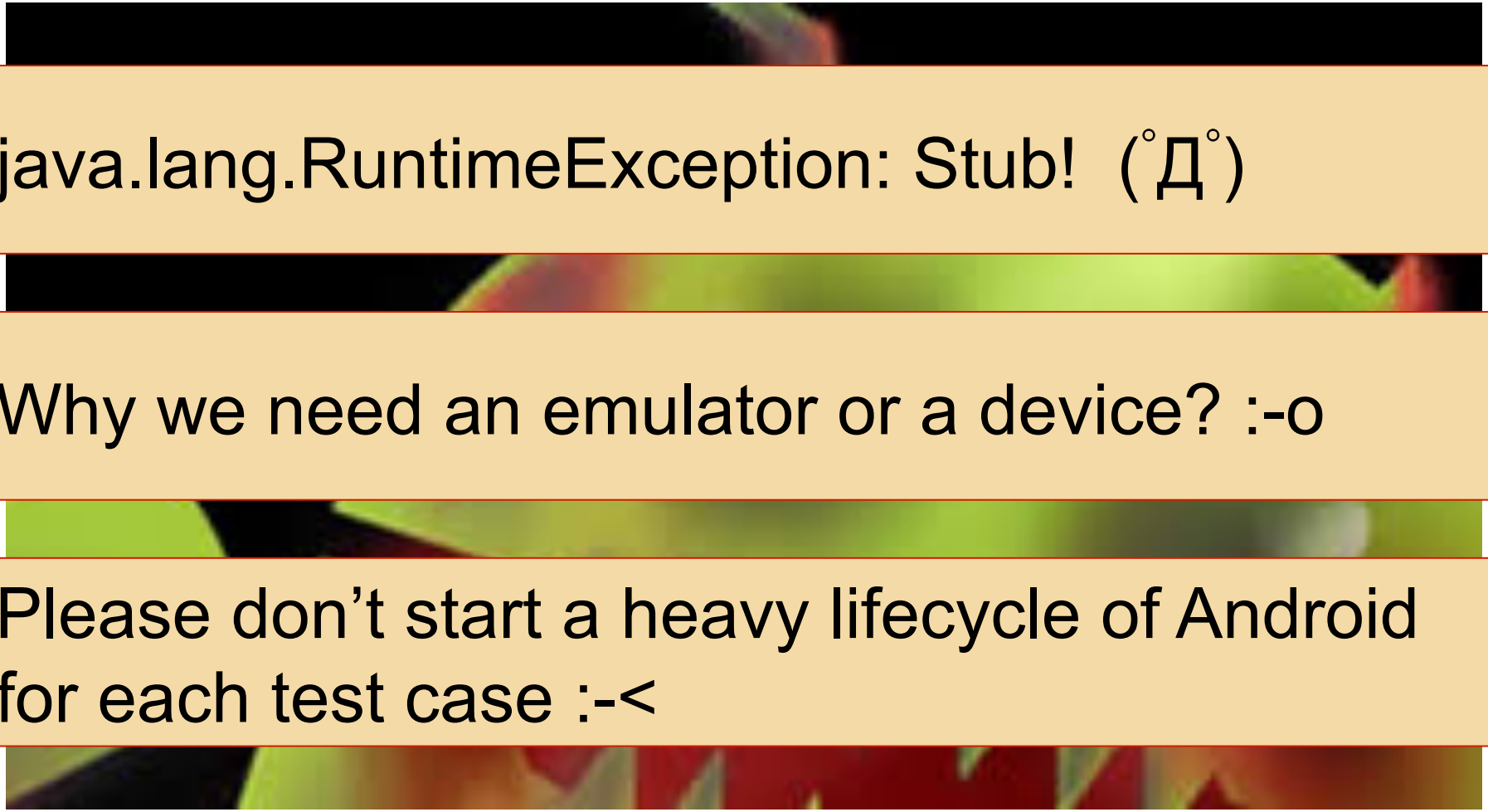
Too difficult to use Android JUnit 😐



java.lang.RuntimeException: Stub!　(゜Д゜)

Why we need an emulator or a device? :-o

Too difficult to use Android JUnit 😐



java.lang.RuntimeException: Stub! (゜Д゜)

Why we need an emulator or a device? :-o

Please don’t start a heavy lifecycle of Android for each test case :-<

Solution to do TDD on Android

Solution to do TDD on Android

- **Robolectric** : Do all unit testing only on JVM
  - http://robolectric.org/
  - Without any emulator or device!

Solution to do TDD on Android

- **Robolectric** : Do all unit testing only on JVM
  - http://robolectric.org/
  - Without any emulator or device!
- **Mockito** : Can use the “Test Double”
  - http://code.google.com/p/mockito/

楽天 Rakuten

Image of Unit testing for Dao by using Robolectric

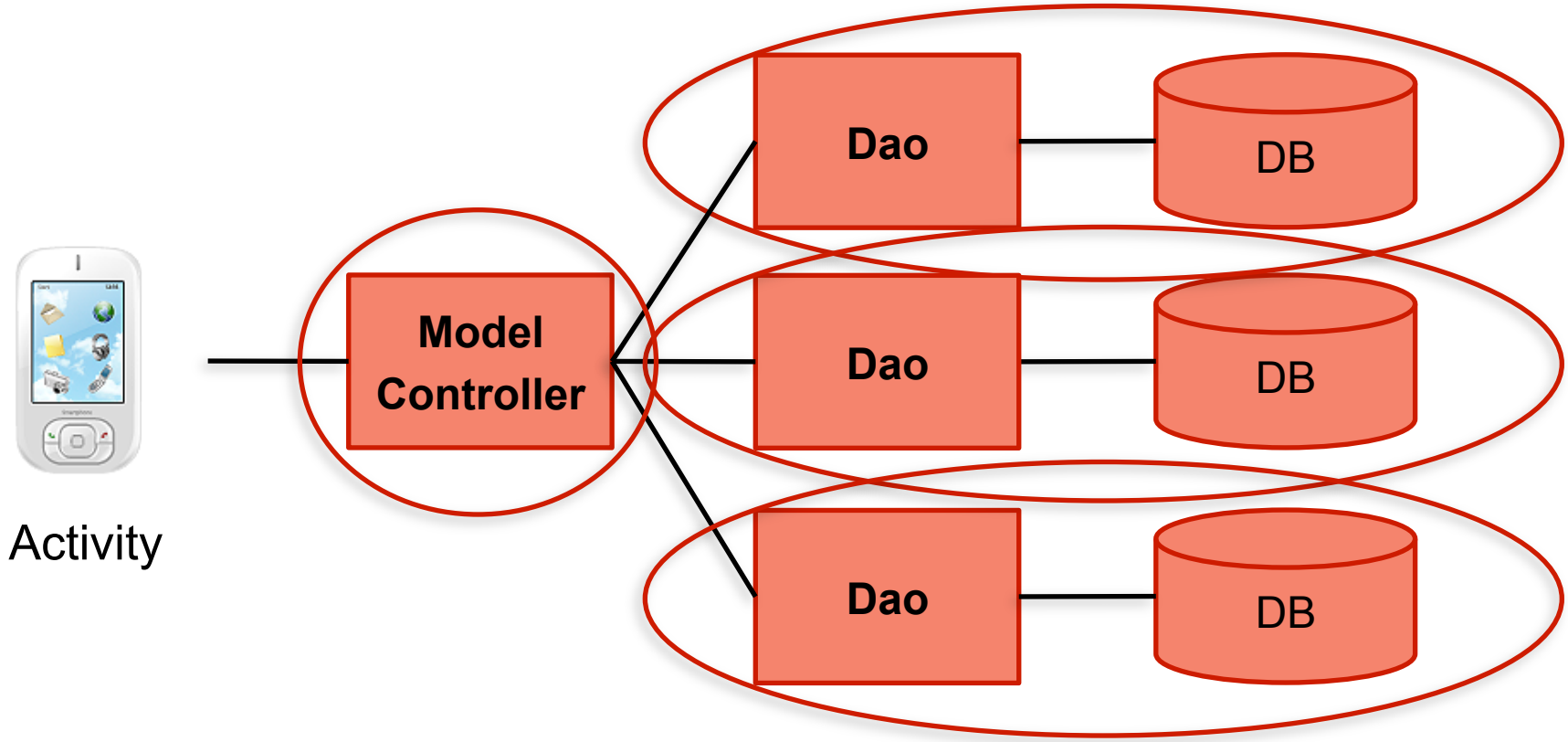
```
@Before
public void setUp() {
        Create database for Test;
        Insert test data;
}

@Test
public void findXxx() {
        Assertions;
}

@After
public void tearDown() {
        Drop Database for Test;
}
```

5 minutes -> **0.5 seconds** to run each test case.

After TDD



- Can test each component **independently and separately**
- It takes **one day** to implement one activity set
  (**five times faster** than at the start of the project)

1. Conditions and Challenges

2. CI/CD

3. TDD

4. BDD

5. Results, Problems, Possibility and Future

6. Conclusions

Challenges

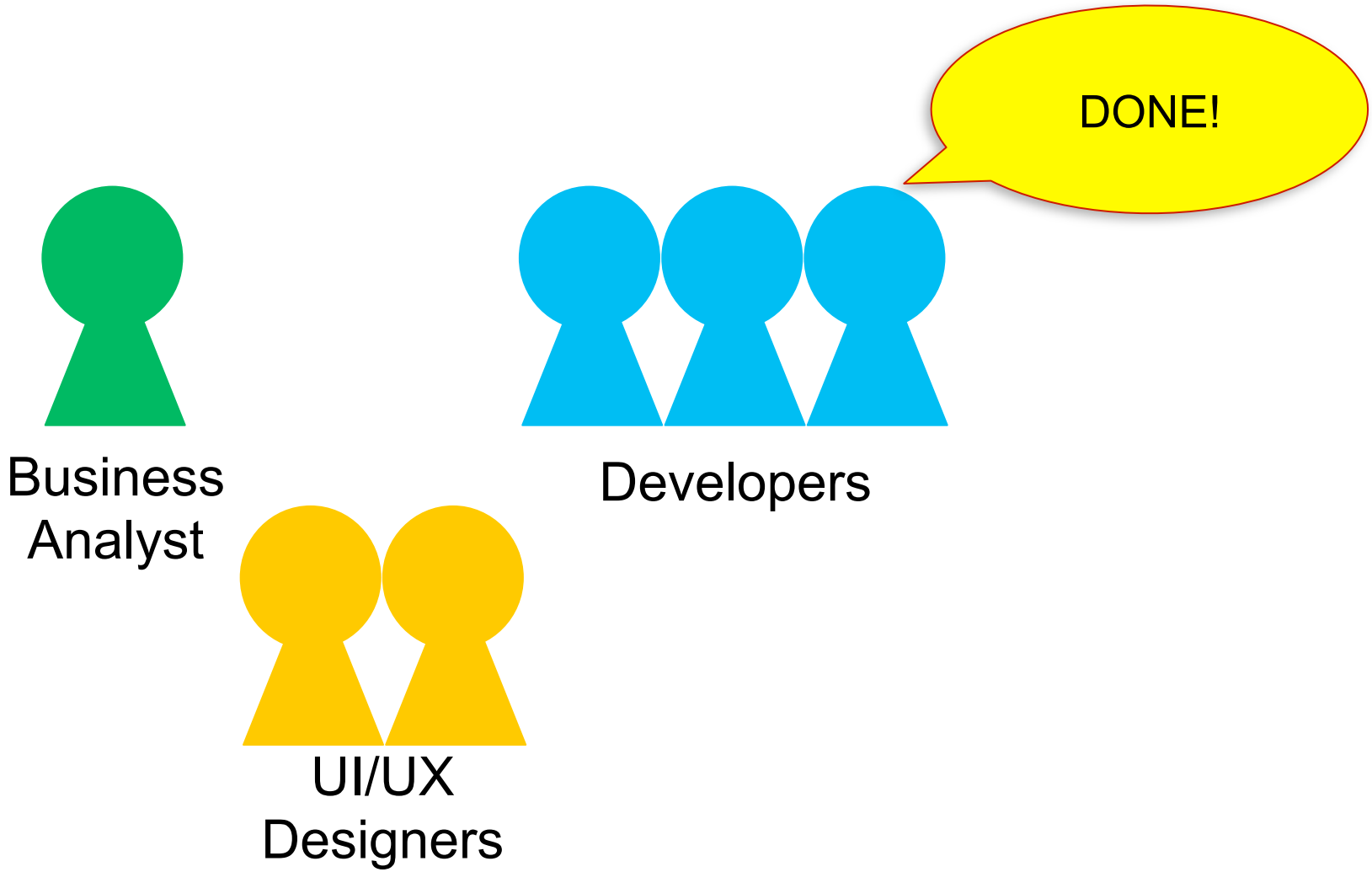
Avoid feature creep

Detect bugs and regressions
on use-cases

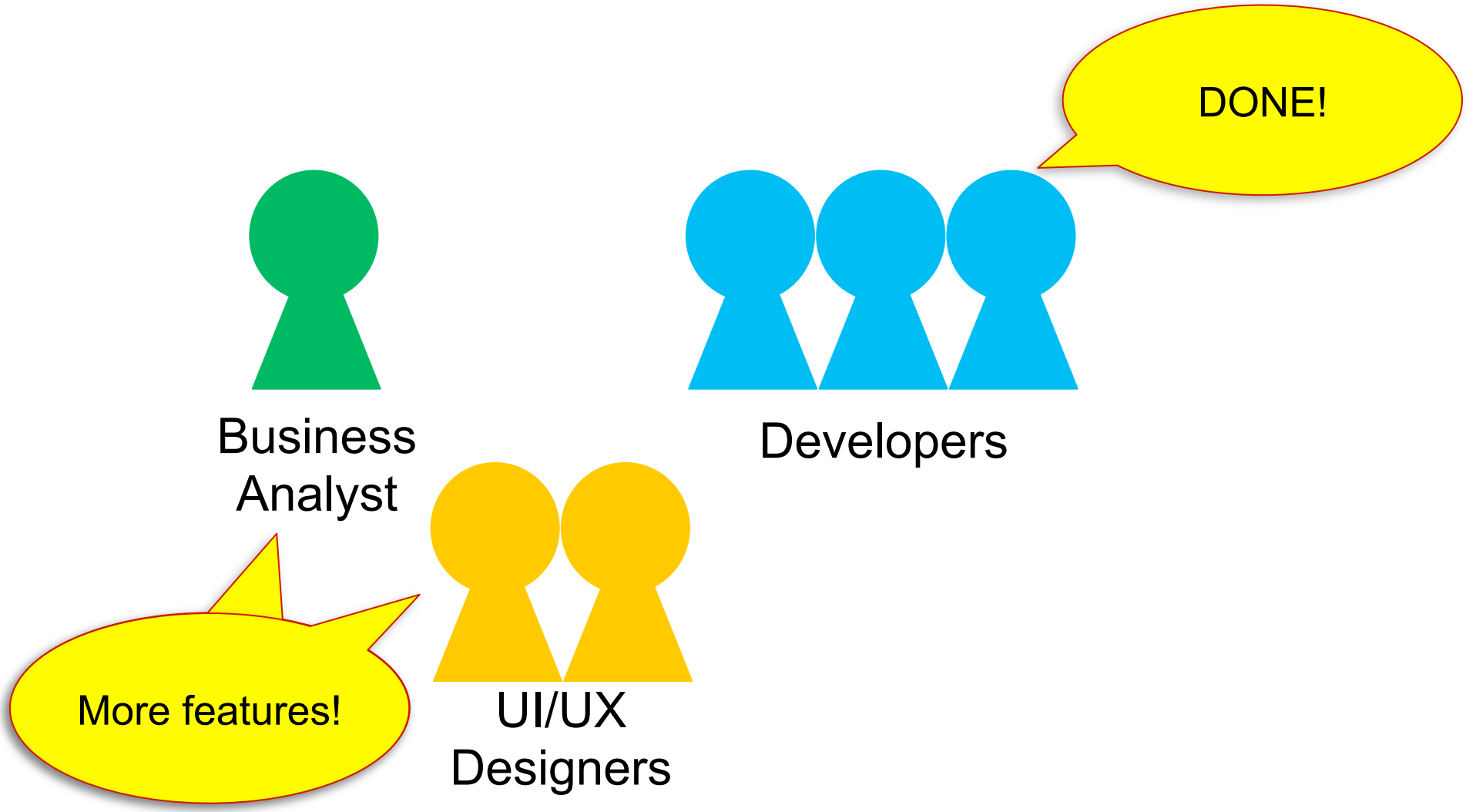
Learn domain knowledge effectively

楽天 Rakuten

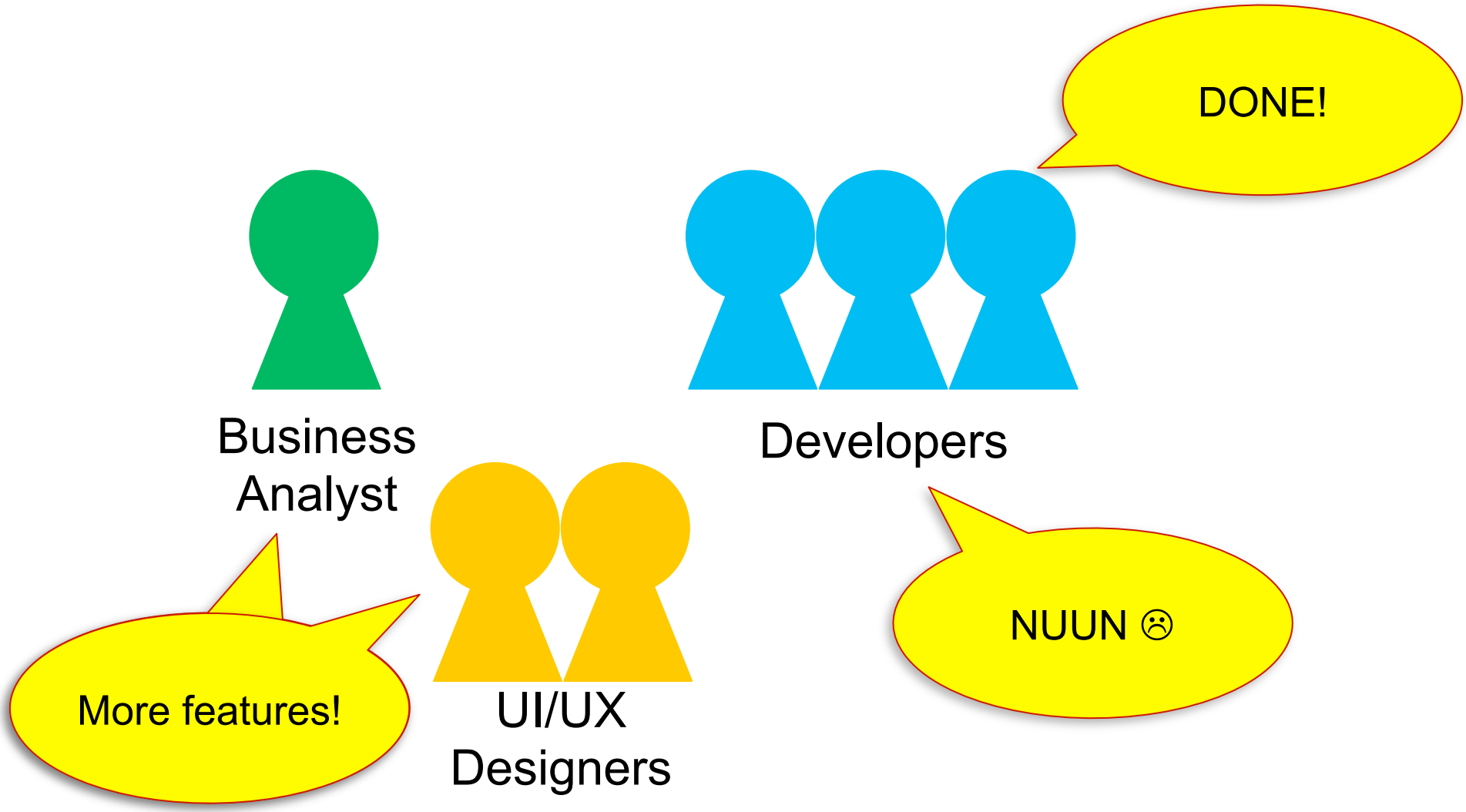
Example of feature creep



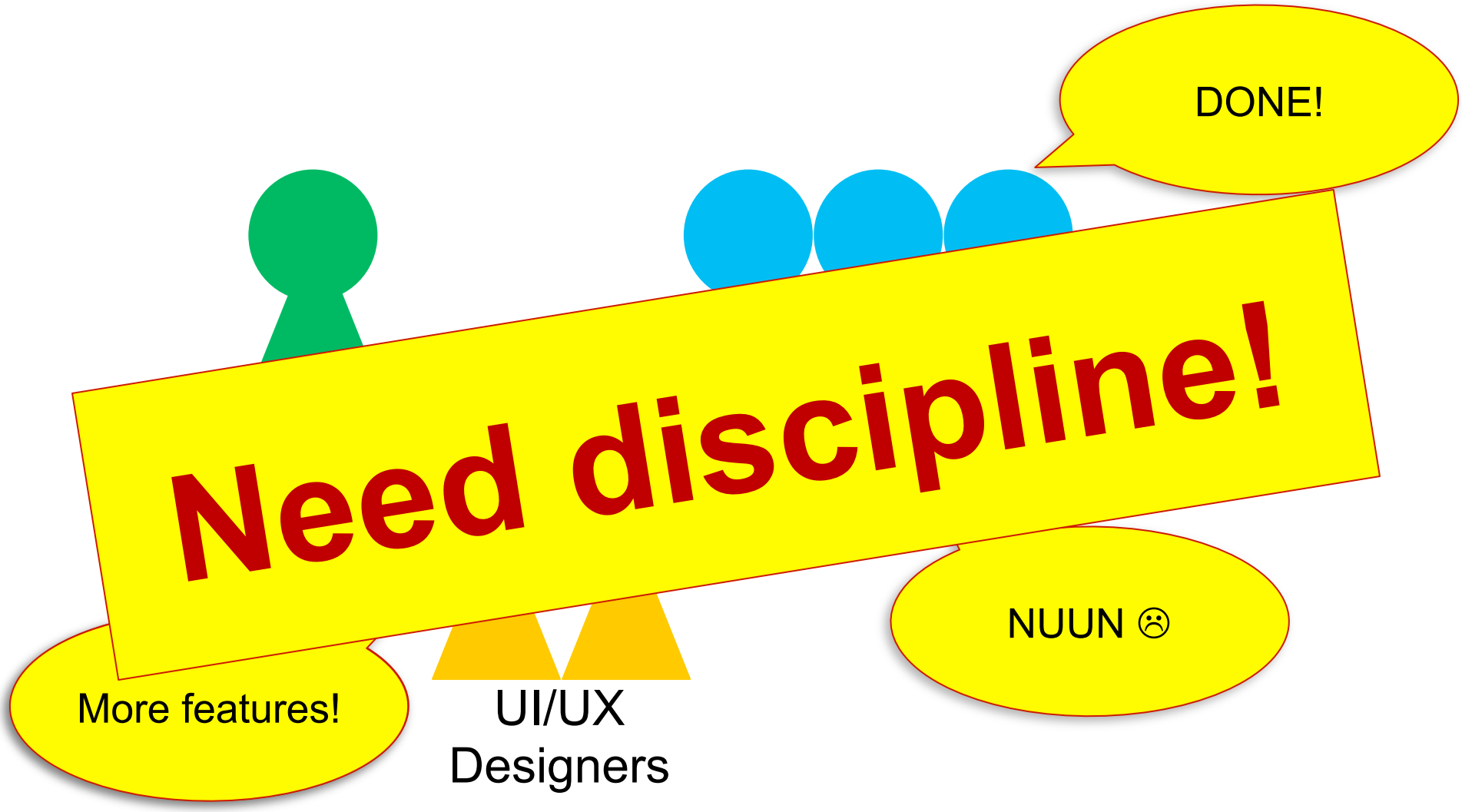
Example of feature creep



Example of feature creep



Example of feature creep



Calabash-android: improve the discipline

- The wrapper of Cucumber for Android
- As an executable specification
- As a communication tool

  Specifying collaboratively with
  business analyst, designers and developers
- By specification with examples

楽天 Rakuten

Example of BDD test scenario with Calabash-Android

Feature: Input
  Scenario: Input today's data

    Given I kick drumroll
    And drumroll show today
    When press next
    Then I should see "xxx" screen

    When I press keys and calculator should show like this:

2	2
0	20
0	200
*	200
3	3
=	600

    Then take photo

…

- Feature : name of all cases
- Scenario : name of each case

These statements are RUNNABLE!

We can write data with table style like this

Process of BDD

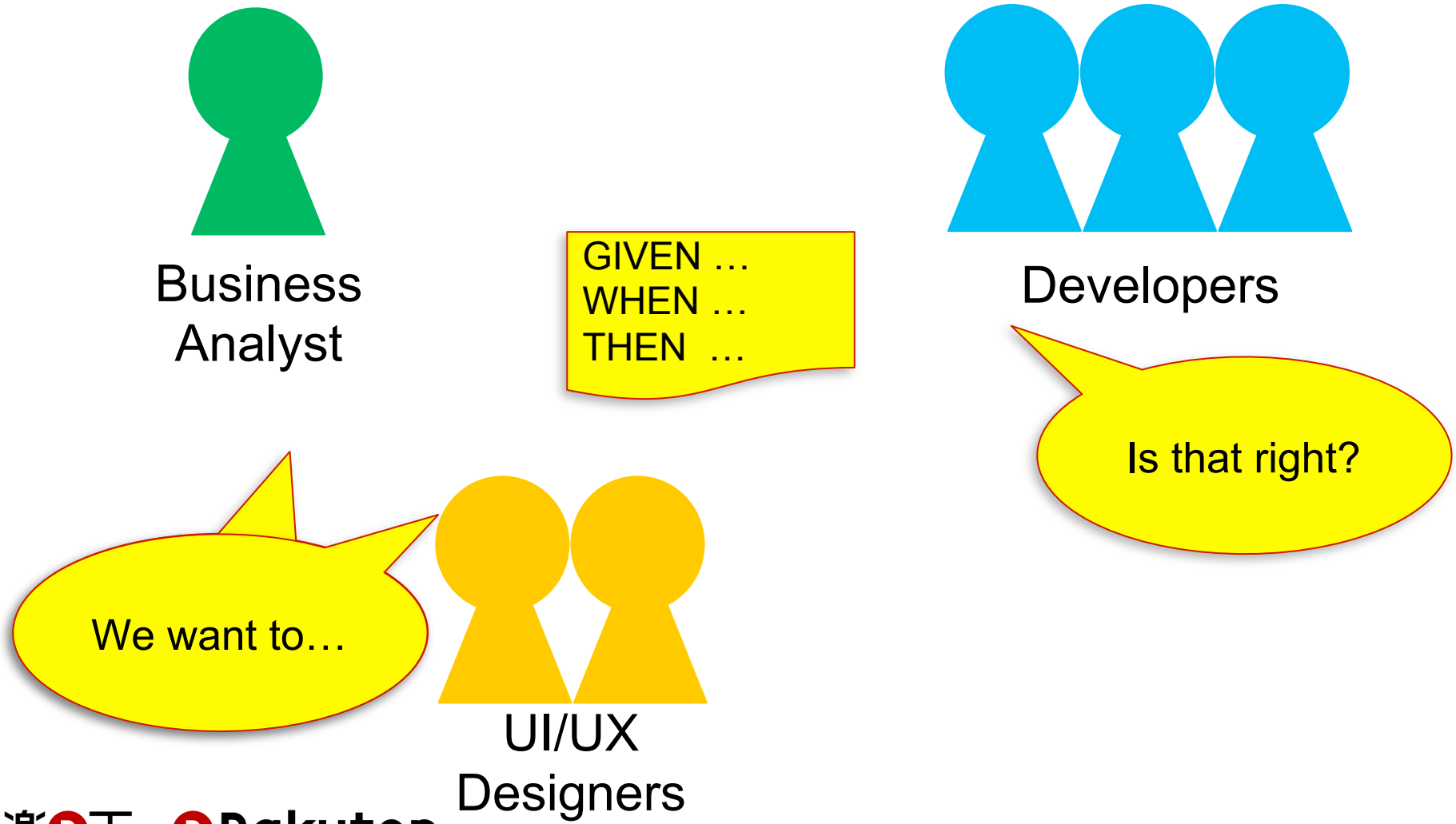
Business Analyst

Developers

UI/UX Designers

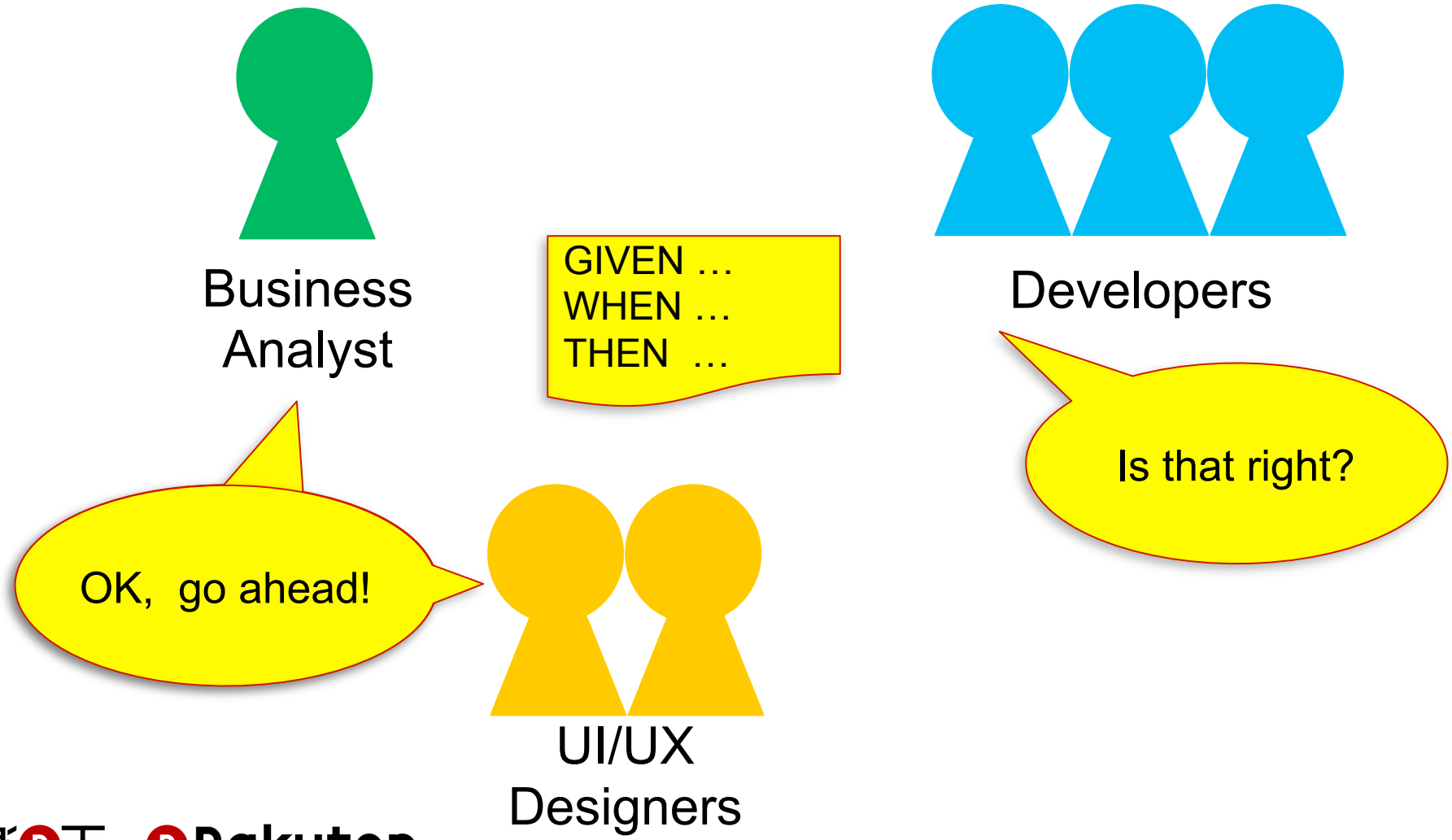
楽天 Rakuten

Process of BDD



楽天 Rakuten

Process of BDD



楽天 Rakuten

Process of BDD

Business Analyst

GIVEN …
WHEN …
THEN …

Developers

Is it OK?

UI/UX Designers

Process of BDD

Business Analyst

GIVEN …
WHEN …
THEN …

Developers

Sure!

UI/UX Designers

Is it OK?

After BDD

- Bugs : **-67%**
- Change requests : **-70%**
- Regressions : **-60%**

After BDD

- Bugs : **-67%**
- Chang

Improved!

1. Conditions and Challenges

2. CI/CD

3. TDD

4. BDD

5. Results, Problems, Possibility and Future

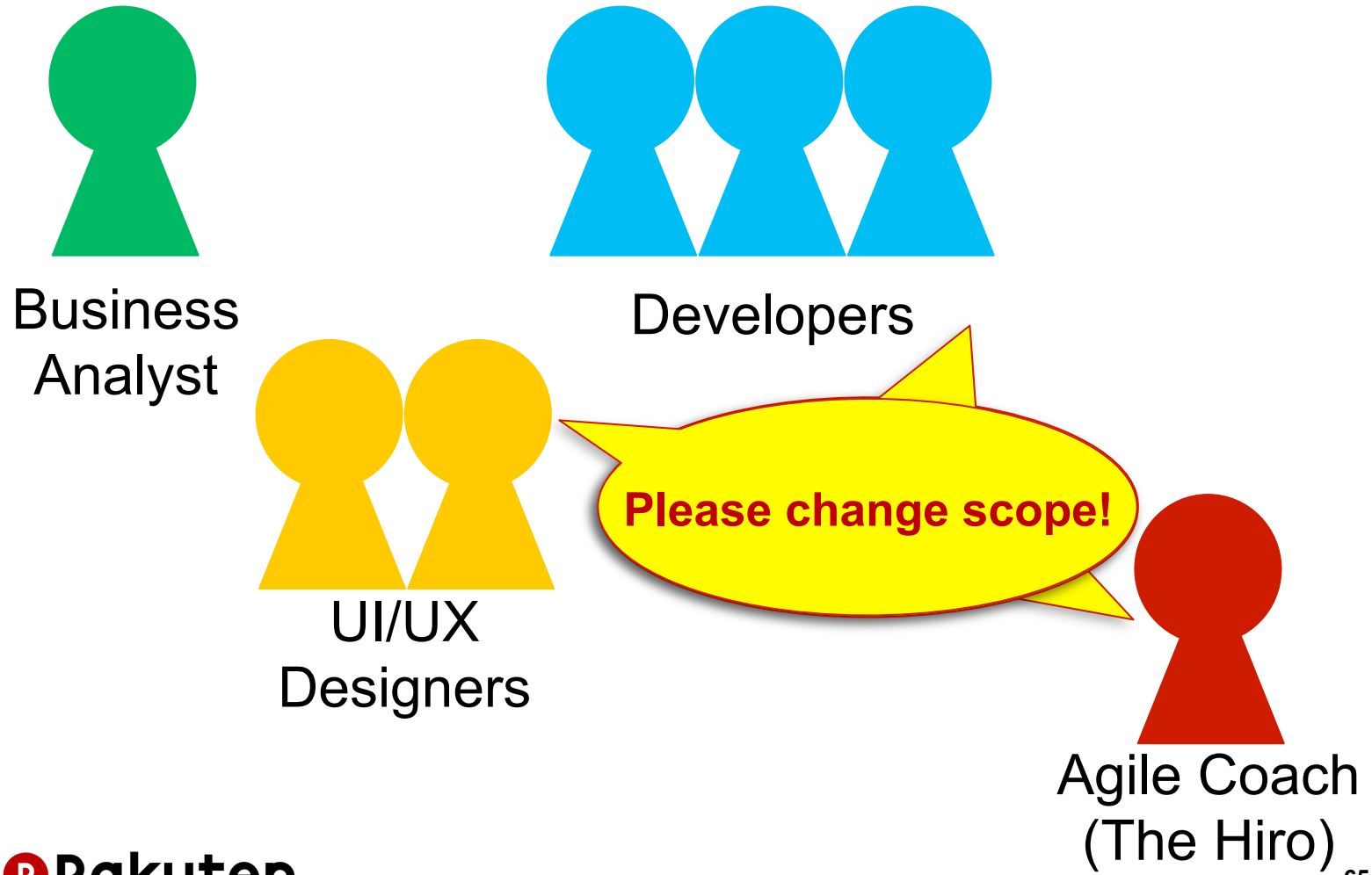
6. Conclusions

Results

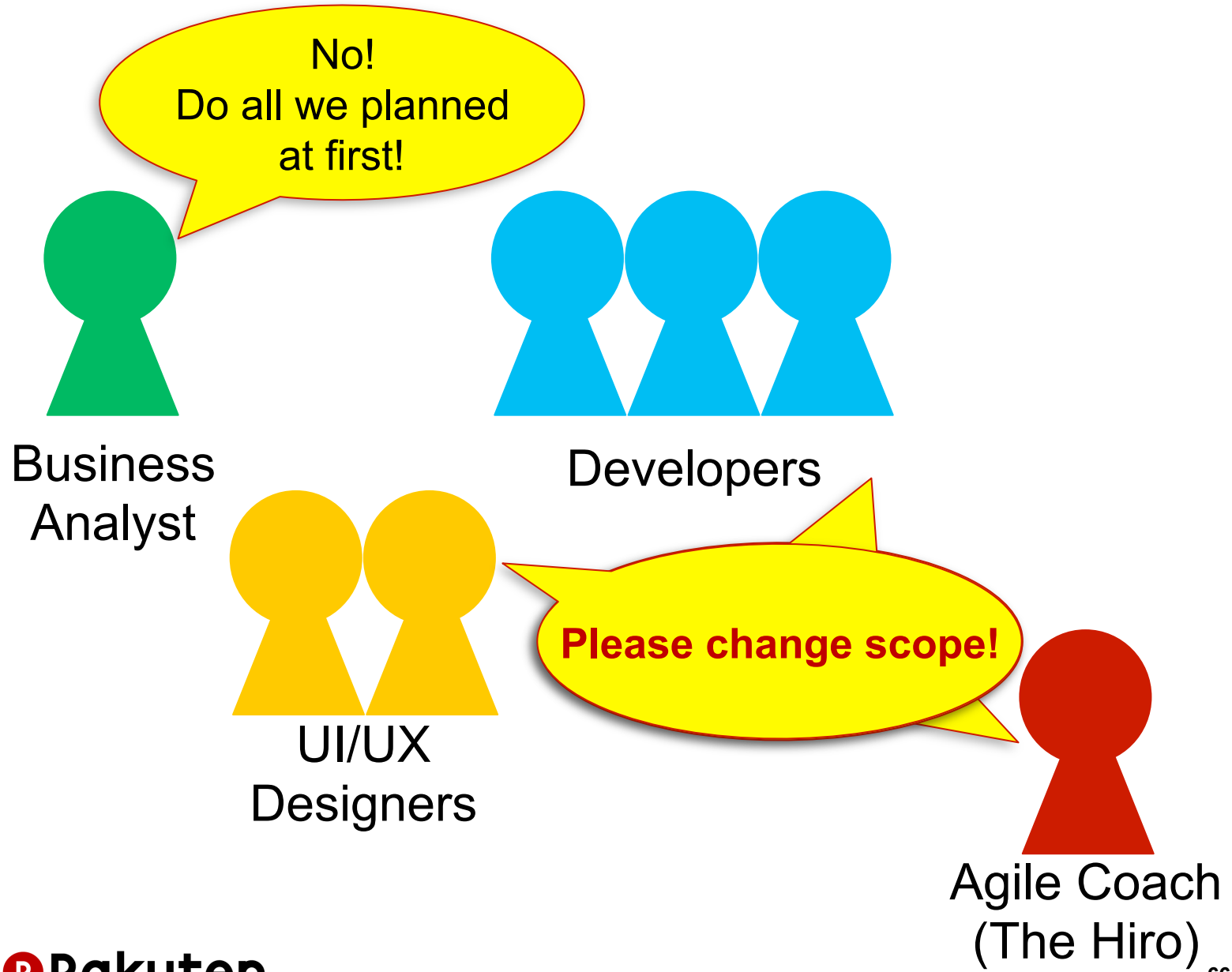
Challenges

楽R天 RRakuten

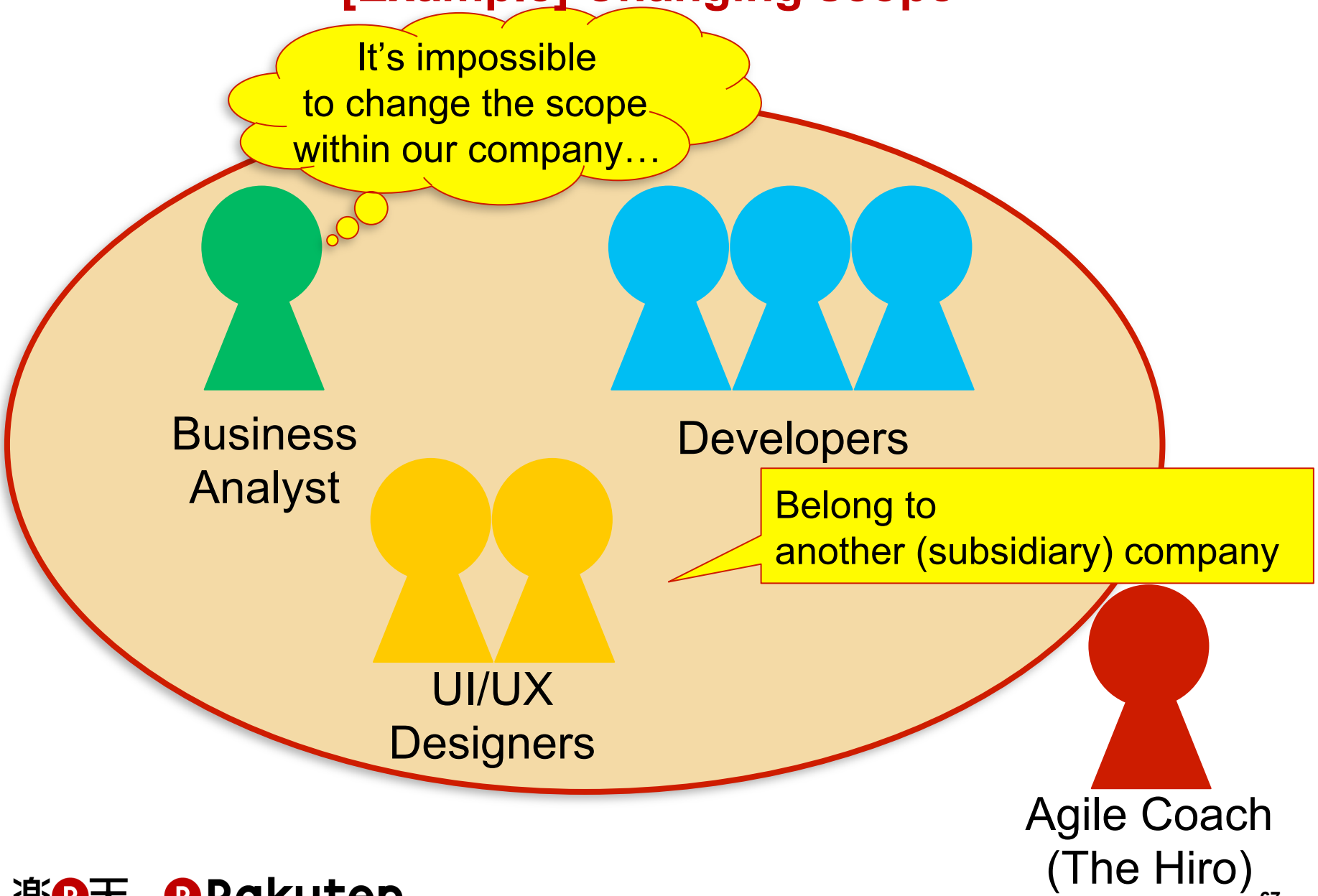
[Example] Changing scope



[Example] Changing scope



[Example] Changing scope



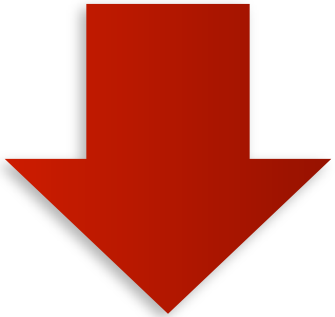
Asked for one executive

YES, YOU CAN!

We changed scope!

Point

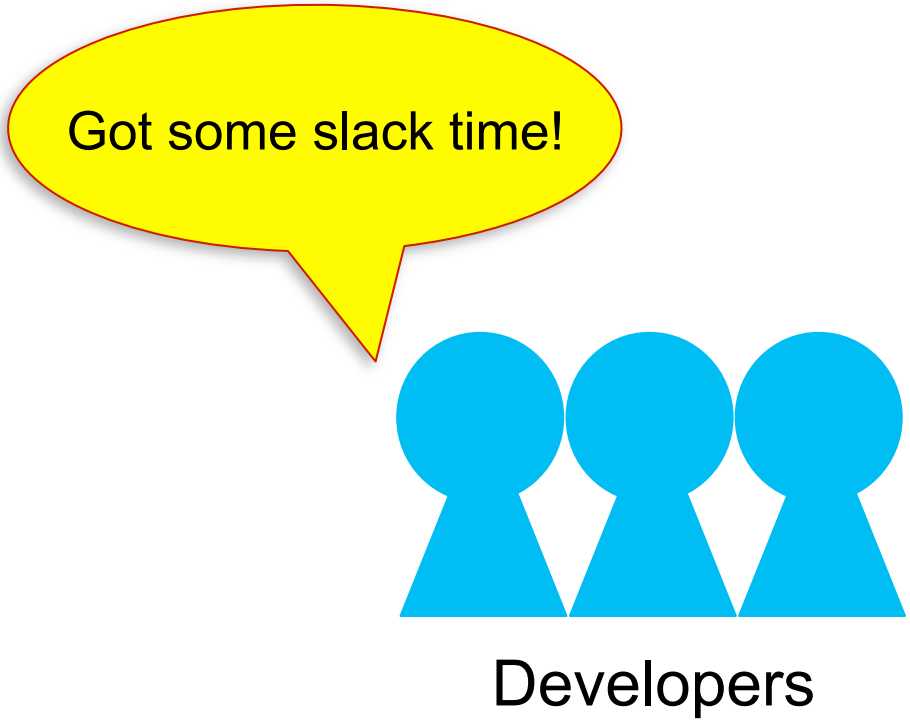
Technical excellence and working software are **not the only way** to improve project.



Anything is OK for improving your situlation!
(Anything goes/Vale tudo)

Possibility and future

[Example] Growing a collaborative culture



[Example] Growing a collaborative culture

Got some slack time!

Developers

Too slow emulator…

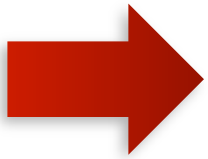
[Example] Growing a collaborative culture

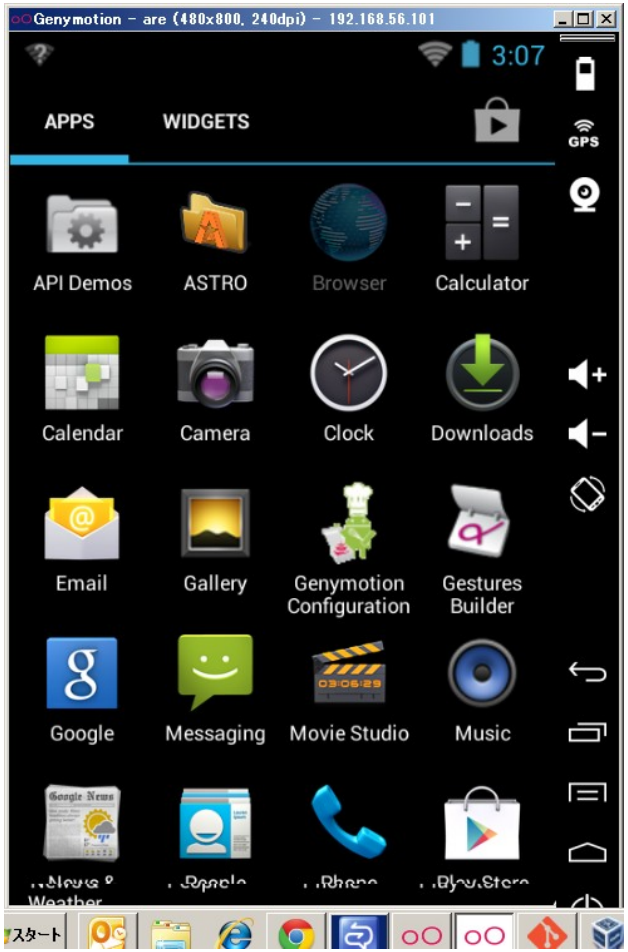
Got some slack time!

How about **Genymotion**?

Developers

Too slow emulator…





楽天 Rakuten

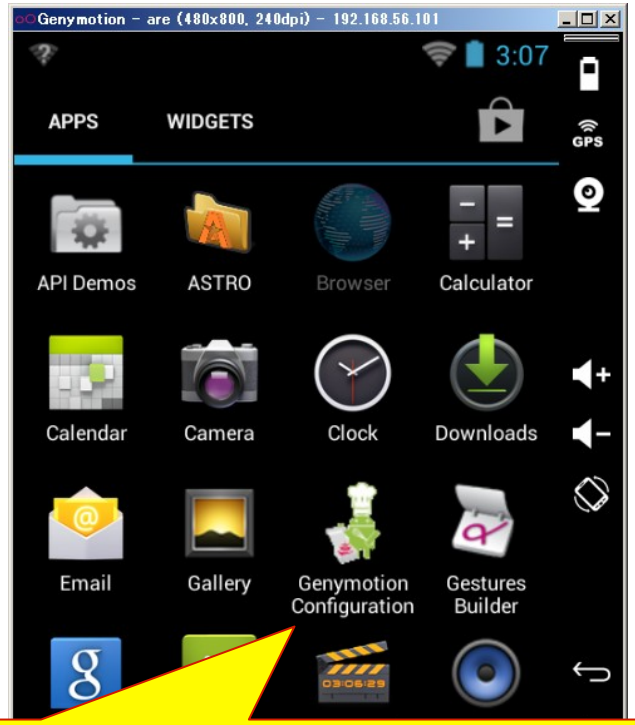
[Example] Growing a collaborative culture

Got some slack time!

How about **Genymotion**?

Developers

Too slow emulator…



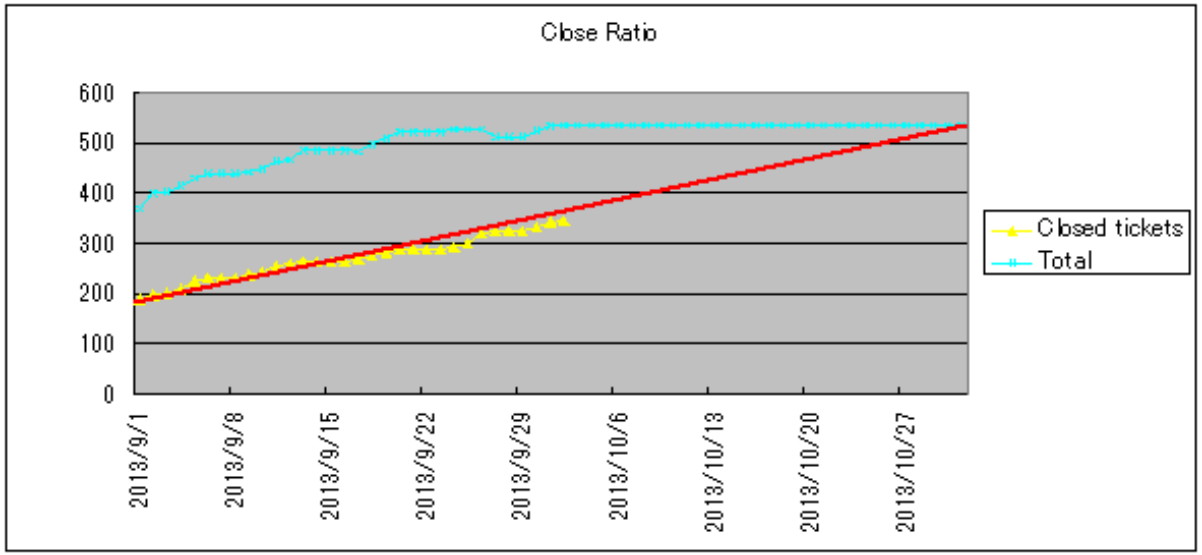
- **Over 10 times faster**
- Can run via Calabash-Android

楽天 Rakuten

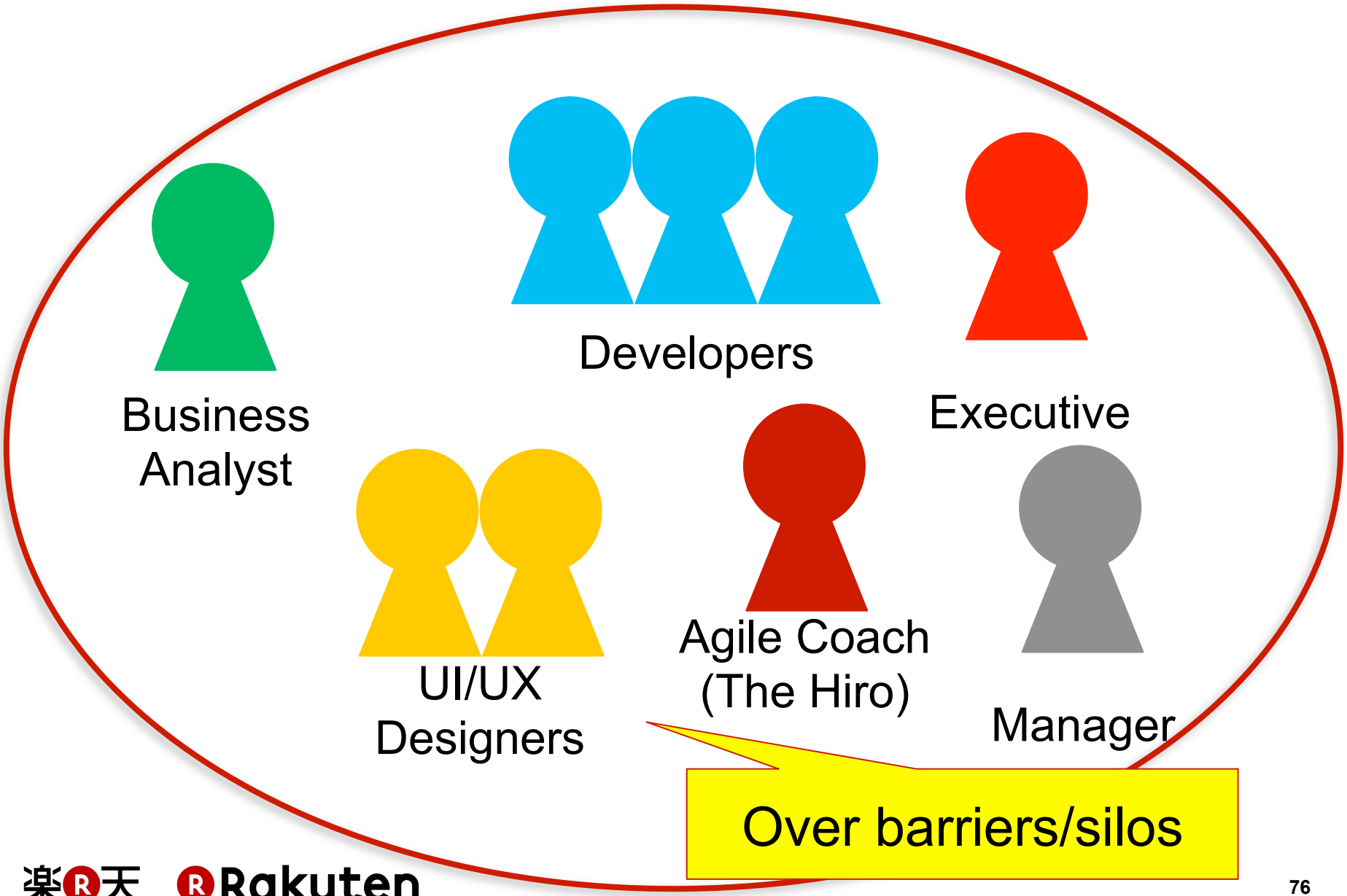
Can enhance “TDD” by numerical measurement

[e.g.]

- Change requests : 3 times/week
- Regression testing : **3 minutes**/change
- Install applications : **2 minutes**/change



Use “TDD” as a measure for total optimization



楽天 Rakuten

Don’t lose the whole picture!

楽天 Rakuten

1. Conditions and Challenges

2. CI/CD

3. TDD

4. BDD

5. Results, Problems, Possibility and Future

6. Conclusions

楽天 Rakuten

Three purposes

Efficiency

Learning

Collaboration

Three approaches

CI/CD

TDD

BDD

Three approaches by

CI/CD

TestFlight

iOS Beta Testing On The Fly

TDD

mockito

BDD

Cucumber

楽天 Rakuten

We found this practice

- through the project
- with passionate members
- with a lot of trial and error

Experience from Gemba
現場主義

Find your **answer**
by **yourself**
through your **experience**

Find your treasure!

楽天 Rakuten