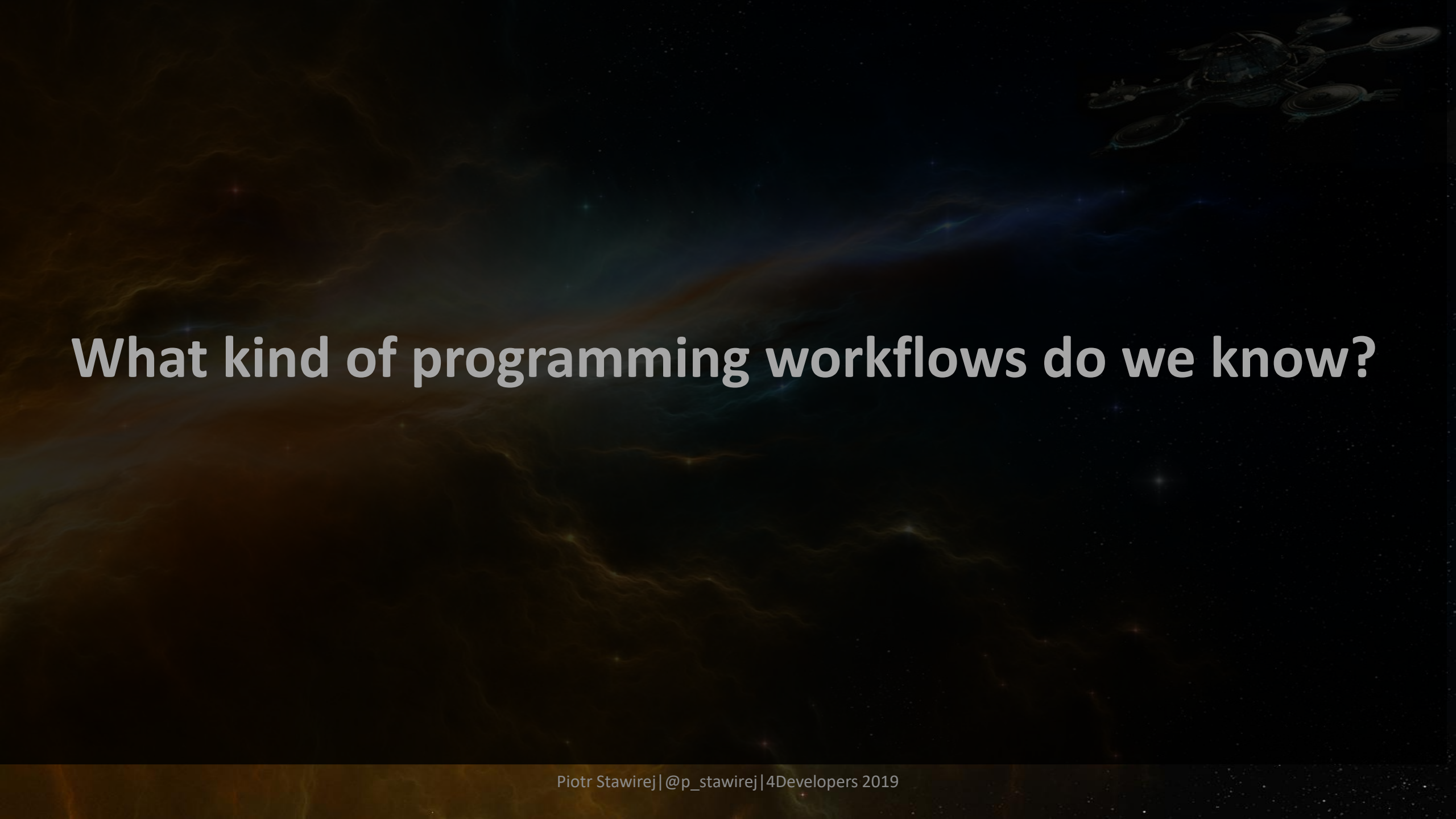




test && commit || revert

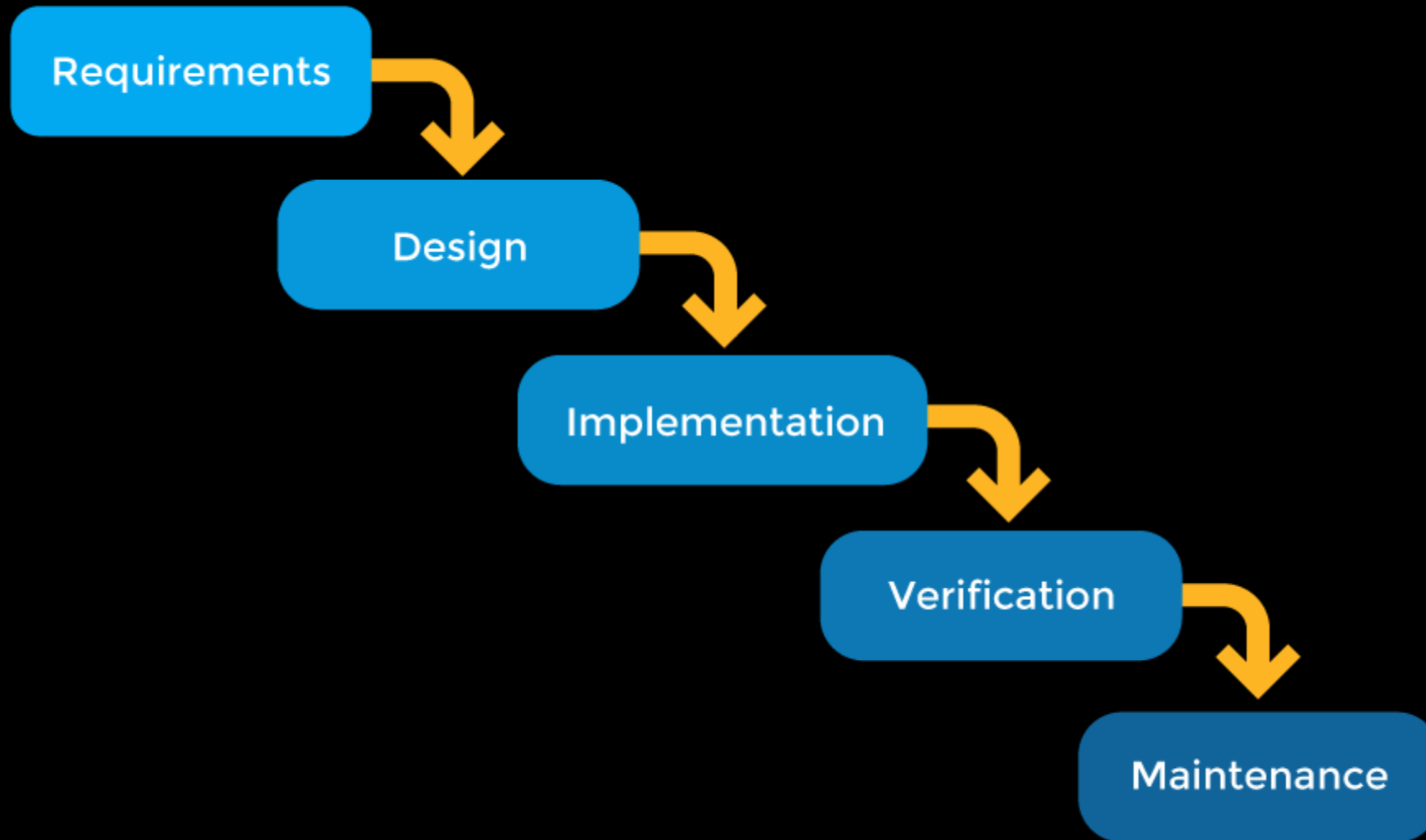
(TCR)

alternative to TDD or eccentricity?

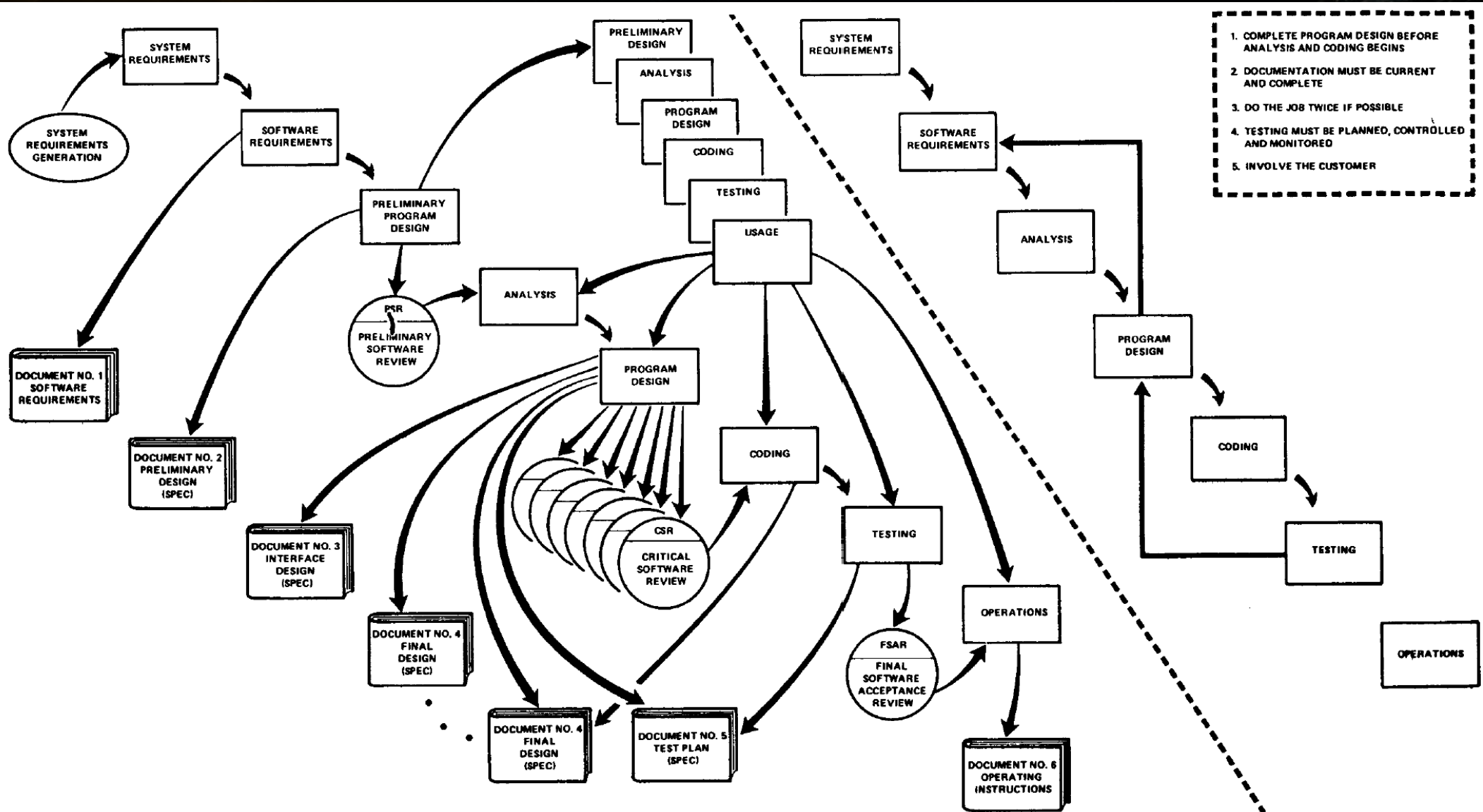
A futuristic drone with multiple rotors is flying in the upper right corner of the frame. The background is a dark, atmospheric scene with swirling orange and blue energy fields or smoke, creating a sense of a high-tech or sci-fi environment.

What kind of programming workflows do we know?

Waterfall

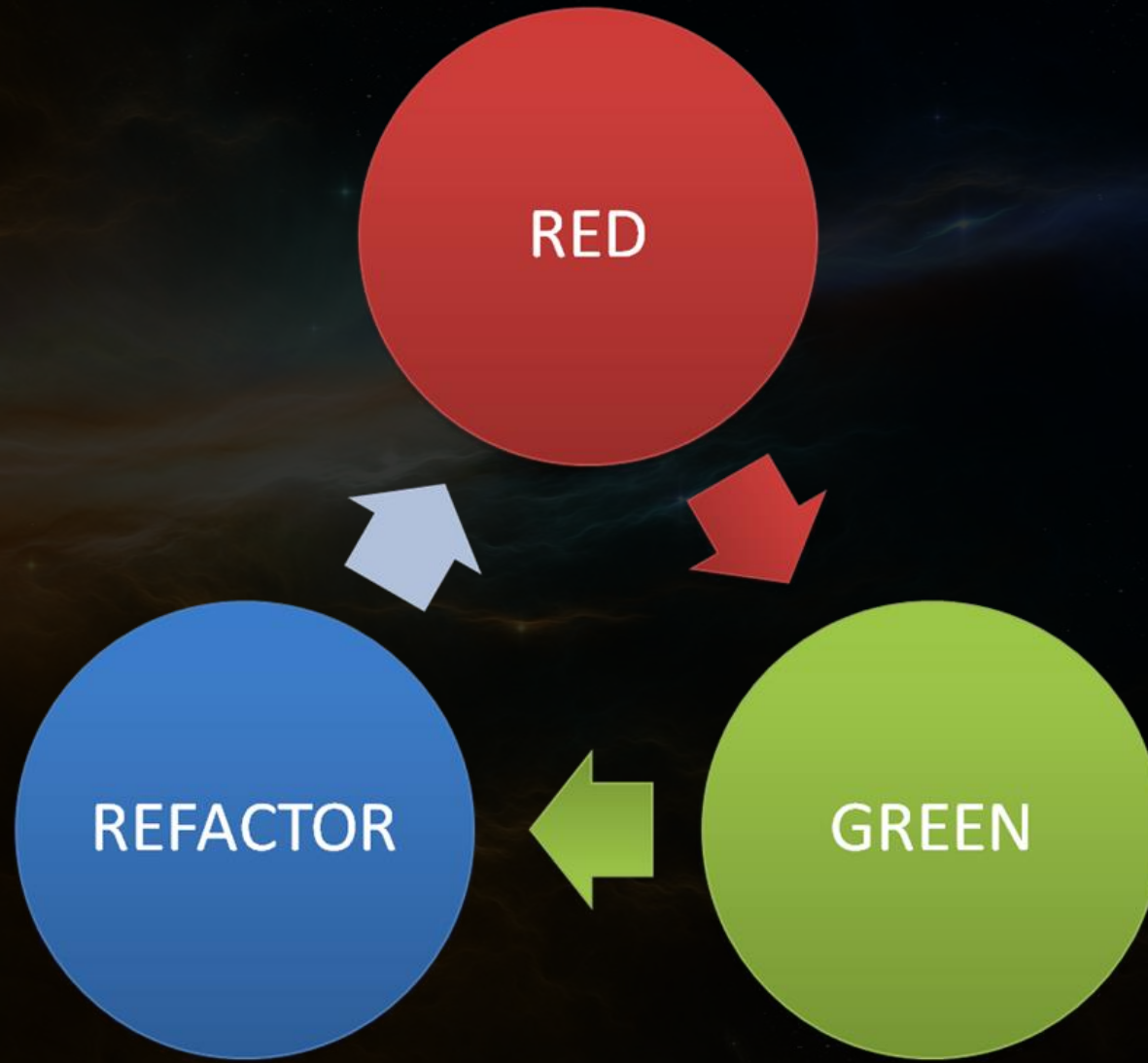


Waterfall



1970 Dr. Winston W. Royce, "Managing the development of large software systems".

TDD/BDD



A futuristic drone with four rotors and a central body is flying in the upper right corner of the image. The background is a dark, starry space with a large, glowing nebula in shades of orange and yellow on the left side. The text "Test after code" is centered in the middle of the image.

Test **after** code

Protoduction



Nightmare

Make a change
It doesn't work
Make a change
It doesn't work
Make a change
It doesn't work

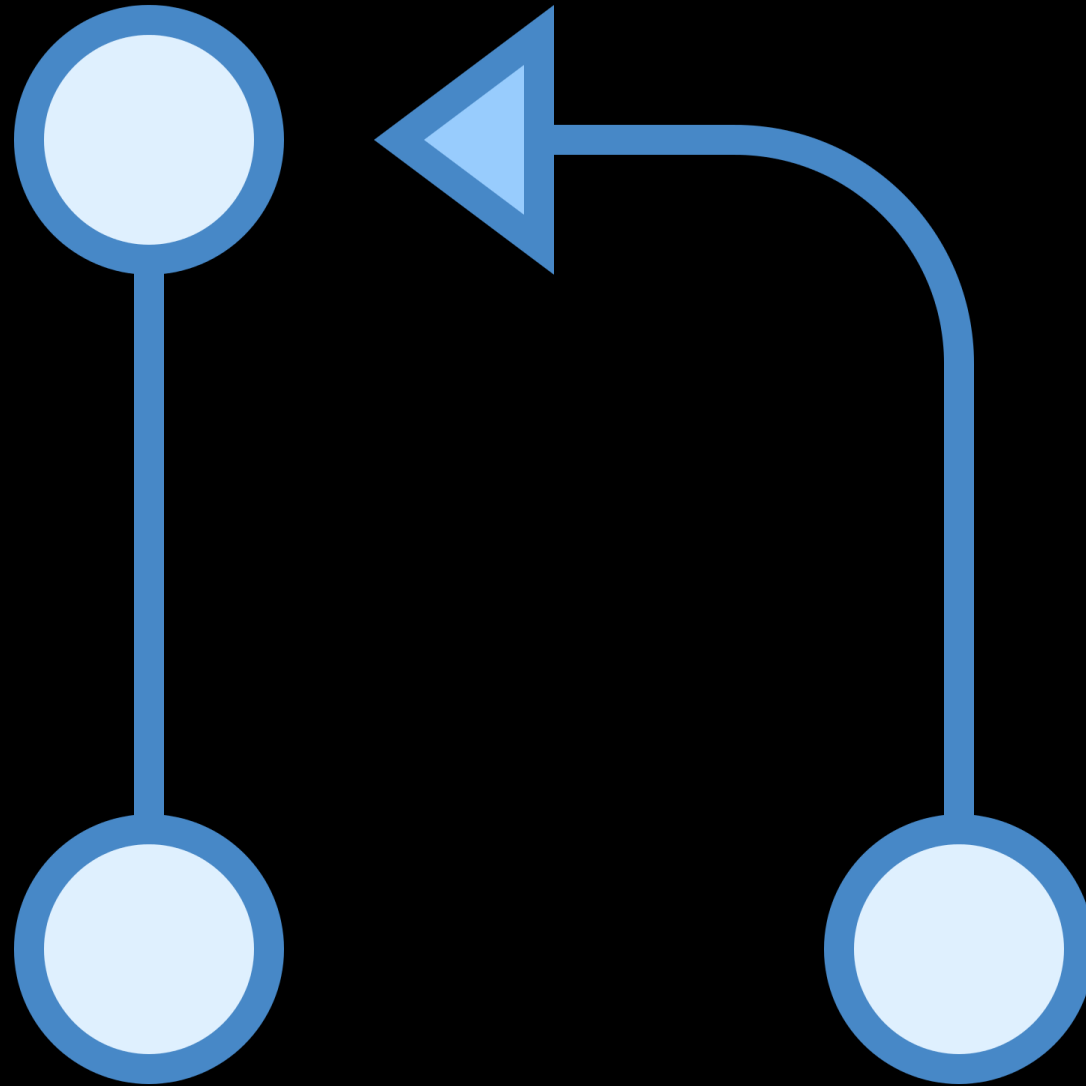




**NO
FEAR**

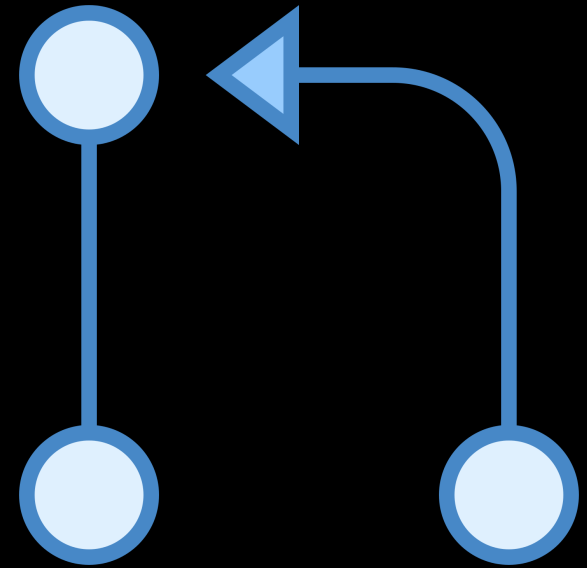
Any problem - revert to last known green state
Cheap to get to known state

Big Pull Request

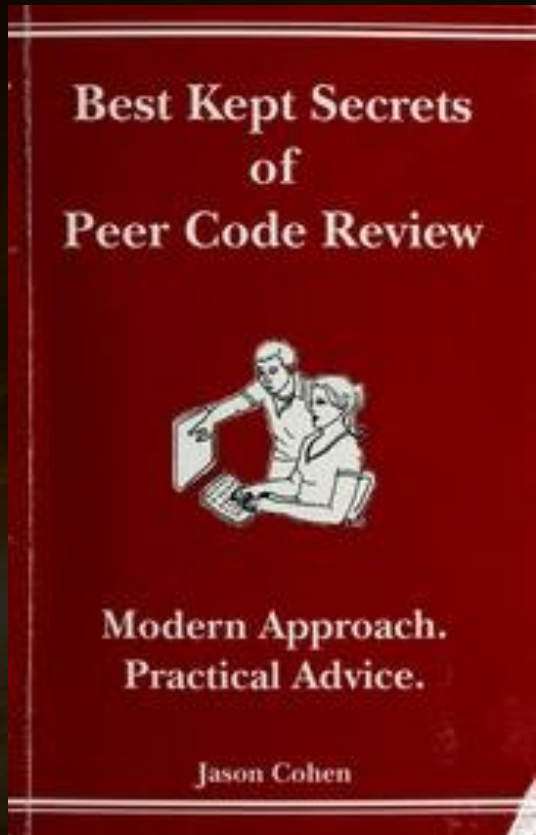


Big Pull Request

What does it mean small PR?



Big Pull Request

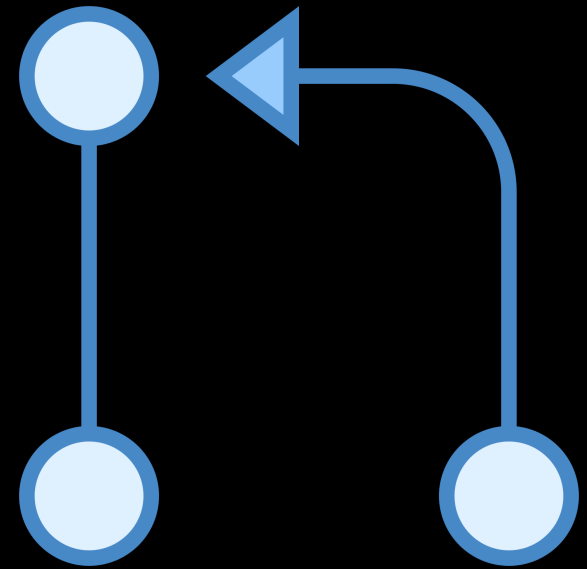


What does it mean small PR?

~400 LOC/h

Brain saturation after 1 hour

200-400 LOC



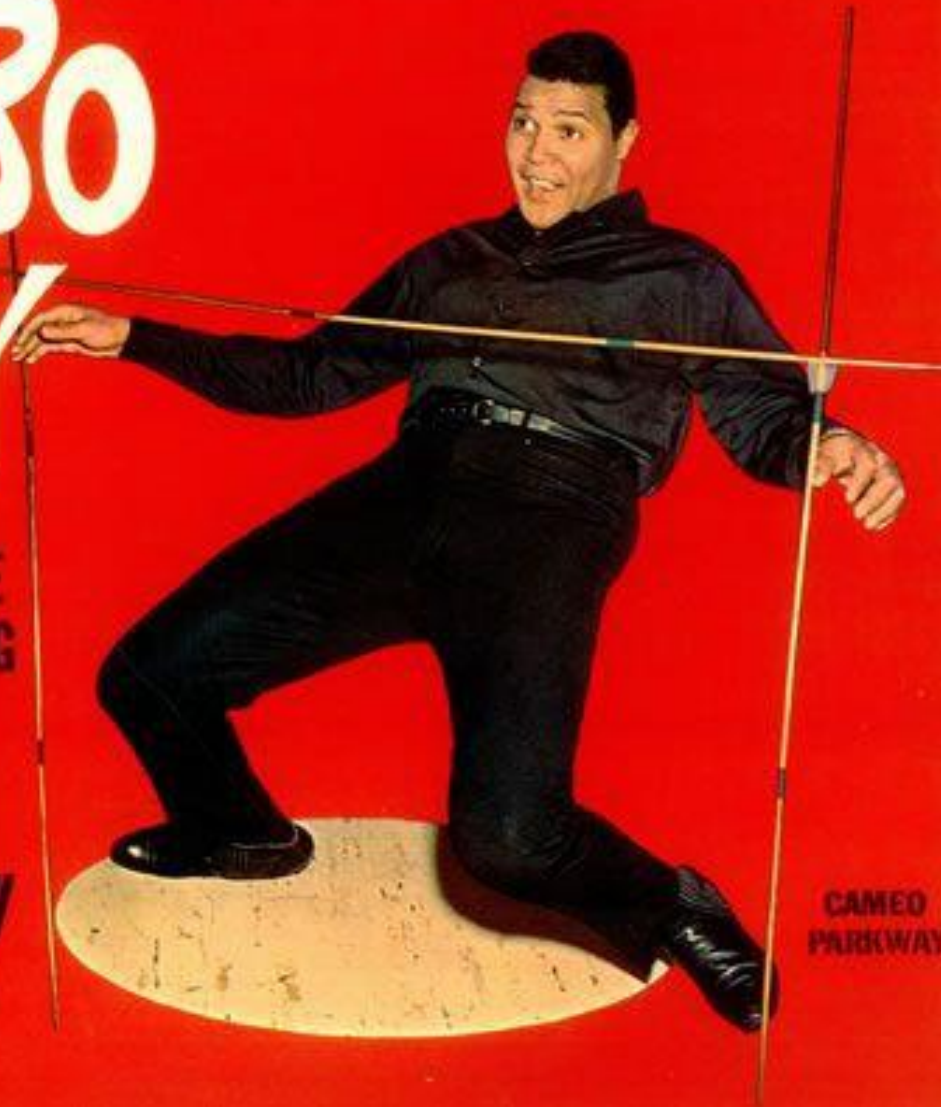
CHUBBY CHECKER

LIMBO

PARTY

THE DANCE CRAZE
THAT'S SWEEPING
THE COUNTRY!

NOW...
HAVE YOUR OWN
LIMBO PARTY!

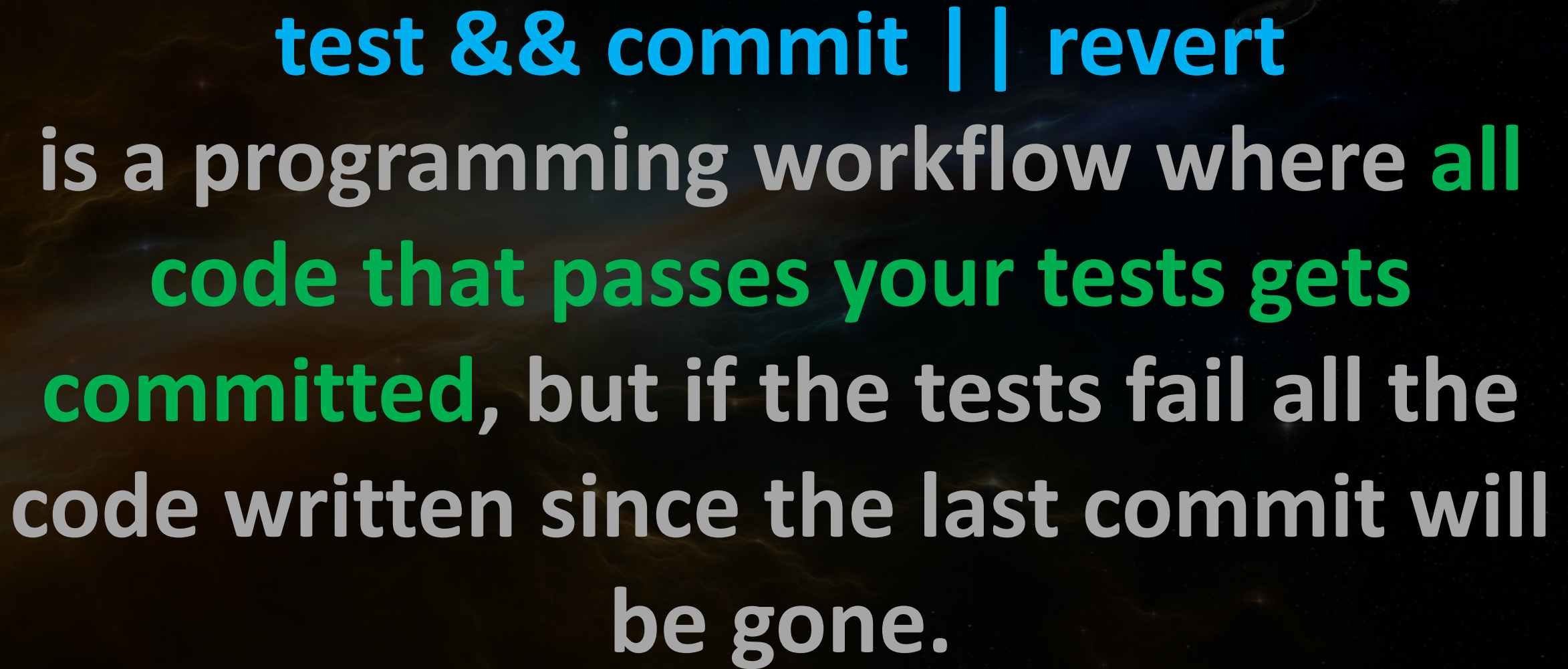


CAMEO
PARKWAY

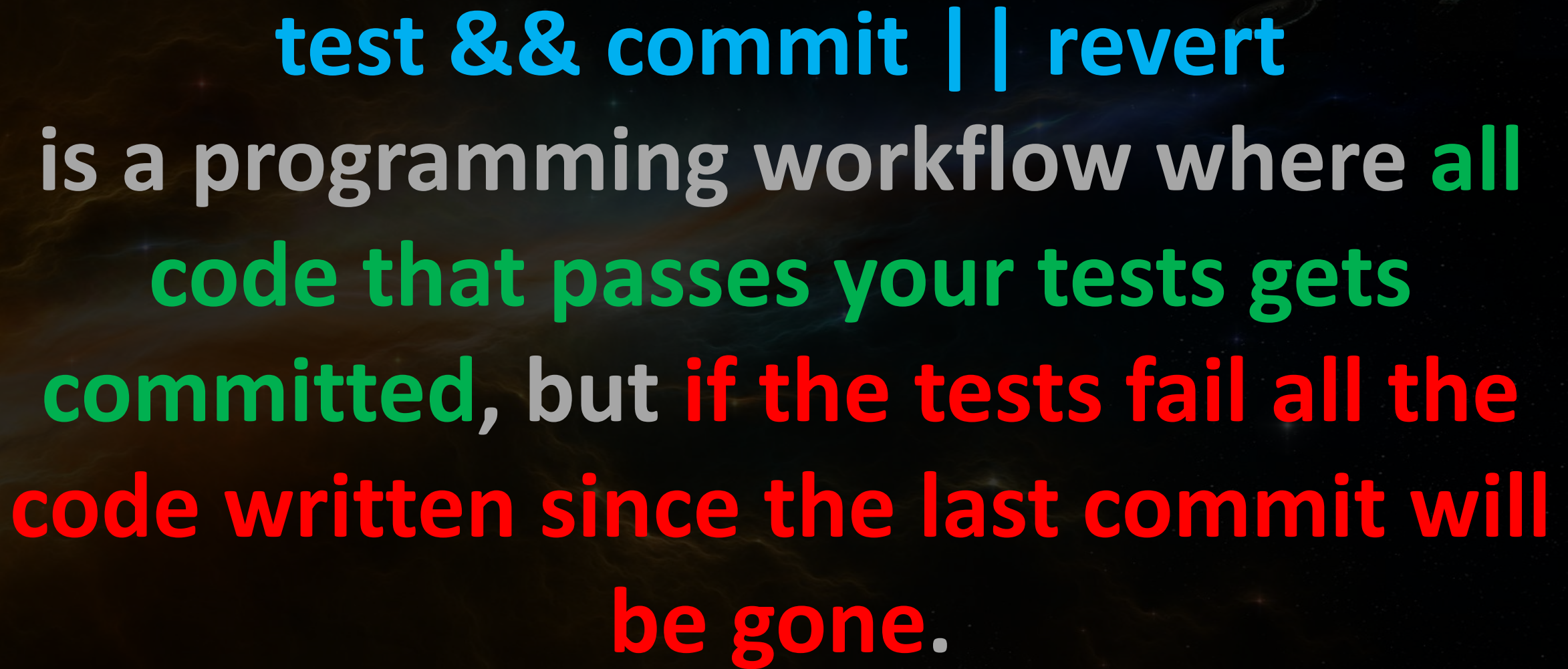


test && commit || revert

is a programming workflow where all code that passes your tests gets committed, but if the tests fail all the code written since the last commit will be gone.



test && commit || revert
is a programming workflow where **all**
code that passes your tests gets
committed, but if the tests fail all the
code written since the last commit will
be gone.

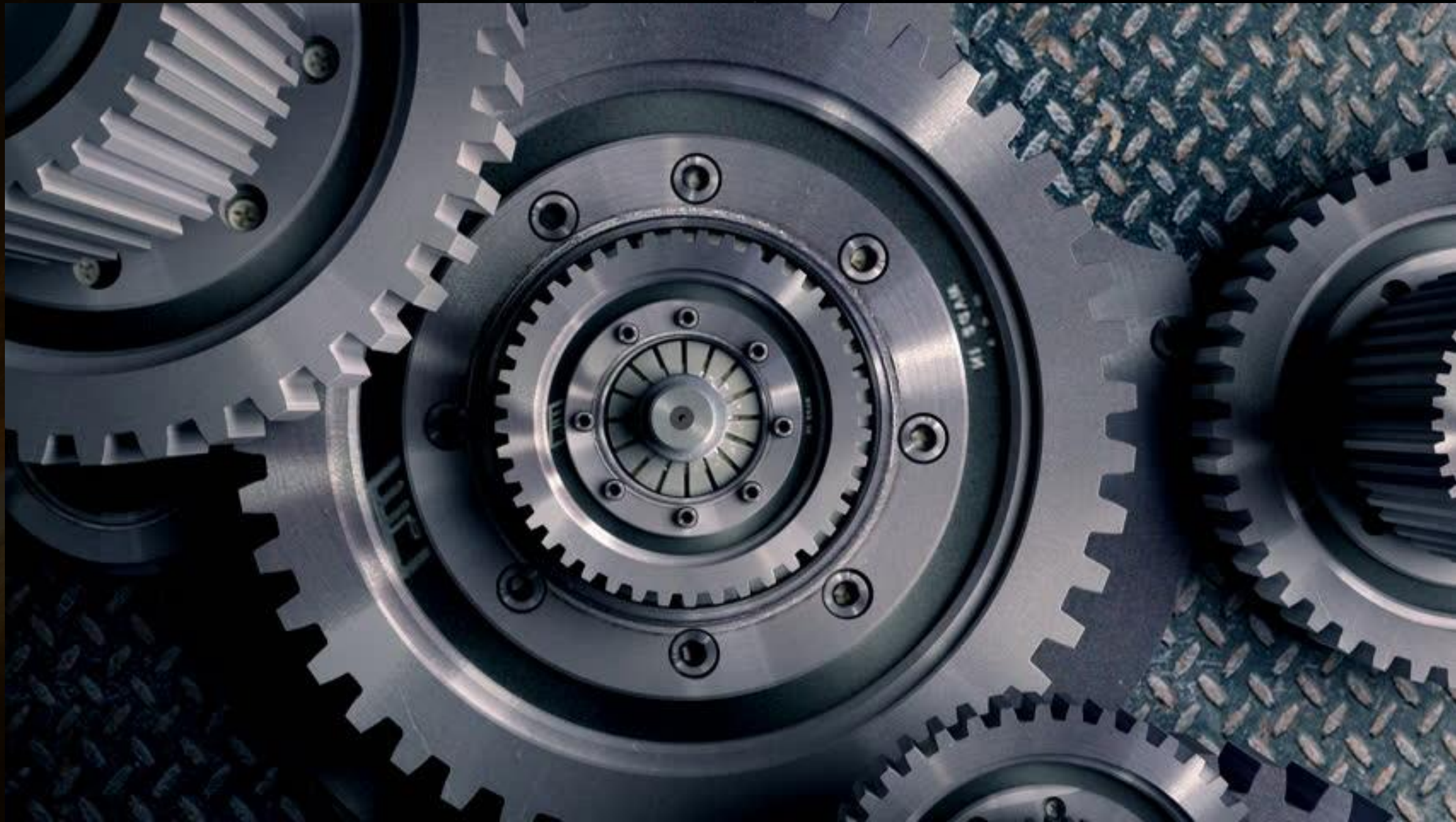


test && commit || revert
is a programming workflow where **all**
code that passes your tests gets
committed, but **if the tests fail all the**
code written since the last commit will
be gone.

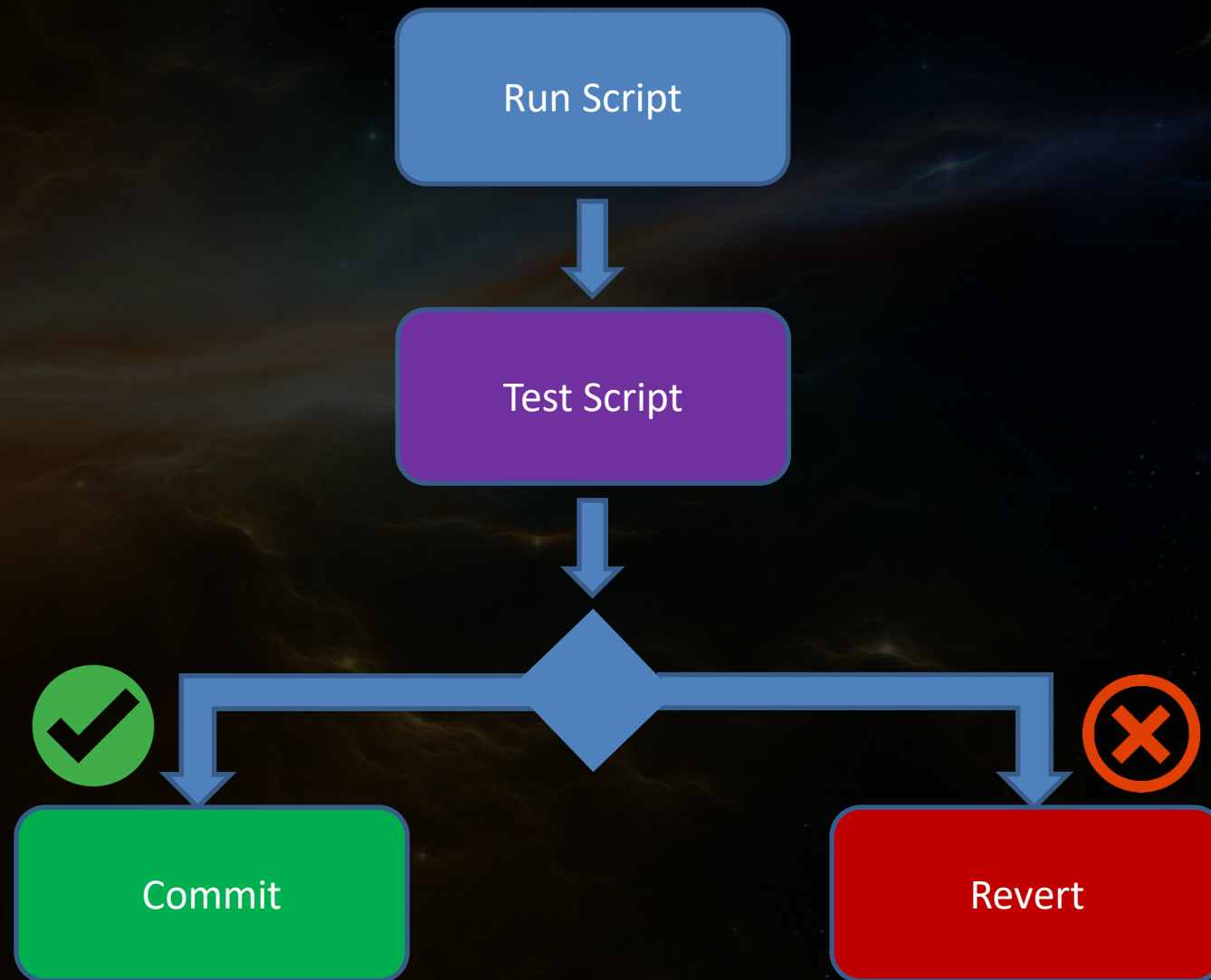




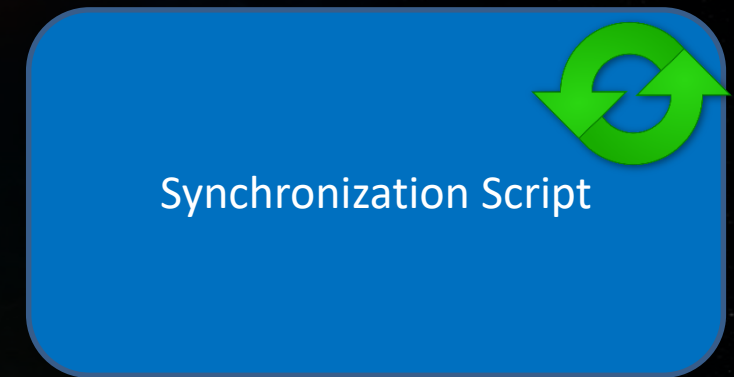
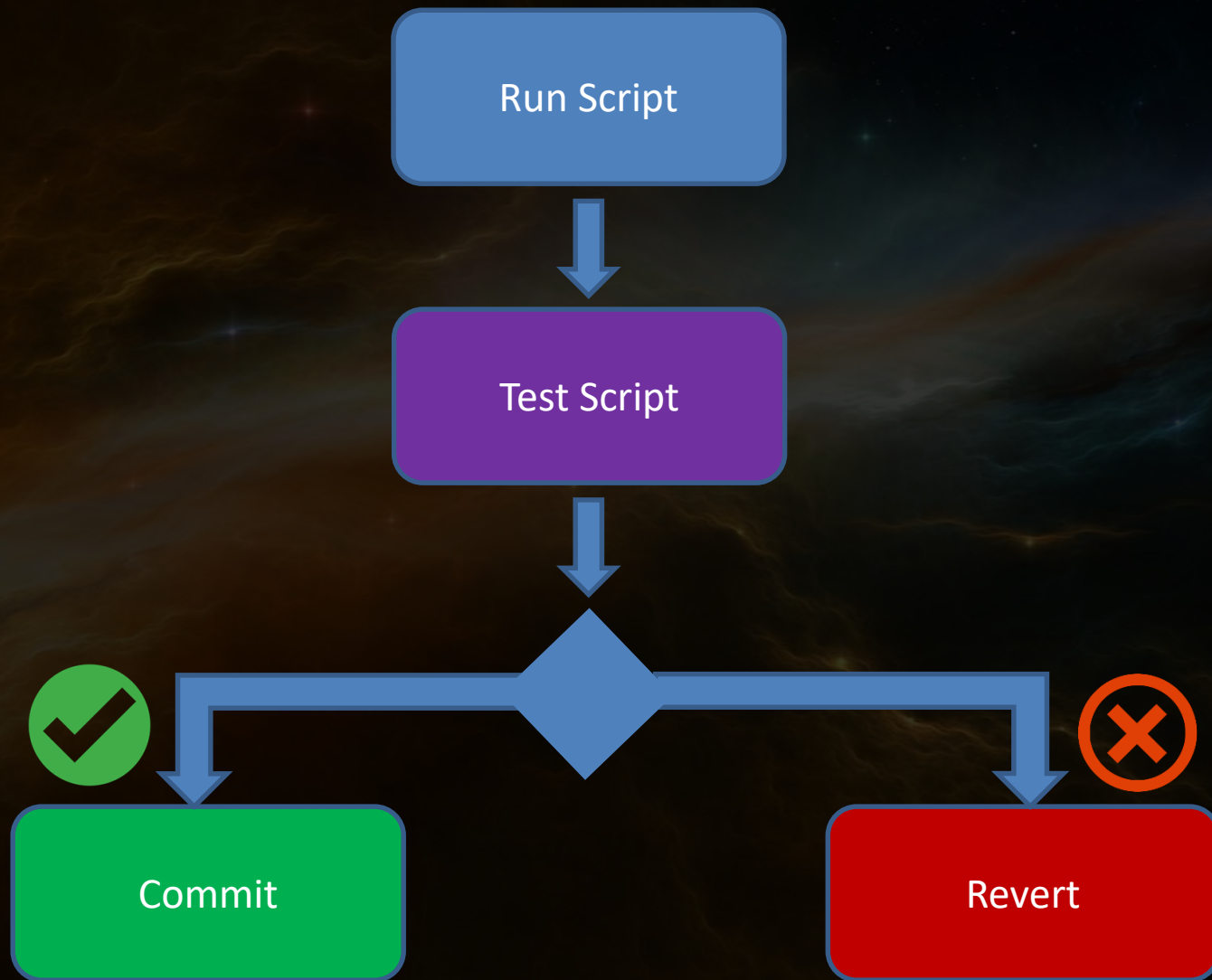
DEMO



Run script



Run script



Synchronization script



```
#!/usr/bin/env bash
```

```
while(true); do  
    git pull --rebase;  
    git push;  
done;
```


Test script



```
#!/usr/bin/env bash
```

```
mvn test
```

Run script



```
./test && git commit -am working || git reset --hard
```


Run script revisited



```
#!/usr/bin/env bash
```

```
test() {
```

```
}
```

```
commit() {
```

```
}
```

```
revert() {
```

```
}
```

```
test && commit || revert
```

Run script



```
#!/usr/bin/env bash
```

```
test() {  
    ./test.sh  
}
```

```
commit() {  
}
```

```
revert() {  
}
```

```
test && commit || revert
```


Run script



```
#!/usr/bin/env bash
```

```
test() {  
    ./test.sh  
}
```

```
commit() {  
    git commit -am working  
}
```

```
revert() {  
}
```

```
test && commit || revert
```

Run script



```
#!/usr/bin/env bash

test() {
    ./test.sh
}

commit() {
    if [[ $# -eq 0 ]]
    then git commit -am working
    else git commit -am "$1"
    fi
}

revert() {
}

test && commit "$@" || revert
```


Run script



```
#!/usr/bin/env bash

test() {
    ./test.sh
}

commit() {
    if [[ $# -eq 0 ]]
    then git commit -am working
    else git commit -am "$1"
    fi
}

revert() {
    git reset --hard
    git clean -fd
}

test && commit "$@" || revert
```

Logging?



```
#!/usr/bin/env bash

test() {
    ./test.sh | tee ./tcrfeedback 2>&1
}

commit() {
    # git add .
    if [[ $# -eq 0 ]]
    then git commit -am working
    else git commit -am "$1"
    fi
}

revert() {
    git reset --hard
    git clean -fd
}

test && commit "$@" || revert
```


Git

any
for a bad purp
abuse. verb
mistreat, m
treat badly
m

Explosion of git history

Copyright Tax



Explosion of git history

--amend

! [rejected] master -> master (non-fast-forward)

error: failed to push some refs to

...

hint: Updates were rejected because the tip of your current branch is behind

hint: its remote counterpart. Integrate the remote changes (e.g.

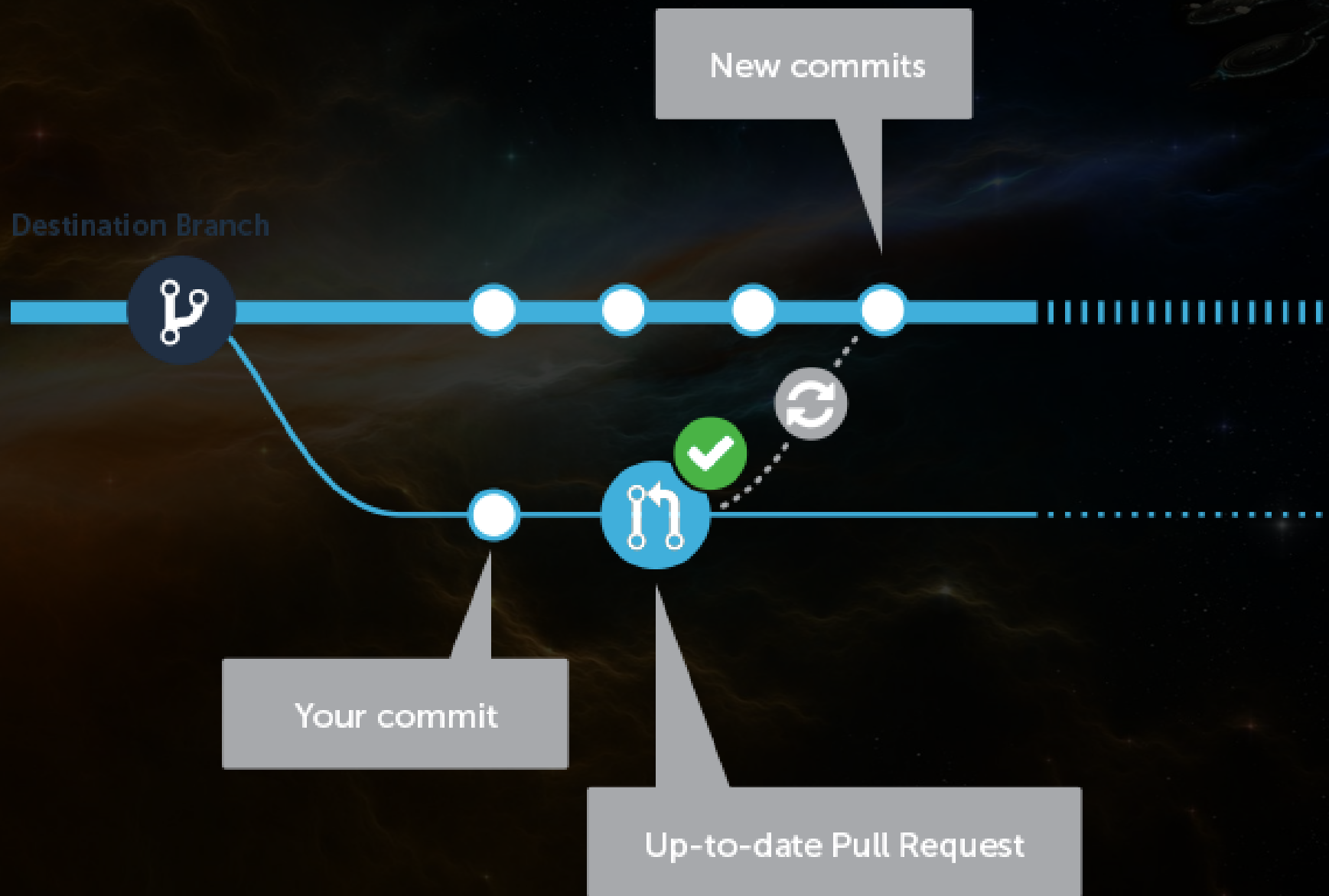
hint: 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

One to many

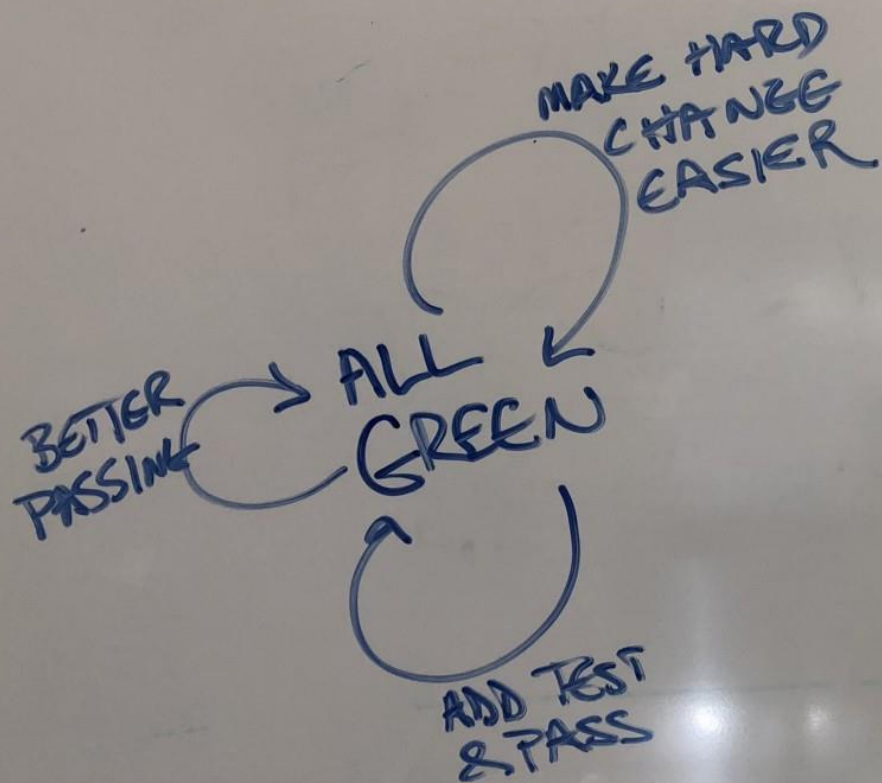


Where is the review part?

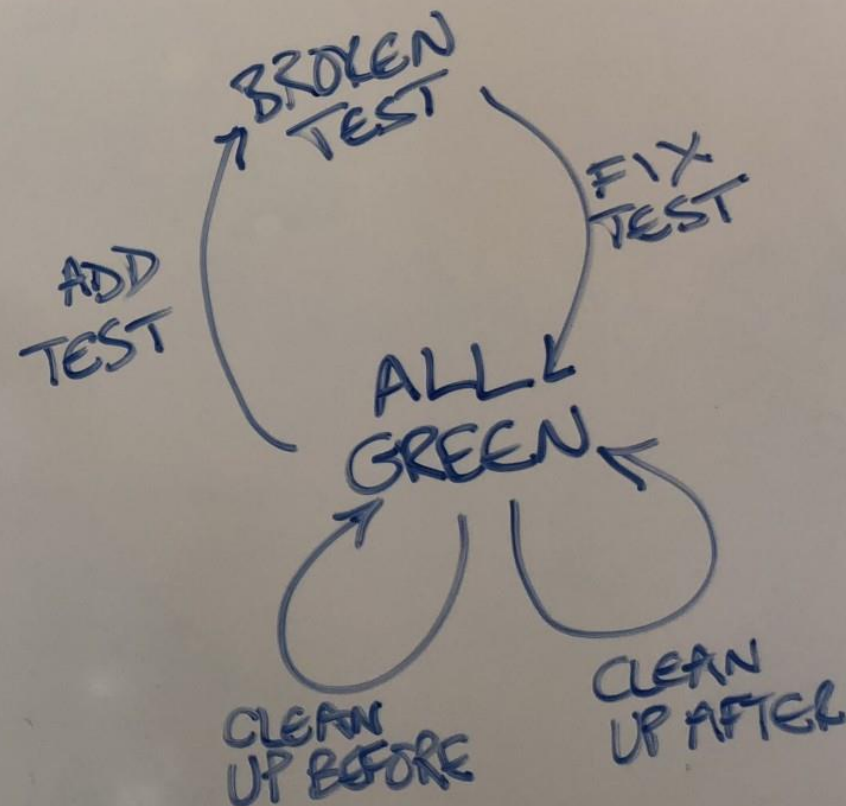


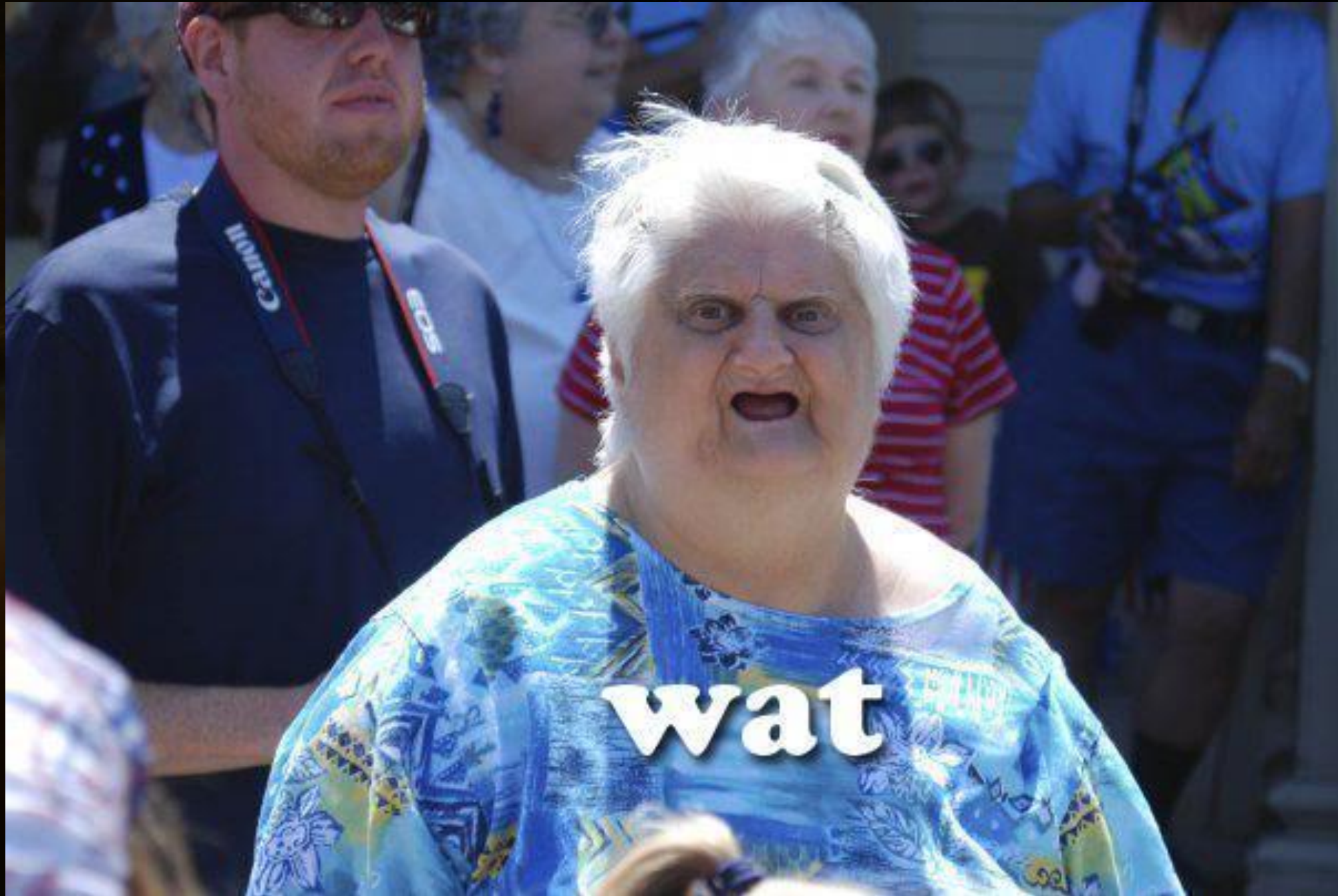
TDD vs TCR

?:



TDD:





Variations

BTCR

The Relaxed/TCTR

The Gentle

The Split

The buddy — Continuous TCR

The watch buddy

The Collaborator

Local Buddy, Remote Team (The Collaborator + The Buddy)

BTCR



```
if(build( ).failed)  
    return
```

```
if(test( ).success)  
    commit( )  
else  
    revert( )
```

The Relaxed/TCTR



```
if(build().failed)  
    return
```

```
if(test().success)  
    commit()
```

```
else  
    revert('src/main')
```



The Buddy - Continuous TCR



```
while(true) {  
    tcr()  
}  
  
function tcr() {  
    if(build().failed)  
        return  
  
    if(test().success)  
        commit()  
    else  
        revert()  
}
```

The Watch Buddy

```
while(true) {  
    block_until_change_in_directory( 'src' )  
    tcr()  
}  
  
function tcr() {  
    if(build().failed)  
        return  
  
    if(test().success)  
        commit()  
    else  
        revert()  
}
```


The Storyteller: Beyond the Buddy



Hey, I recognized an error at line 87 at file Source.java, as I executed the Test 'testFeatureX.java'. To get in a green state ASAP, I delete the lines, that caused the problems. OK?

It seems to work so far. I save the changes and rerun the tests.

TDD and TCR



Do we need hybrid solution?

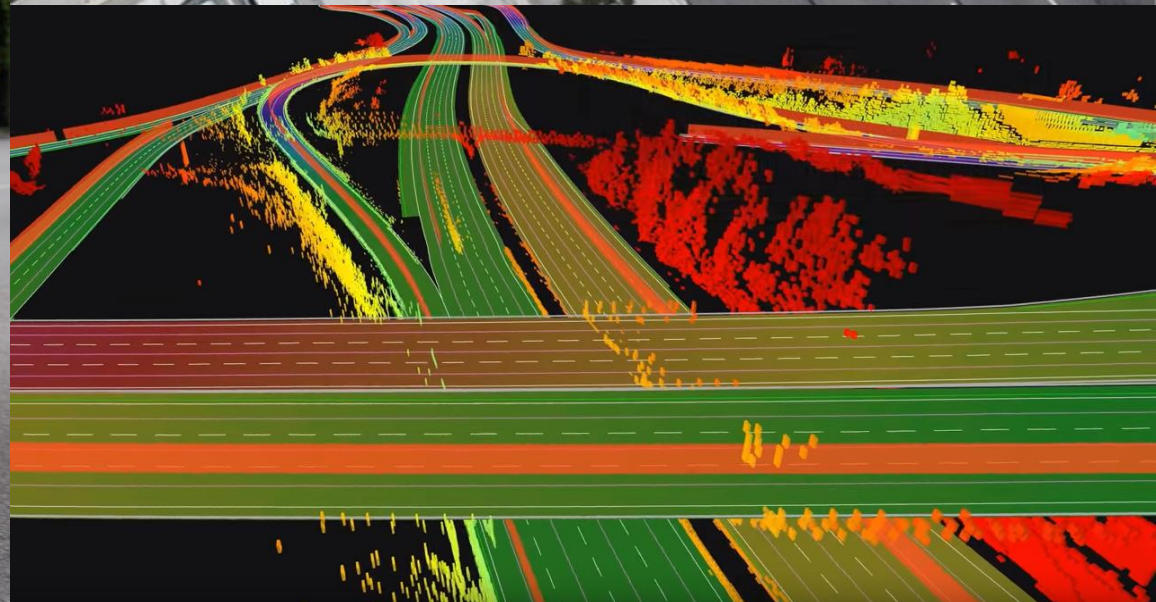
E.g.

(test && commit) || test_code_only || revert

WOW, can I use it?

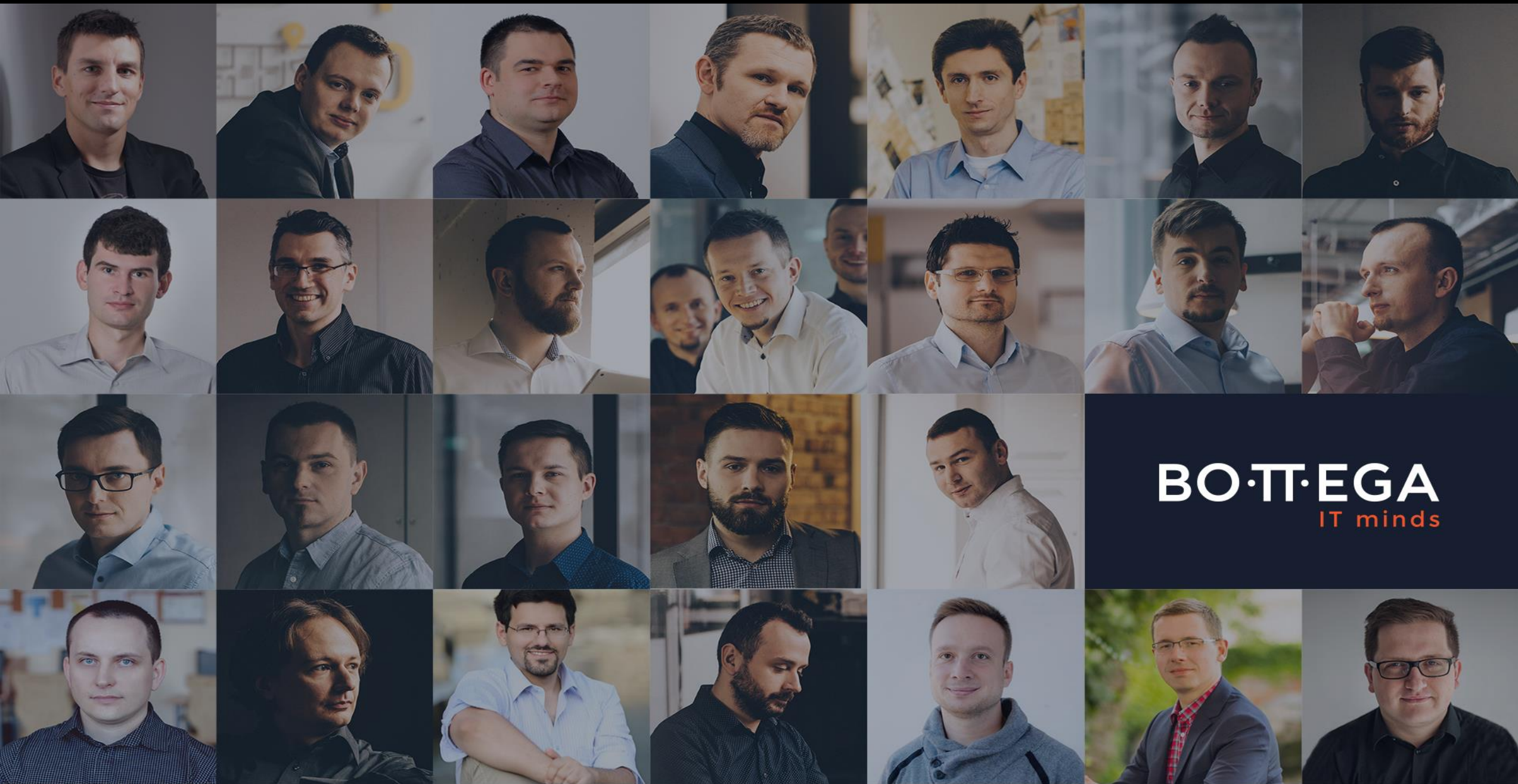


TOMTOM





Well yes, but actually no





Takeaways

Alternative workflow.

Small commits.

Cheap to get to known state.

Work/think for minutes, seconds.

Experiment, combine with TDD.

Thank You!

<https://bitbucket.org/piotrstawirej/tcr>

<http://piotrstawirej.bitbucket.io>

@p_stawirej

