



@brunoboucard

42skillz 



@tpierrain

# LA JOURNÉE DU DOMAIN-DRIVEN DESIGN ( AVEC UNE POINTE DE BDD )

FEBRUARY 19 2018  
AXA - WASQUEHAL

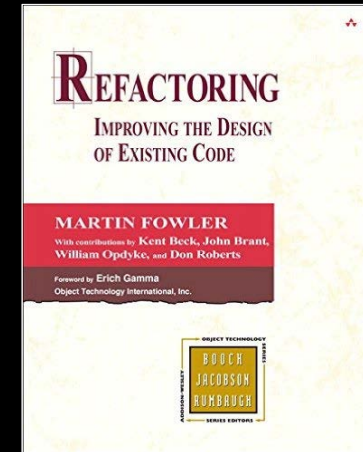
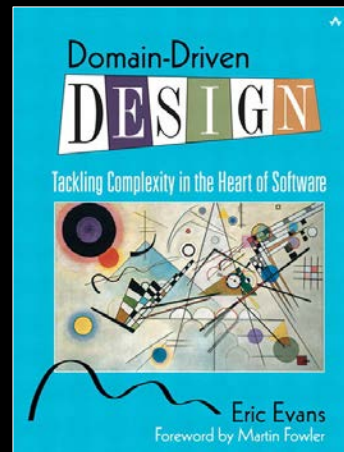
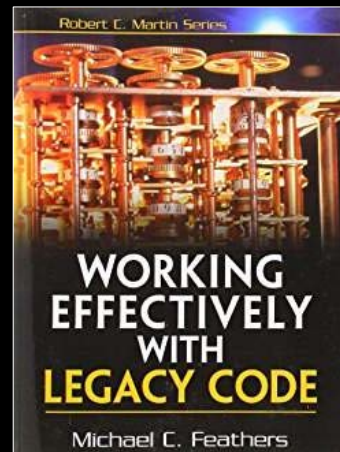
# WELCOME

CAN YOU PLEASE SIGN THE FORM ;-)

## QUESTION

WHAT ARE YOUR EXPECTATIONS  
FOR TODAY?

# WE HAVE SOME GIFTS FOR YOU



#DDDAXA

# - MENU -

WELCOME

EXAMPLE MAPPING (WORKSHOP)

PAUSE

TEST HARNESS (LIVE CODING)

PAUSE

DDD ALL THE WAY (LIVE CODING)

PAUSE

HEXAGONAL ARCHITECTURE FTW! (LIVE CODING)

WRAP-UP AND GIFTS

PREAMBLE:

EXAMPLE MAPPING

# WHAT'S BDD?

BDD PRACTITIONERS EXPLORE, DISCOVER, DEFINE  
THEN DRIVE OUT THE DESIRED BEHAVIOUR OF SOFTWARE  
USING CONVERSATION, CONCRETE EXAMPLES AND AUTOMATED TESTS

MATT WYNNE, THE CUCUMBER BOOK

# BEHAVIOUR DRIVEN DEVELOPMENT DEFINED



Discovery

Formulation

Automation



# EXAMPLE MAPPING

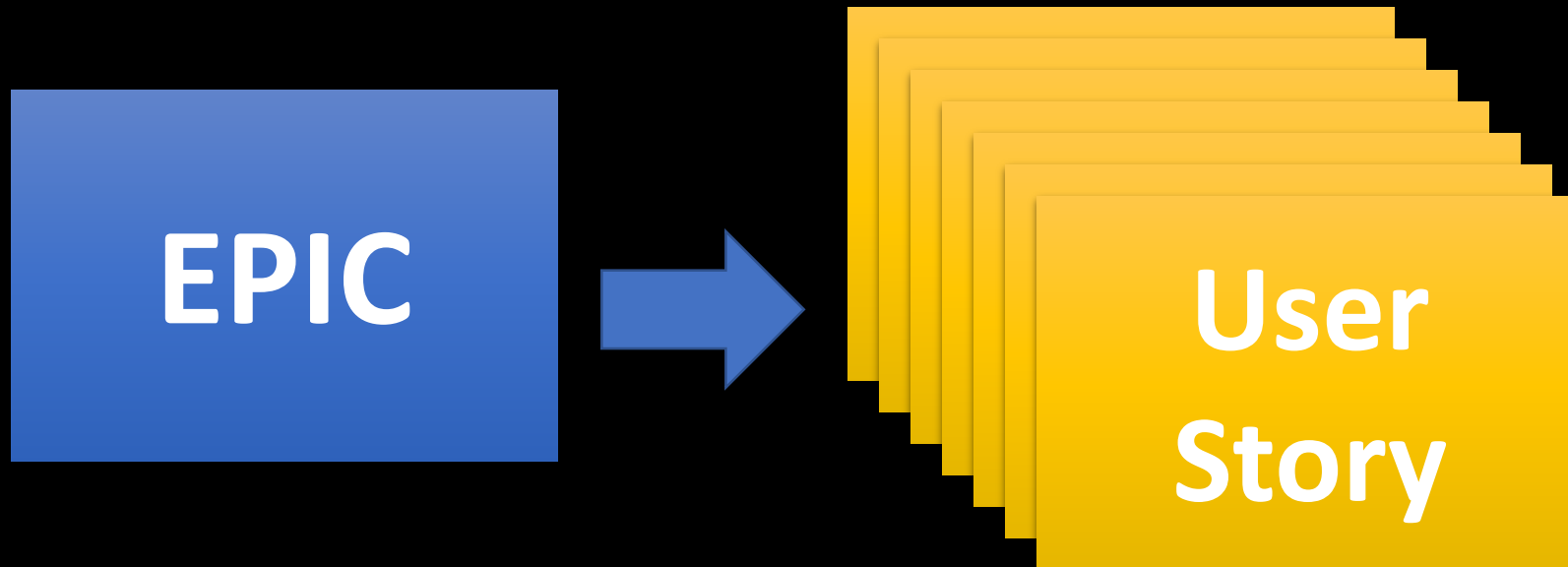
Discovery



```
graph TD; Discovery[Discovery] --> Arrow[ ];
```

ONE MINUTE, PLEASE!  
BACK TO BASICS

# HOW WE CRUNCH EPIC IN USER STORIES?



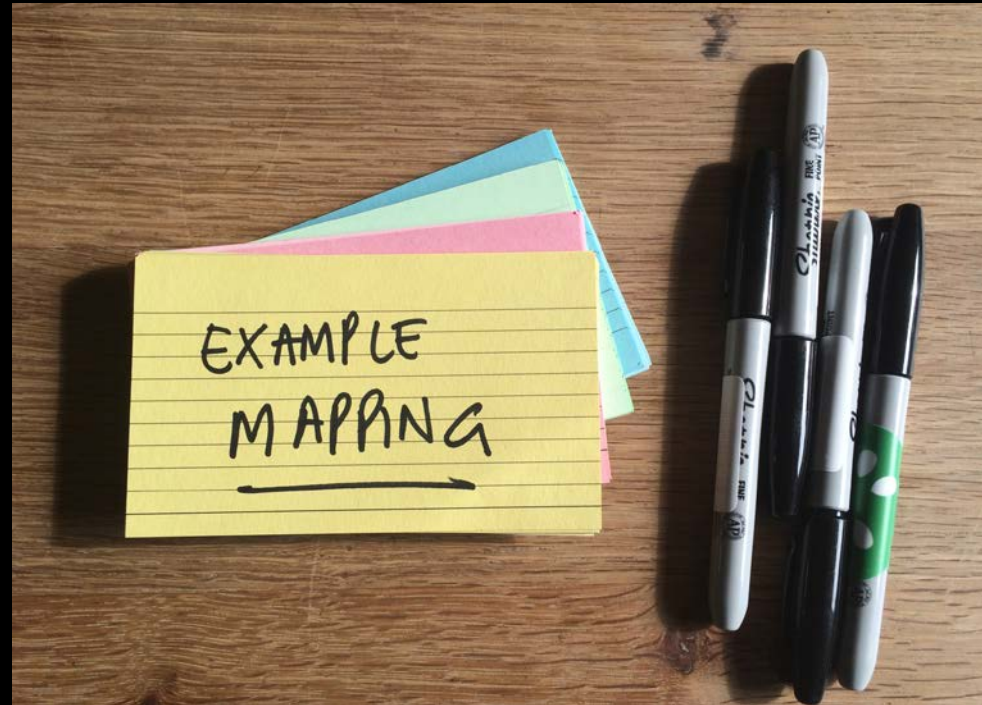
# ARE YOU INVEST?

**I**ndependent  
**N**egotiable  
**V**aluable  
**E**stimable  
**S**mall (Sized appropriately)  
**T**estable

<http://agileforall.com/wp-content/uploads/2012/01/Story-Splitting-Flowchart.pdf>

HOW DO WE CRUNCH  
A USER STORY?

# WE USE EXAMPLE MAPPING AS DISCOVERY WORKSHOP

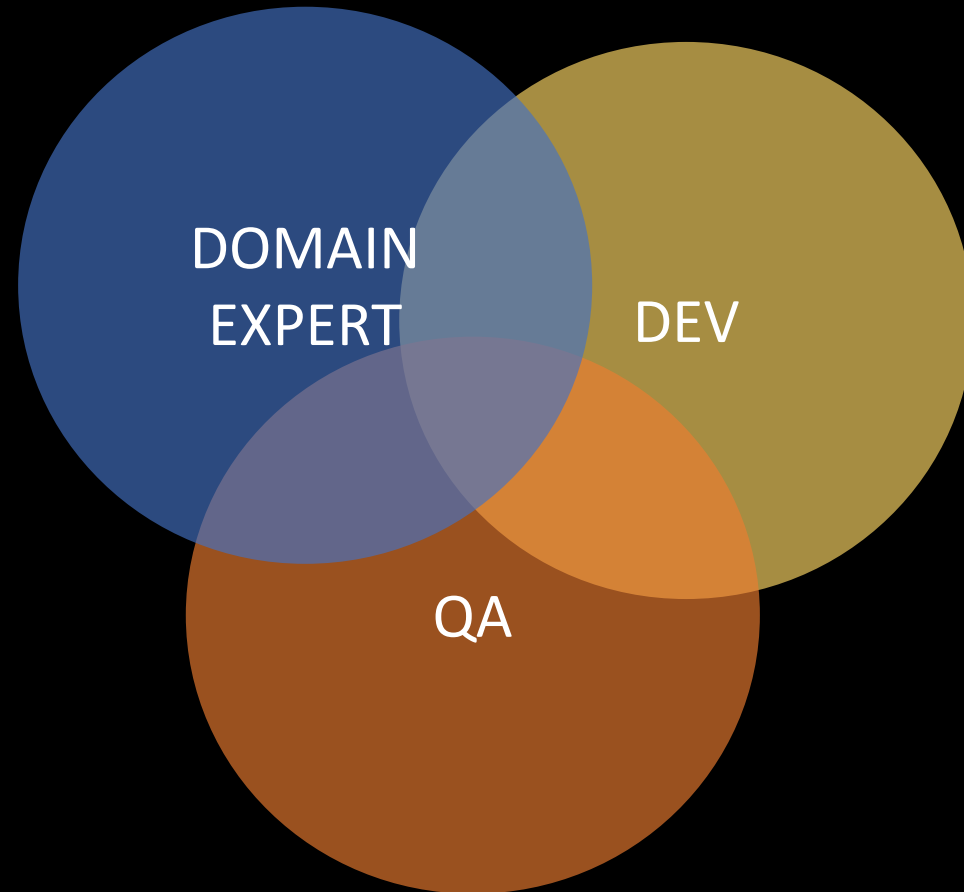


LOW-TECH METHOD – MATT WYNNE

<https://cucumber.io/blog/2015/12/08/example-mapping-introduction>

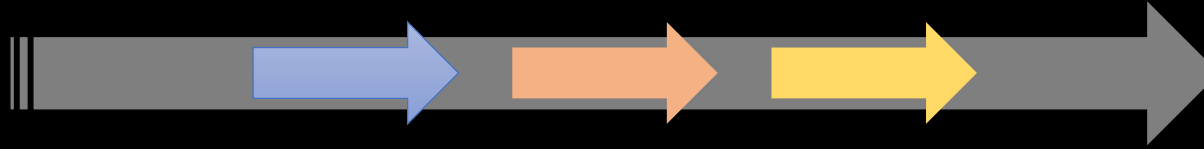
LOW-TECH BUT  
STRUCTURED

# ALIGN MENTAL MODELS

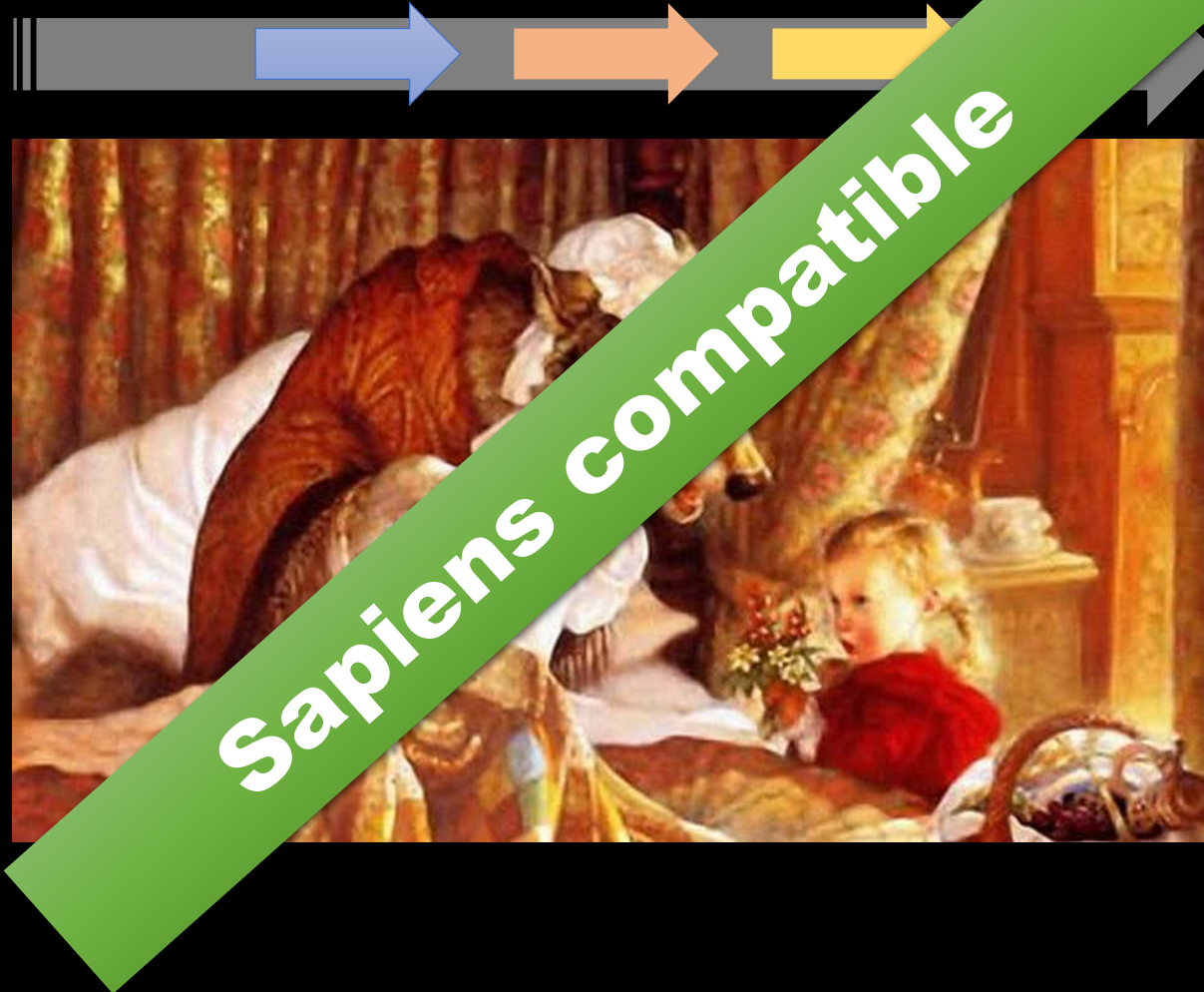




# ONCE UPON A TIME A STORY



# ONCE UPON A TIME A STORY



# COLLABORATIVE WORKSHOP

DOMAIN  
EXPERT

QA

DEV

#DDDaxa



Having conversations  
is more important than  
capturing conversations  
is more important than  
automating conversations

Liz Keogh

# STRUCTURED CONVERSATIONS

DOMAIN  
EXPERT

QA

DEV

#DDDaxa

KEEP  
CALM  
AND  
COME  
WITH  
AN  
EXAMPLE

Concrete examples please!



25 MINUTES PER USER STORY

DOMAIN  
EXPERT

QA

DEV

#DDDaxa

***FAST &  
FURIOUS***

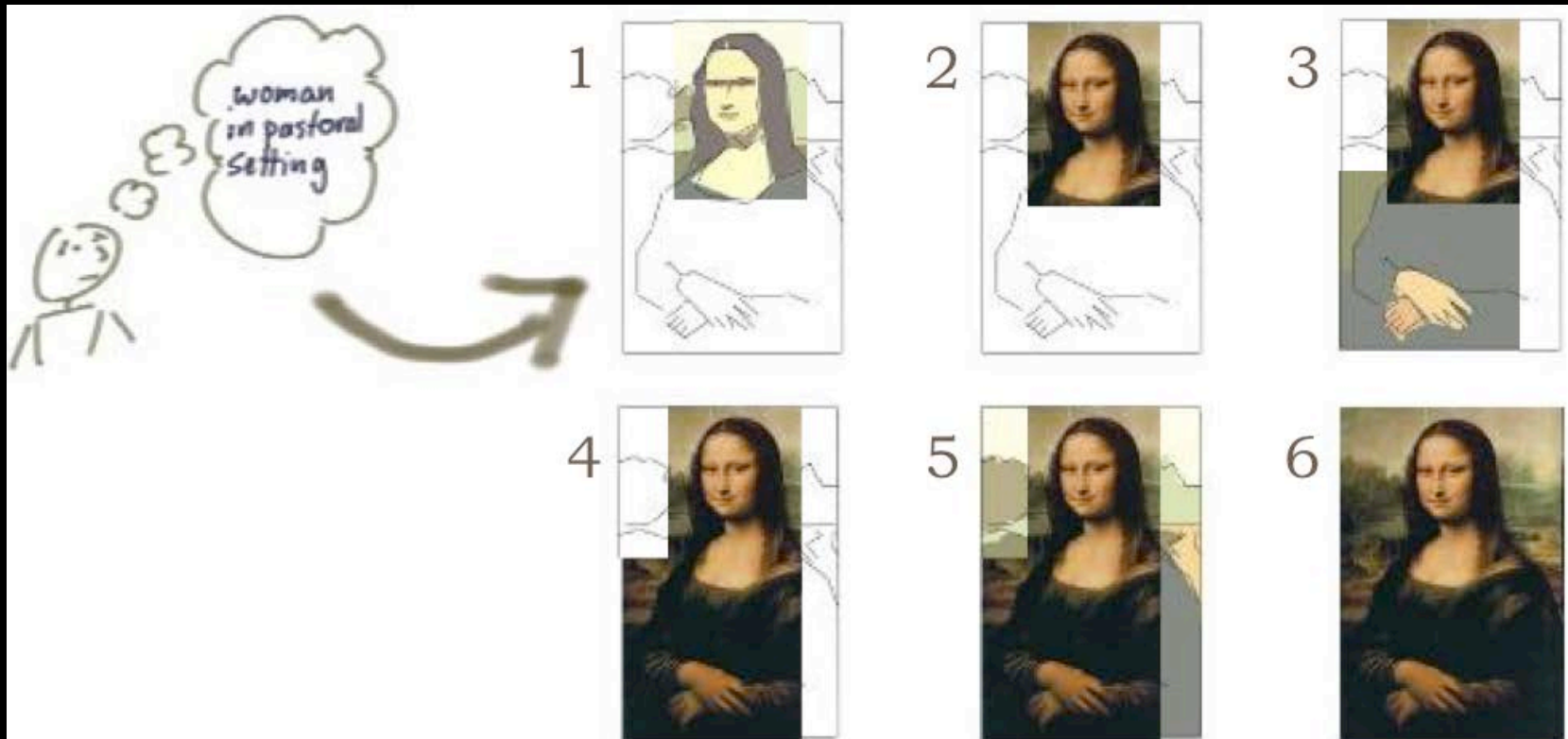
# PROGRESSIVE FOCUS

DOMAIN  
EXPERT

QA

#DDData

DEV



# THE WORKSHOP SHOULD ENDING WITH CONSENSUS

DOMAIN  
EXPERT

QA

DEV

#DDDaxa



# THE WORKSHOP SHOULD ENDING WITH CONSENSUS

DOMAIN  
EXPERT

QA

DEV

#DDData





WELL ...

I WANT AN EXAMPLE PLEASE!

# SHOUTY PRODUCT

DOMAIN  
EXPERT

QA

DEV

#DDDaxa

A NEW TWITTER-LIKE SOCIAL MEDIA APP – CALLED **SHOUTY**.

USERS OF THE APP WILL BE ABLE TO 'SHOUT' – AND WILL BE HEARD BY OTHER USERS WHO ARE WITHIN **1000M** OF THE SHOUTER.

THE TARGET PLATFORM IS GPS-ENABLED SMARTPHONES

# EXAMPLE MAPPING

DOMAIN  
EXPERT

QA

DEV

#DDDaxa

Hear Shout

*In range shout is heard*

Give us a concrete example  
please!

DOMAIN  
EXPERT

QA

DEV

# EXAMPLE MAPPING

Hear Shout

*In range shout is heard*

Not hear own shout

- \* Lucy is at [0, 0]
- \* Sean is at [0, 900]
- \* Sean shouts

=> Lucy hears Sean

Give us a concrete example  
please!

DOMAIN  
EXPERT

QA

DEV

# EXAMPLE MAPPING

Hear Shout

*In range shout is heard*

Not hear own shout

- \* Lucy is at [0, 0]
- \* Sean is at [0, 900]
- \* Sean shouts

=> Lucy hears Sean

- \* Lucy is at [0, 0]
- \* Sean is at [800, 800]
- \* Sean shouts

=> Sean hears nothing

DOMAIN  
EXPERT

QA

DEV

# EXAMPLE MAPPING

**Hear Shout***In range shout is heard***Not hear own shout****Out of range shout is not  
heard**

- \* Lucy is at [0, 0]
- \* Sean is at [0, 900]
- \* Sean shouts

**=> Lucy hears Sean**

- \* Lucy is at [0, 0]
- \* Sean is at [800, 800]
- \* Sean shouts

**=> Sean hears nothing****Give us a concrete example  
please!**

DOMAIN  
EXPERT

QA

DEV

# EXAMPLE MAPPING

**Hear Shout***In range shout is heard***Not hear own shout****Out of range shout is not heard**

- \* Lucy is at [0, 0]
- \* Sean is at [0, 900]
- \* Sean shouts

**=> Lucy hears Sean**

- \* Lucy is at [0, 0]
- \* Sean is at [800, 800]
- \* Sean shouts

**=> Sean hears nothing**

- \* Lucy is at [0, 0]
- \* Sean is at [0, 1000]
- \* Sean shouts

**=> Lucy hears nothing**

DOMAIN  
EXPERT

QA

DEV

# EXAMPLE MAPPING

**Hear Shout*****Shout a message > 2000  
characters******In range shout is heard*****Not hear own shout****Out of range shout is not  
heard**

- \* Lucy is at [0, 0]
- \* Sean is at [0, 900]
- \* Sean shouts

**=> Lucy hears Sean**

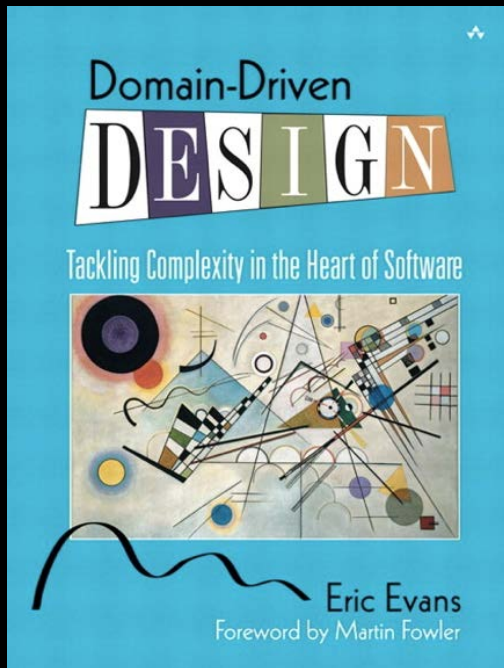
- \* Lucy is at [0, 0]
- \* Sean is at [800, 800]
- \* Sean shouts

**=> Sean hears nothing**

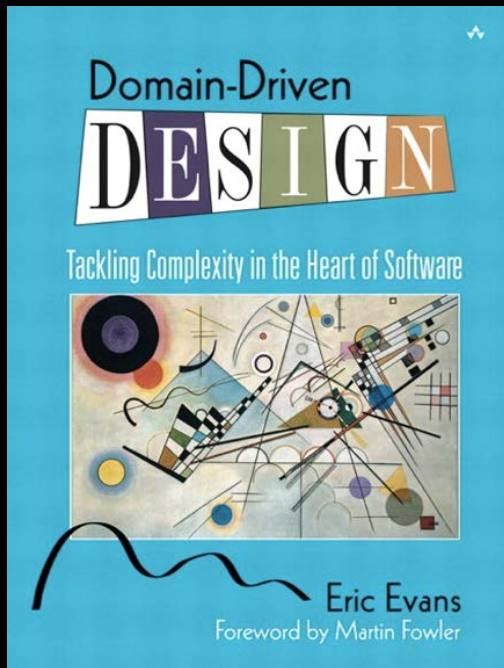
- \* Lucy is at [0, 0]
- \* Sean is at [0, 1000]
- \* Sean shouts

**=> Lucy hears nothing**



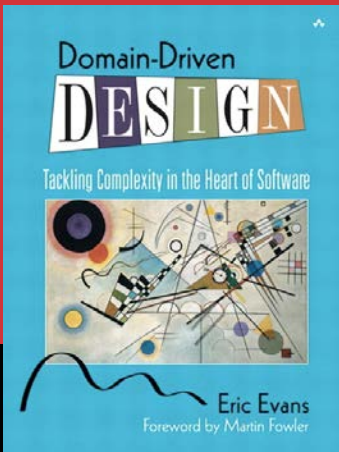


?



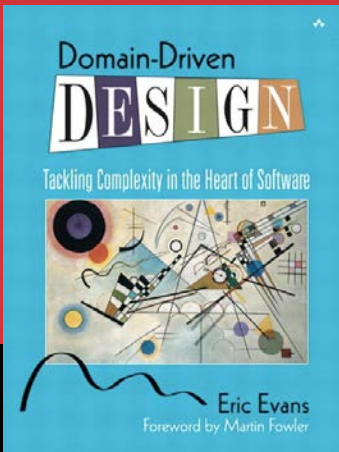
# AN APPROACH

# A TOOLBOX



# AN APPROACH

FOCUS ON BUSINESS VALUE  
LANGUAGE AND CONTEXTS ARE KEY  
MAKE THE IMPLICIT, EXPLICIT



# A TOOLBOX

## TACTICAL PATTERNS

## STRATEGICAL PATTERNS



# OUR DOMAIN FOR TODAY



# THE CONTEXT

US:



TRAIN-TRAIN, THE  
STARTUP

IN FRONT OF:

HASSAN-CEHEF, THE  
TRAIN OPERATOR

TRAIN-TRAIN

## OUR DOMAIN FOR TODAY

EIL CIV 0019		Ticket to LONDON ST-PANCRAS Traveller: JIM MORIARTY			
Train: <b>9043</b>		Date: <b>07/04</b>		<p>HOLD THIS BAR CODE FIRMLY AGAINST THE READER ON THE TICKET GATE</p> 	
From <b>PARIS NORD</b>		Departs <b>16:13</b> local time			
To <b>LONDON ST-PANCRAS</b>		Arrives <b>17:39</b> local time			
Class: <b>Standard Premier</b>		Coach <b>A</b>			
Booking reference(s): <b>QTFPZJ</b>		Seat <b>8</b>			
You have booked a regular seat					
<p><b>Additional information</b> Check-in closes 30 minutes before departure EXCHANGEABLE WITH FEE BEFORE DEPARTURE / NON REFUNDABLE</p>					
EUR 189.00					



TRAIN-TRAIN

OUR DOMAIN FOR TODAY

EIL

CIV 0019

Ticket to LONDON ST-PANCRAS

Traveller: JIM MORIARTY

Train

9043

Date: 07/04

Departs

16:13 local time

From

PARIS NORD

To

LONDON ST-PANCRAS

Class: Standard Premier

Booking reference(s)

QTFPZJ

You have booked a regular seat

Coach

A

Seat

8

Additional information


Check-in closes 30 minutes before departure

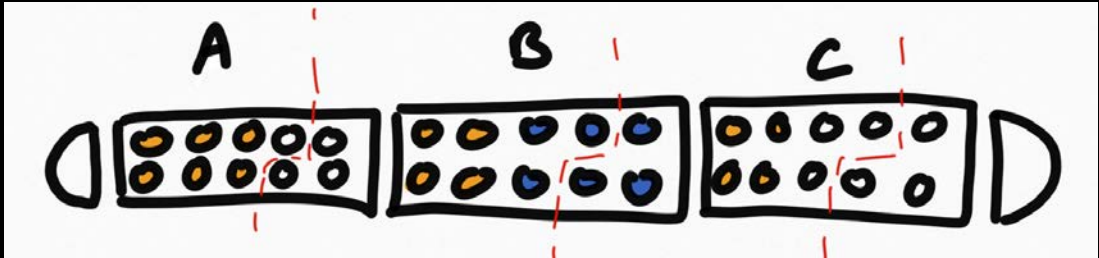
EXCHANGEABLE WITH FEE BEFORE DEPARTURE / NON REFUNDABLE

EUR 189.00

EUROSTAR™

HOLD THIS BAR CODE FIRMLY AGAINST THE READER ON THE TICKET GATE





TRAIN-TRAIN

# THE CREW

THE DOMAIN GUY



THE TECHIES



TRAIN-TRAIN

# CHALLENGES

- TRAIN FILLING OPTIMIZATIONS
- INTEGRATION WITH 2 HASSAN CEHEF BACKENDS

## TRAIN-TRAIN

# BIZ RULES<sup>(\*)</sup>

- FOR A TRAIN OVERALL, NO MORE THAN 70 PERCENT SEATS MAY BE RESERVED IN ADVANCE
- YOU MUST PUT ALL THE SEATS FOR ONE RESERVATION IN THE SAME COACH

(\*) FROM EMILY BACHE'S TRAIN RESERVATION KATA

TRAIN-TRAIN

# SITUATION TODAY

- A V1 HAS BEEN RELEASED FEW MONTH AGO
- NEW RULE TO BE IMPLEMENTED
- EVOLUTION COST HAS RISEN
- RECURRENT PENALTIES FROM THE OPERATOR

EXAMPLE MAPPING THIS!

I WANT TO RESERVE 2 SEATS



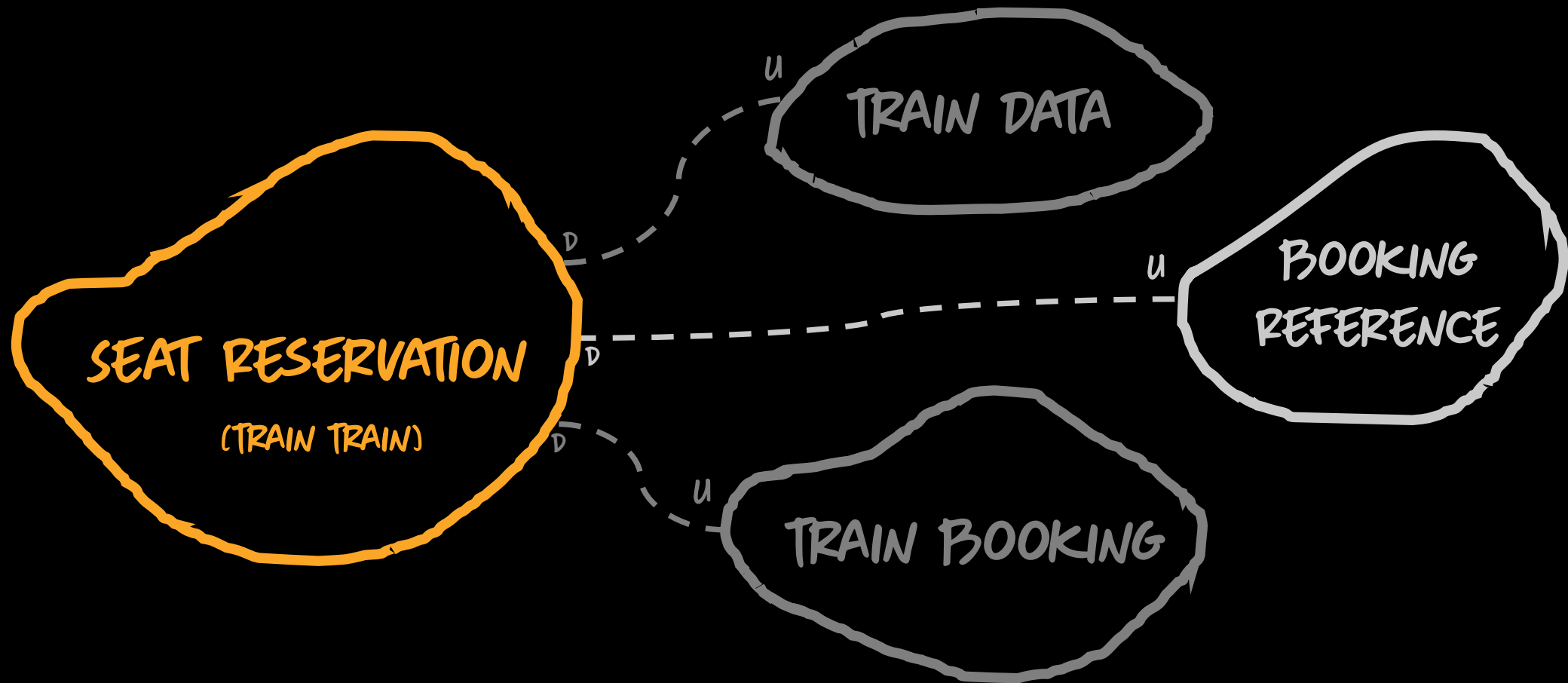
# SOME WORDS FROM OUR DOMAIN EXPERT

- TRAIN
- TRAIN ID
- SEATS
- COACHES
- RESERVATION ATTEMPT
- FULFILLED

- TO BOOK
- BOOKING REFERENCE
- RESERVATION
- RESERVATION FAILURE
- ...



# MANY BOUNDED CONTEXTS...





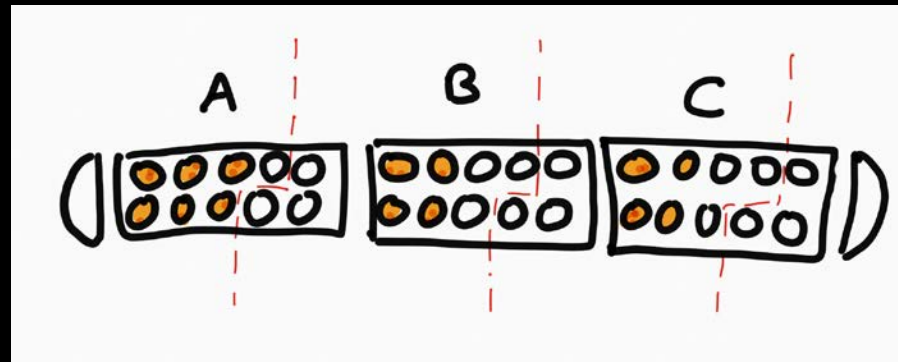
TRAIN-TRAIN

DEMO!

## OUR MISSION

## NEW RULE!

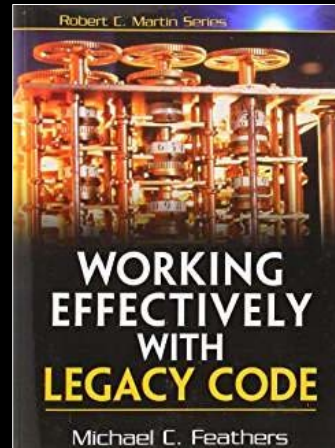
"IDEALLY, NO INDIVIDUAL COACH SHOULD HAVE NO MORE THAN 70 PERCENT RESERVED SEATS EITHER"



TRAIN-TRAIN

EPISODE 1:

WE NEED A TEST HARNESS!



TRAIN-TRAIN

LET'S DIVE INTO THE CODE



TRAIN-TRAIN

OUR SYSTEM IS REQUESTED LIKE...

I WANT 3 SEATS FOR TRAIN 9...

```
{"train_id": "9043-2017-04-07", "number_of_seats": 3}
```

TRAIN-TRAIN

OUR SYSTEM ANSWERS LIKE...

WHEN RESERVATION **SUCCEED**

```
{"train_id": "1434-2017-06-07", "booking_reference": "75bcd15", "seats": ["1A", "1B"]}
```

WHEN RESERVATION FAIL

```
{"train_id": "1434-2017-06-07", "booking_reference": "()", "seats": []}
```



TRAIN-TRAIN

TIME TO ADD OUR FIRST  
ACCEPTANCE TESTS

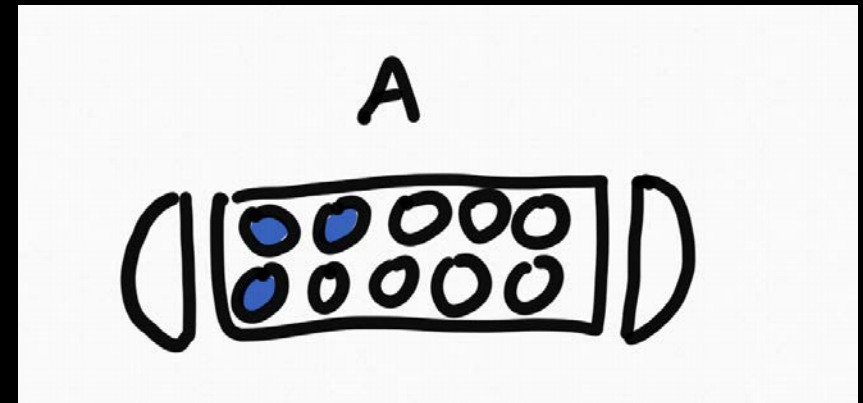
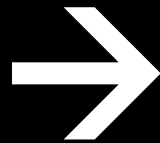
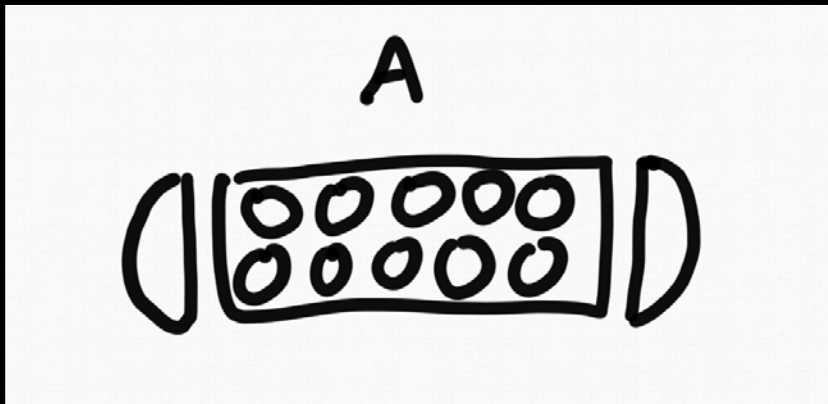
TRAIN-TRAIN

TEST #1

SIMPLE RESERVATION

TRAIN-TRAIN

## TEST #1 - SIMPLE RESERVATION



I WANT 3 SEATS

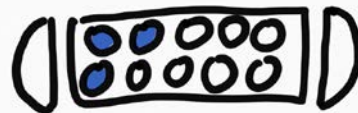
TRAIN-TRAIN

## TEST #1 - SIMPLE RESERVATION

### EXPECTED RESULT

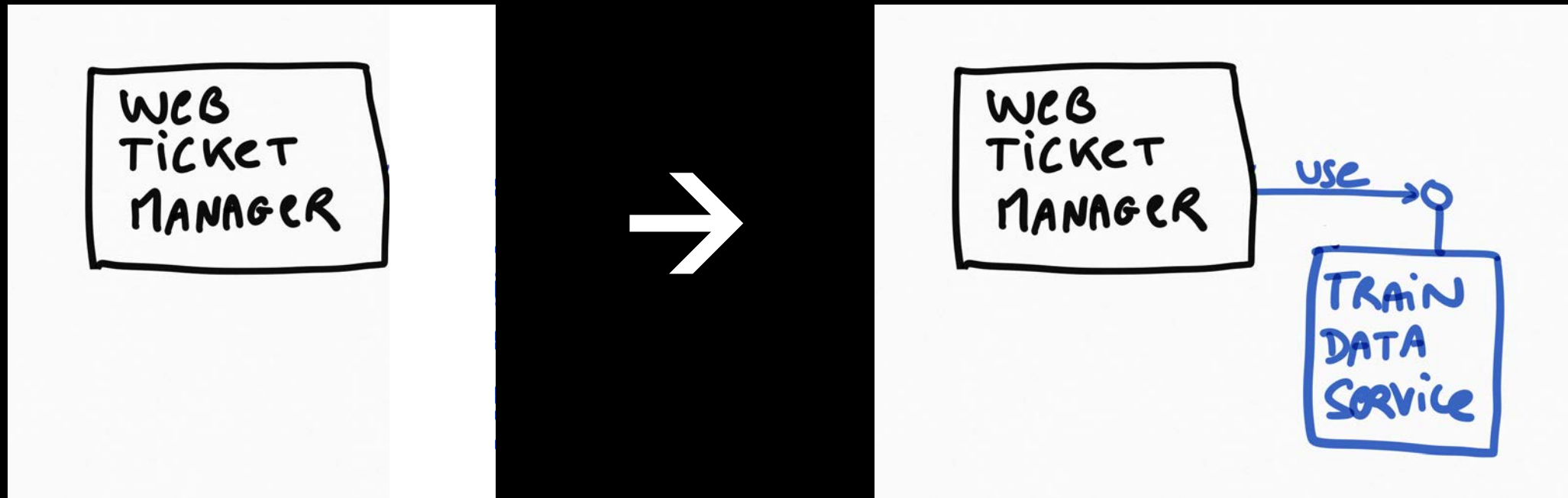
```
{"train_id": "9043-2017-06-07", "booking_reference": "75bcd15", "seats": ["1A", "2A", "3A"]}
```

A



## TRAIN-TRAIN

WE NEED SOMETHING TO BE **STUBBED**!



## TRAIN-TRAIN

# HOW HASSAN-CEHEF PROVIDES TRAIN TOPOLOGIES

```
{"seats": {"1A": {"booking_reference": "", "seat_number": "1", "coach": "A"}, "2A": {"booking_reference": "", "seat_number": "2", "coach": "A"}}}
```

---

## TRAIN-TRAIN

## HOW HASSAN-CEHEF PROVIDES BOOKING REFERENCES

You can use this service to get a unique booking reference. Make a GET request to:

```
http://localhost:8082/booking_reference
```

This will return a string that looks a bit like this:

```
75bcd15
```

TRAIN-TRAIN

NEXT!



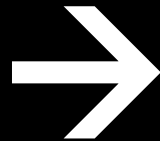
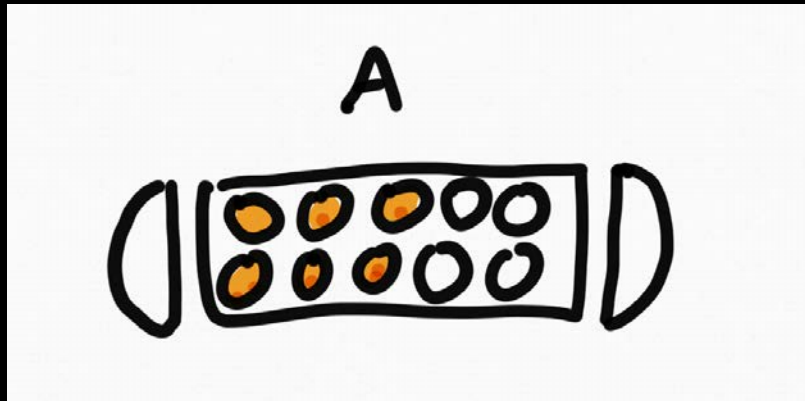
TRAIN-TRAIN

TEST #2

THIS TRAIN IS FULL...

TRAIN-TRAIN

TEST #2 – THIS TRAIN IS FULL




CAN'T EXCEED  
70 PERCENT OF TRAIN  
CAPACITY!

I WANT 2 SEATS

## TRAIN-TRAIN

TEST #2 — THIS TRAIN IS FULL

CAN'T EXCEED  
70 PERCENT OF TRAIN  
CAPACITY!



```
{"train_id": "1434-2017-06-07", "booking_reference": "", "seats": []}
```

TRAIN-TRAIN

NEXT!

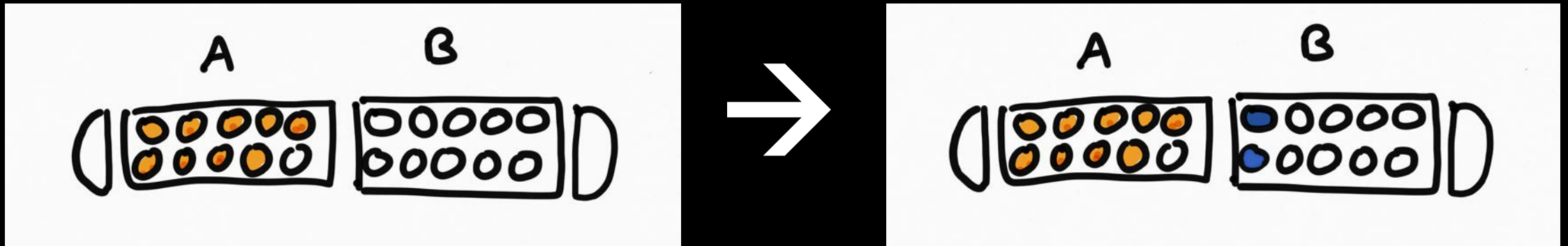
TRAIN-TRAIN

TEST #3

ALL SEATS IN THE SAME COACH  
PLEASE!

TRAIN-TRAIN

TEST #3 – SAME COACH PLEASE!



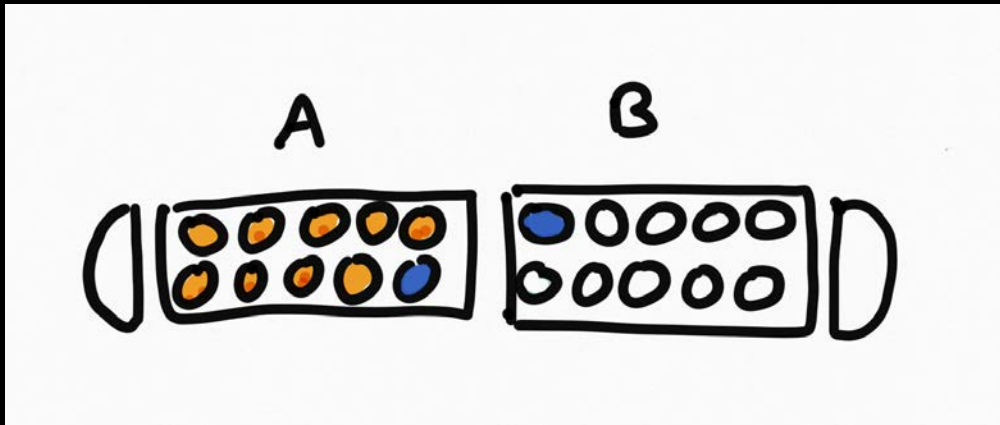
I WANT 2 SEATS

TRAIN-TRAIN

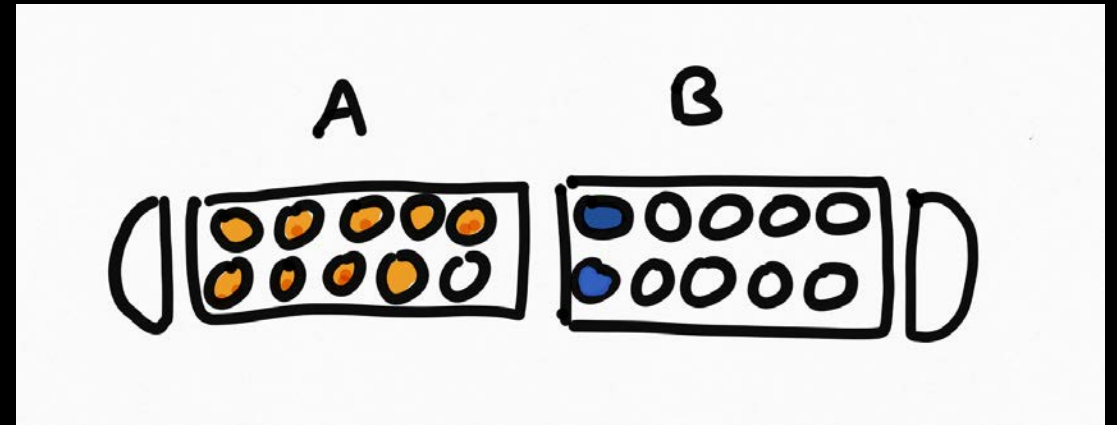
HOUSTON?

TRAIN-TRAIN

TEST #3 – SAME COACH PLEASE!



WHAT WE HAVE...

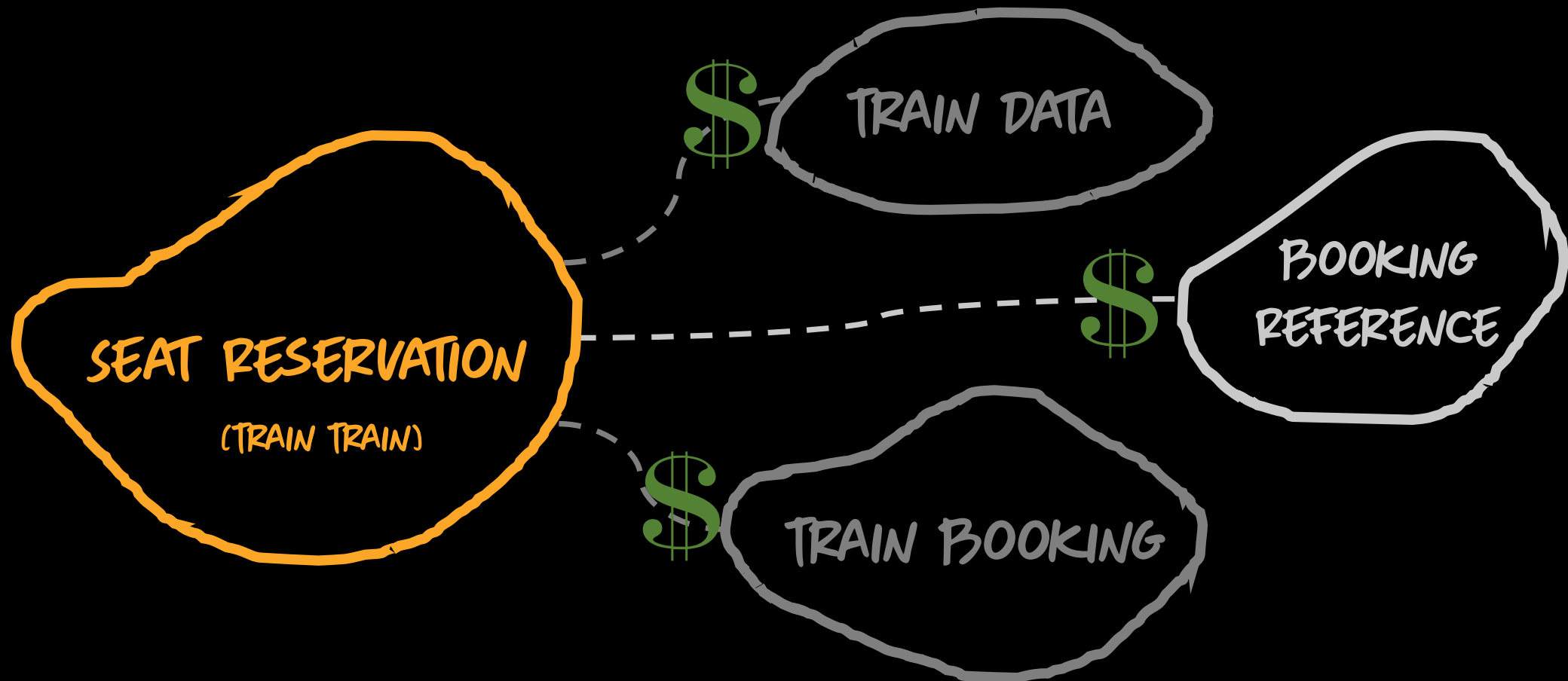


WHAT WE SHOULD HAVE



EXTRA FEES? THAT MAKES SENSE...

EXTRA FEES? THAT MAKES SENSE...



TRAIN-TRAIN

FIX THAT BUG?

OR

ADD THE FEATURE?

TRAIN-TRAIN

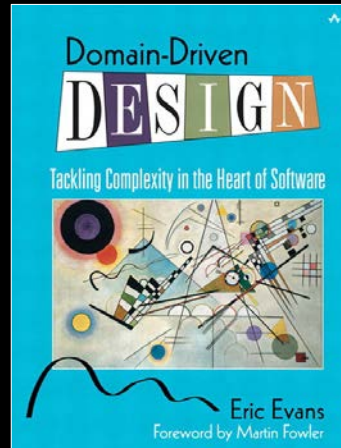
15 MINUTES PAUSE

NEXT: DDD

TRAIN-TRAIN

EPISODE 2:

WHERE IS THE DOMAIN?!?



TRAIN-TRAIN

# REMEMBER THE BUG?



TRAIN-TRAIN

# EXTRACT AND MOVE

## CLASS A

```
.....  
.....  
.....  
.....  
.....  
.....  
  
.....  
.....  
.....  
.....  
.....
```

## CLASS B

```
..  
.....  
.....  
.....
```



EXTRACT AND MOVE

CLASS A

.....

.....

.....

.....

.....

.....

.....

.....

..

.....

.....

CLASS B

..

.....

.....

.....

EXTRACT CODE

# EXTRACT AND MOVE

## CLASS A

.....  
.....  
  
.....  
.....  
  
..  
.....  
.....

.....  
.....  
.....  
.....

METHOD EXTRACTED

## CLASS B

..  
.....  
  
.....  
.....

## EXTRACT AND MOVE

CLASS A

```
.....  
.....  
  
.....  
.....  
  
..  
.....  
.....
```

CLASS B

```
..  
.....  
  
.....  
.....
```

```
.....  
.....  
.....  
.....
```

MOVE METHOD

## EXTRACT AND MOVE

### CLASS A

```
.....  
.....  
  
.....  
.....  
..  
.....  
.....
```

### CLASS B

```
..  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....
```

METHOD MOVED

# EXTRACT AND MOVE

## CLASS A

```
.....  
.....  
  
.....  
.....  
  
..  
.....  
.....
```

## CLASS B

```
..  
.....  
  
.....  
.....  
  
.....  
.....  
.....  
.....
```

TRAIN-TRAIN

SOME DDD PATTERNS

LET'S **ZOOM** 1 SECOND

SOME DDD PATTERNS

ENTITY  
OR  
VALUE TYPE?



## SOME DDD PATTERNS

# ENTITY

-

HAVE IDENTITY AND LIFECYCLE

MUTABLE

IDENTIFIER EQUALITY

SOME DDD PATTERNS

# VALUE TYPE

-

NO IDENTITY

IMMUTABLE

EQUALITY BY PROPERTIES

## SOME DDD PATTERNS

### ENTITY

-

HAVE IDENTITY AND LIFECYCLE

MUTABLE

IDENTIFIER EQUALITY

### VALUE TYPE

-

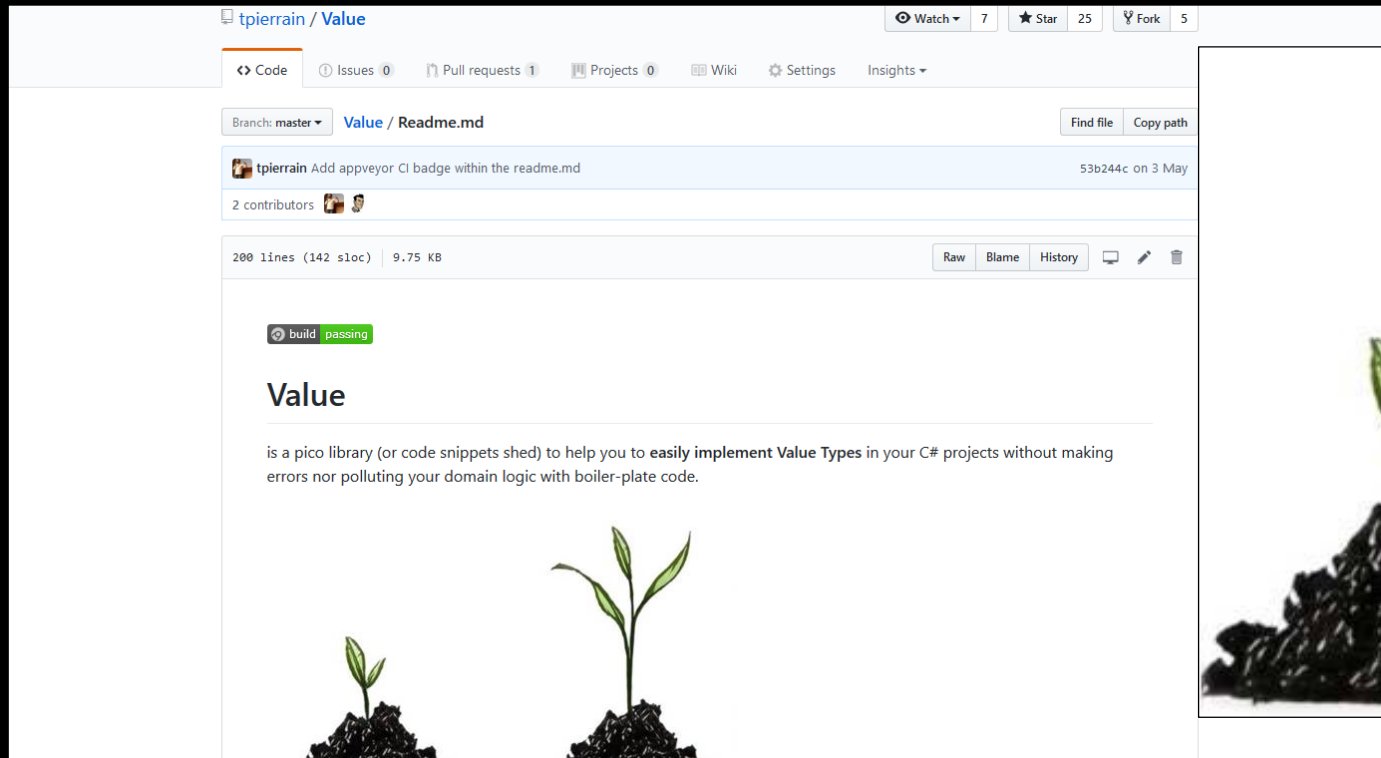
NO IDENTITY

IMMUTABLE

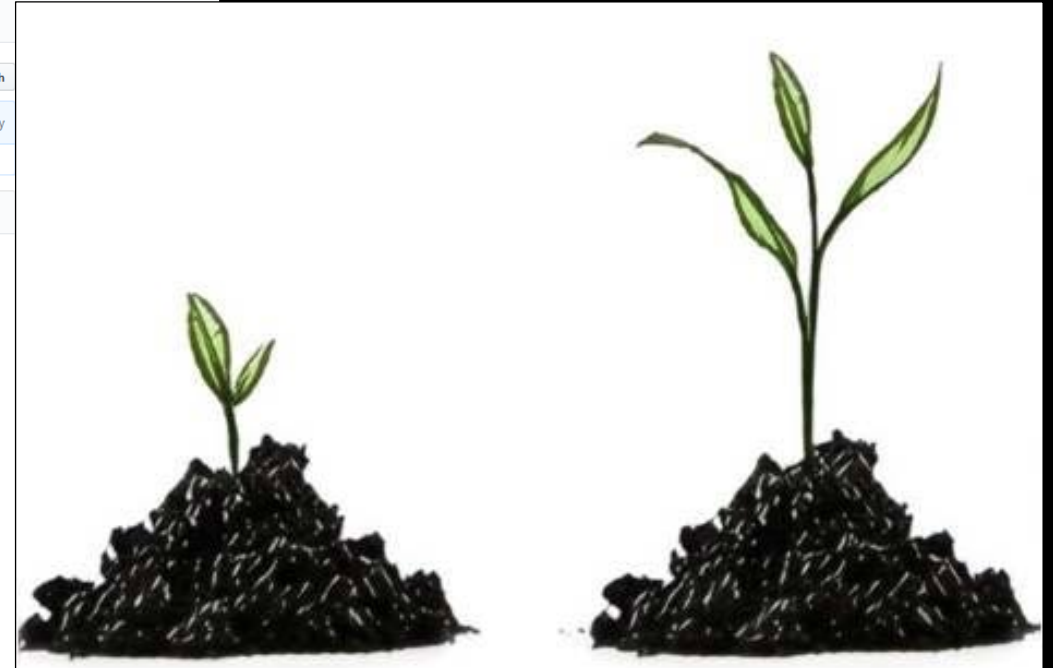
EQUALITY BY PROPERTIES

## SOME DDD PATTERNS

## VALUE TYPES POWERED BY "VALUE"



The screenshot shows the GitHub repository page for 'tpierrain / Value'. The repository has 7 watchers, 25 stars, and 5 forks. The main branch is 'master'. The README file is selected, showing a commit by 'tpierrain' with the message 'Add appveyor CI badge within the readme.md' on May 3. The repository contains 200 lines of code (142 SLOC) and is 9.75 KB in size. A green badge indicates that the build is passing. The README text describes 'Value' as a pico library (or code snippets shed) to help easily implement Value Types in C# projects without making errors or polluting domain logic with boiler-plate code. At the bottom of the README, there are two illustrations of small green seedlings growing out of mounds of dark soil.



## SOME DDD PATTERNS

# CLOSURE OF OPERATION

-

AN OPERATION WHOSE RETURN TYPE IS THE SAME AS  
THE TYPE OF ITS IMPLEMENTER

E.g.: `Coach otherCoach = coach.AddSeat(seat);`

SOME DDD PATTERNS

# CLOSURE OF OPERATION

-

DO YOU SEE THE BENEFITS?

SOME DDD PATTERNS

# CLOSURE OF OPERATION

-

FOSTER IMMUTABILITY

REDUCE COUPLING

( EASIER TO "MOVE" CODE )

Here, the Coach type stands by itself

SOME DDD PATTERNS

# AGGREGATE

-

COLLECTION OF ENTITIES AND VALUE TYPES  
TREATED AS A CONCEPTUAL WHOLE



SOME DDD PATTERNS

AGGREGATES IN HERE?

## SOME DDD PATTERNS

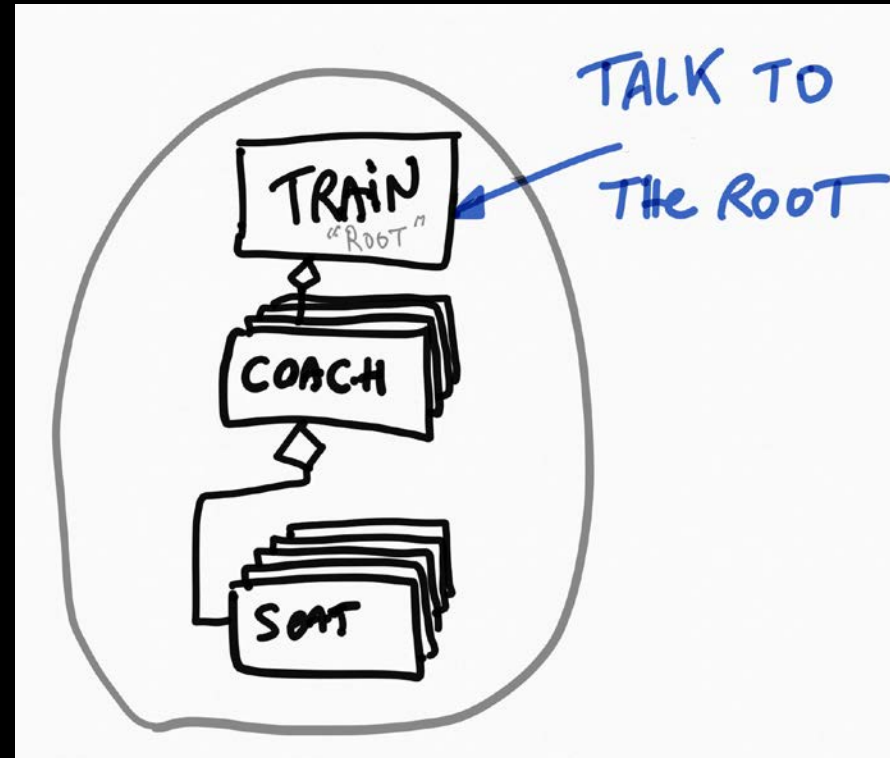
IS COACH AN  
AGGREGATE?

SOME DDD PATTERNS

NO.  
NOT YET!

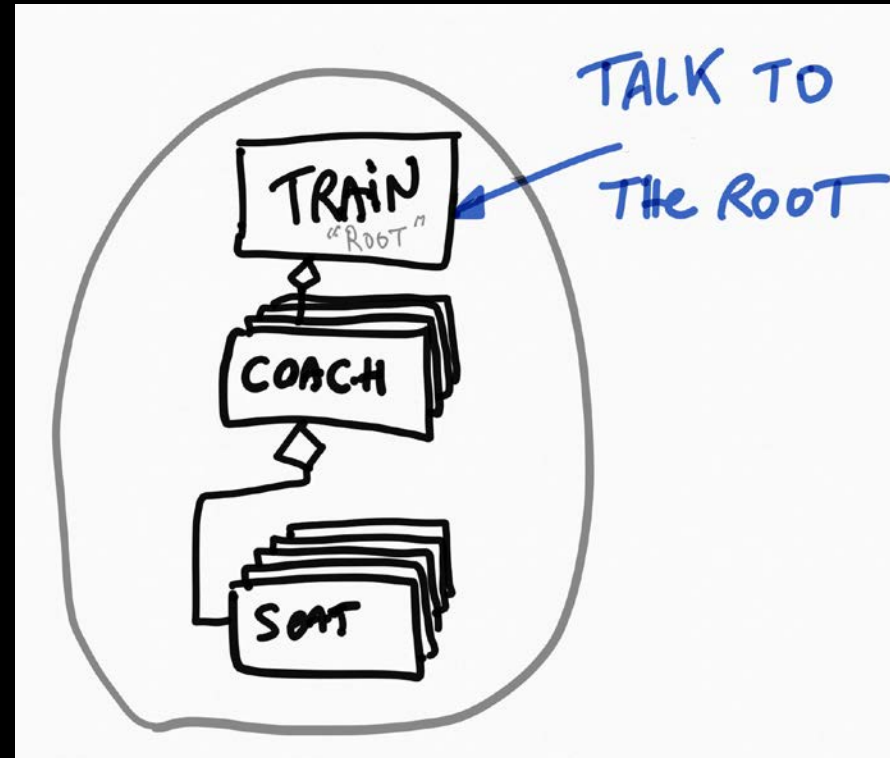
## SOME DDD PATTERNS

AGGREGATES NEEDS  
INVARIANTS



## SOME DDD PATTERNS

THINK ABOUT IT:  
TRAIN, COACH AND SEAT  
MAY ALL BE VALUE TYPES  
HERE ( IMMUTABLE )

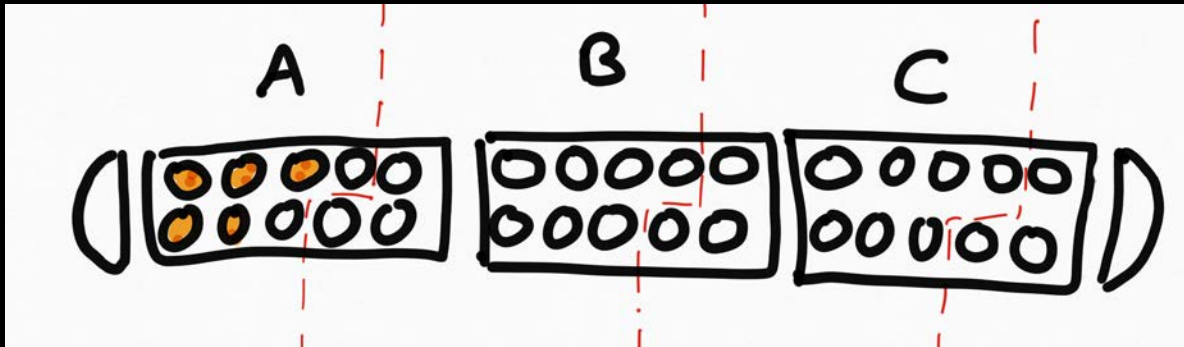


TRAIN-TRAIN

NEXT!

## TRAIN-TRAIN

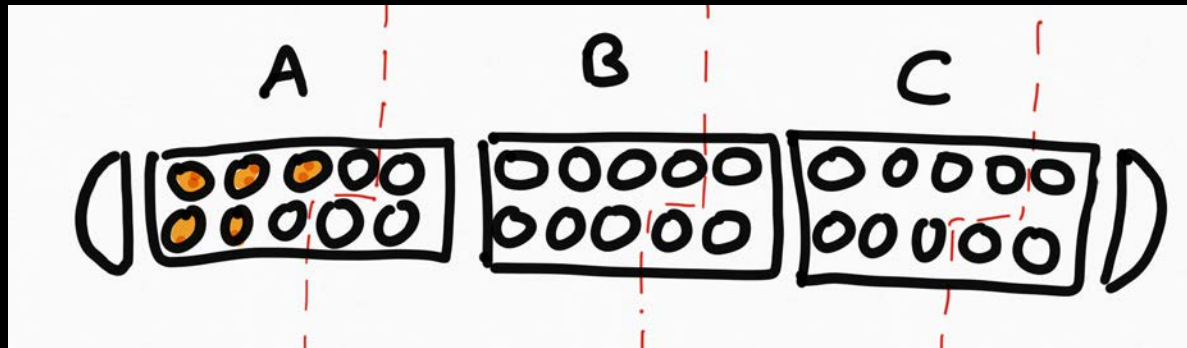
TEST #4 – IDEALLY, NO MORE THAN 70% OF COACH



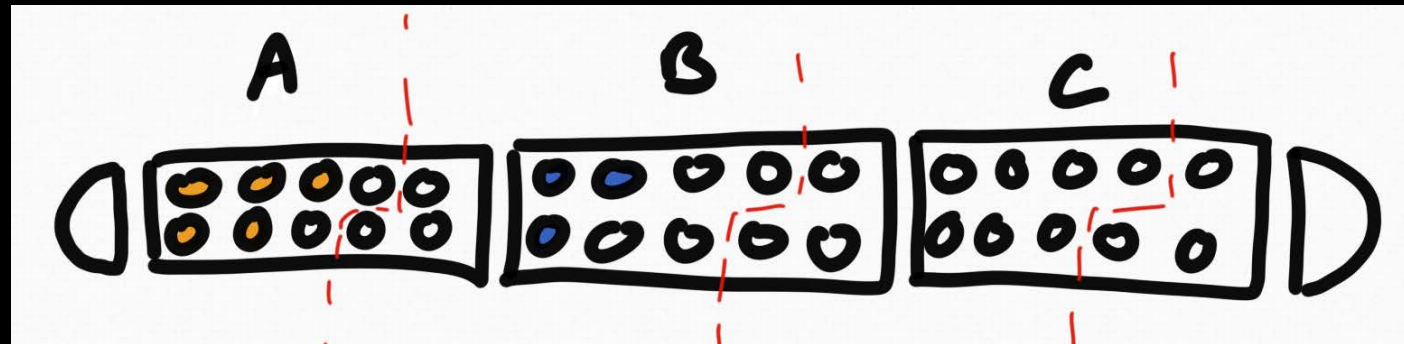
I WANT 3 SEATS

## TRAIN-TRAIN

TEST #4 – IDEALLY, NO MORE THAN 70% OF COACH



I WANT 3 SEATS



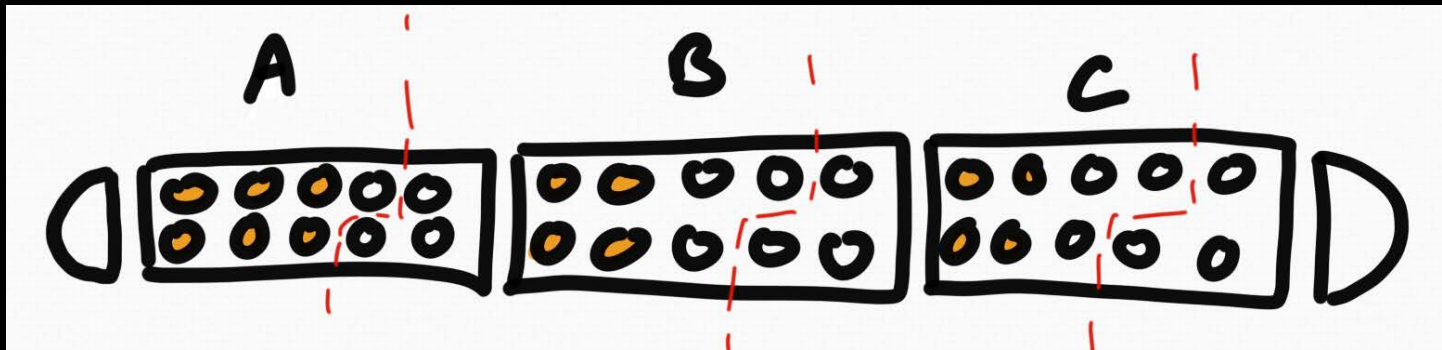


TRAIN-TRAIN

NEXT!

TRAIN-TRAIN

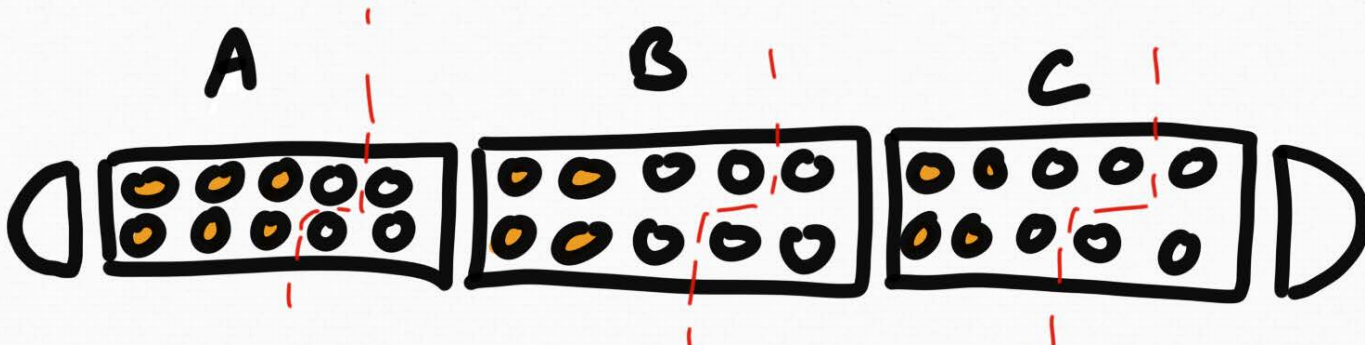
TEST #5 — NON IDEAL CASE



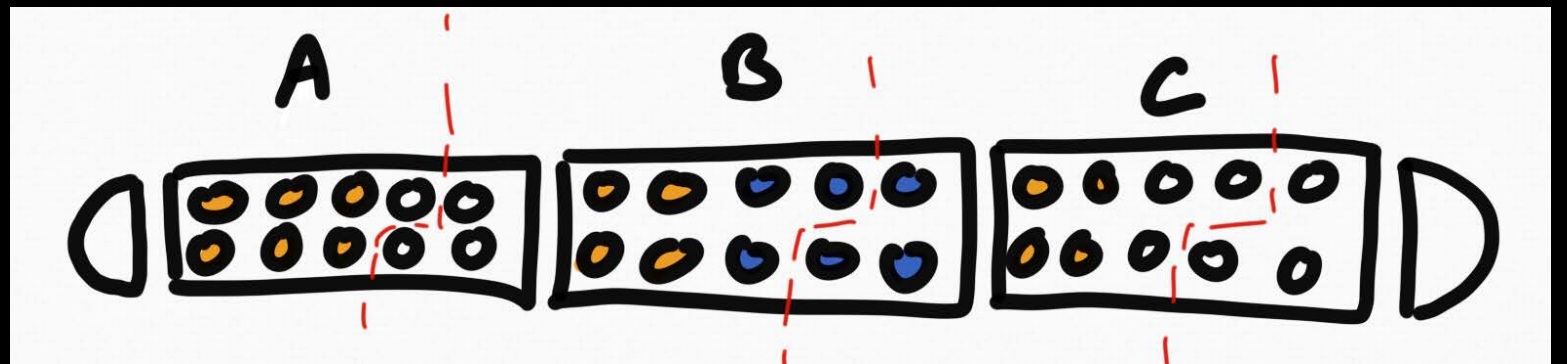
I WANT 6 SEATS

TRAIN-TRAIN

## TEST #5 - NON IDEAL CASE



I WANT 6 SEATS



TRAIN-TRAIN

MISSION ACCOMPLISHED ;-)

TRAIN-TRAIN

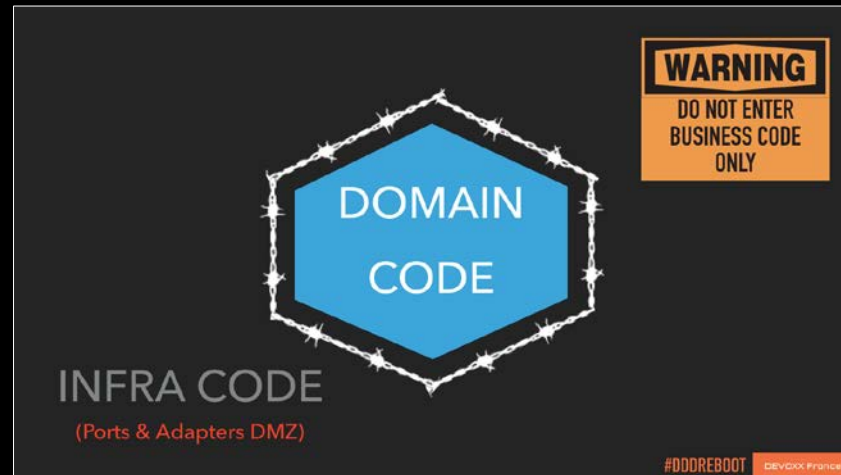
15 MINUTES PAUSE

NEXT: HEXAGONAL ARCHITECTURE

TRAIN-TRAIN

EPISODE3:

# HEXAGONAL ARCHITECTURE FTW!



HEXAGONAL ARCHITECTURE

HEXAGONAL ARCHITECTURE?

HEXAGONAL ARCHITECTURE

ALISTAIR COCKBURN





MANY BENEFITS

# HEXAGONAL ARCHITECTURE

EMBRACE CHANGE  
PROTECT YOUR DOMAIN CODE  
ENFORCE TESTABILITY

MANY BENEFITS

# HEXAGONAL ARCHITECTURE

EMBRACE CHANGE

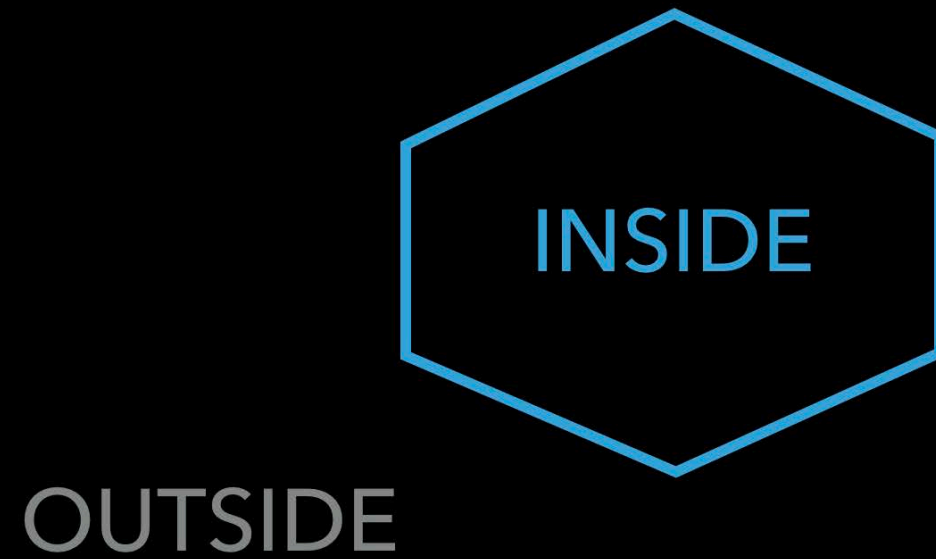
TODAY >> PROTECT YOUR DOMAIN CODE << TODAY

ENFORCE TESTABILITY

# HEXAGONAL ARCHITECTURE

HOW?

# HEXAGONAL ARCHITECTURE



# HEXAGONAL ARCHITECTURE



INFRA

(HTTP, Db, MoM,...)

# HEXAGONAL ARCHITECTURE



INFRA CODE

(Adapters DMZ)

# HEXAGONAL ARCHITECTURE

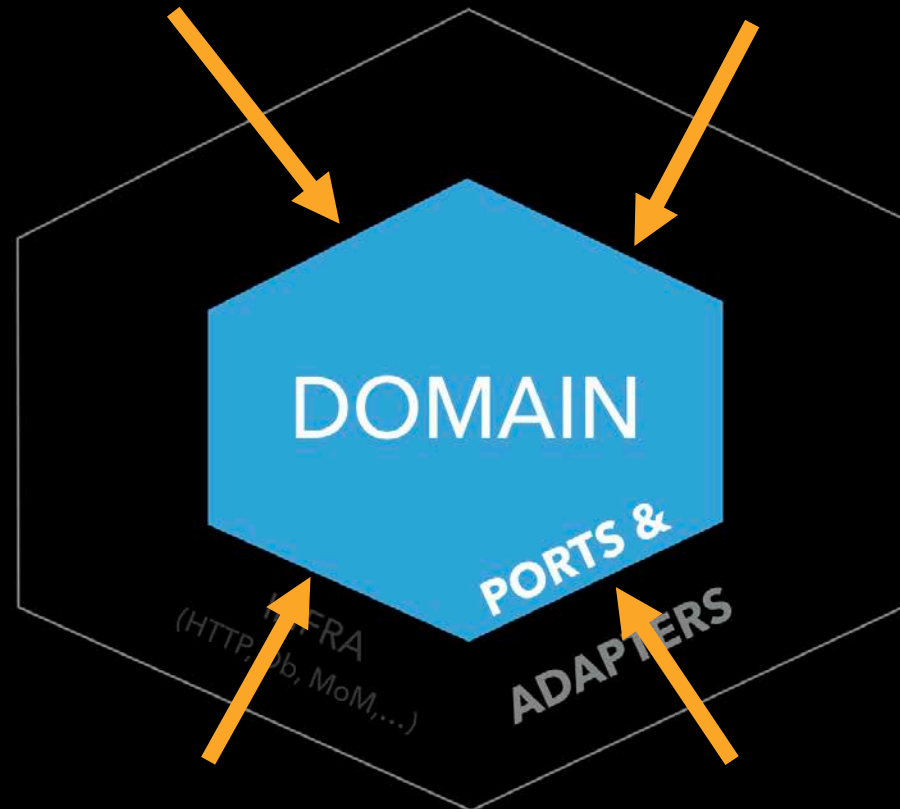


INFRA CODE

>> playground <<

## HEXAGONAL ARCHITECTURE

DEPENDENCIES  
ALWAYS  
TOWARD  
THE INSIDE

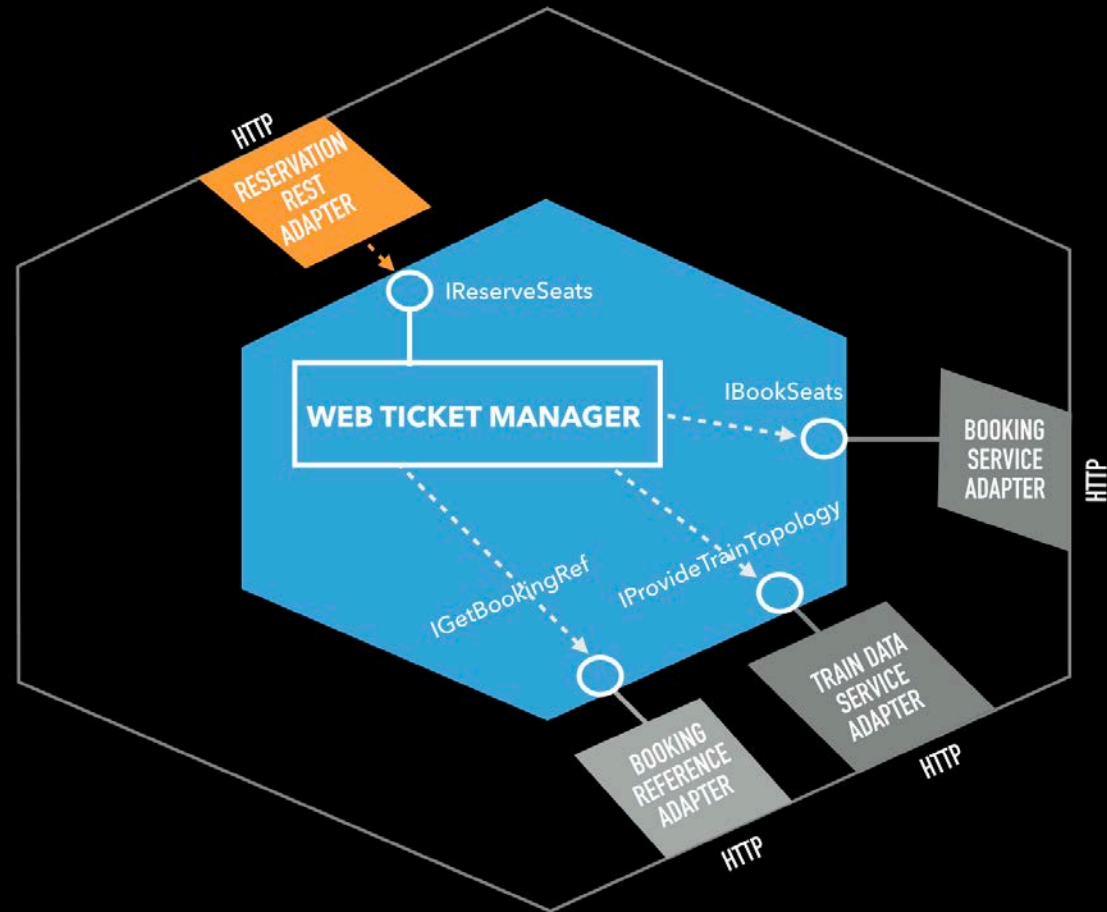


~~DIP~~ PRINCIPLE  
"CONFIGURABLE  
DEPENDENCIES"



## HEXAGONAL ARCHITECTURE

○  
PORTS  
IN THE  
DOMAIN



ADAPTERS  
IN THE  
INFRA

## HEXAGONAL ARCHITECTURE

PORTS AND ADAPTERS

TO GO IN

TO GO OUT

## HEXAGONAL ARCHITECTURE

# A 3 STEPS INITIALIZATION

THE "KEY RING" METAPHOR

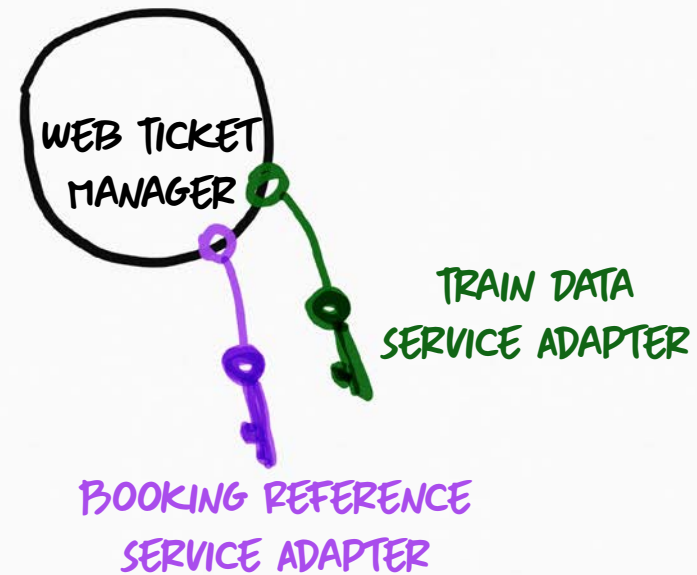
## HEXAGONAL ARCHITECTURE

1. INSTANTIATE THE  
"I NEED TO GO  
OUT" ADAPTERS



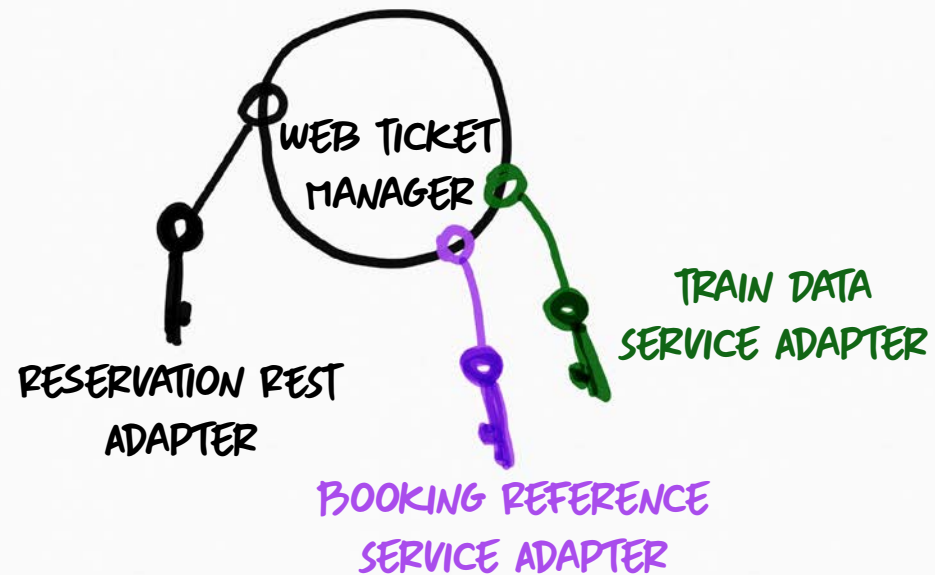
## HEXAGONAL ARCHITECTURE

### 2. INSTANTIATE THE HEXAGON



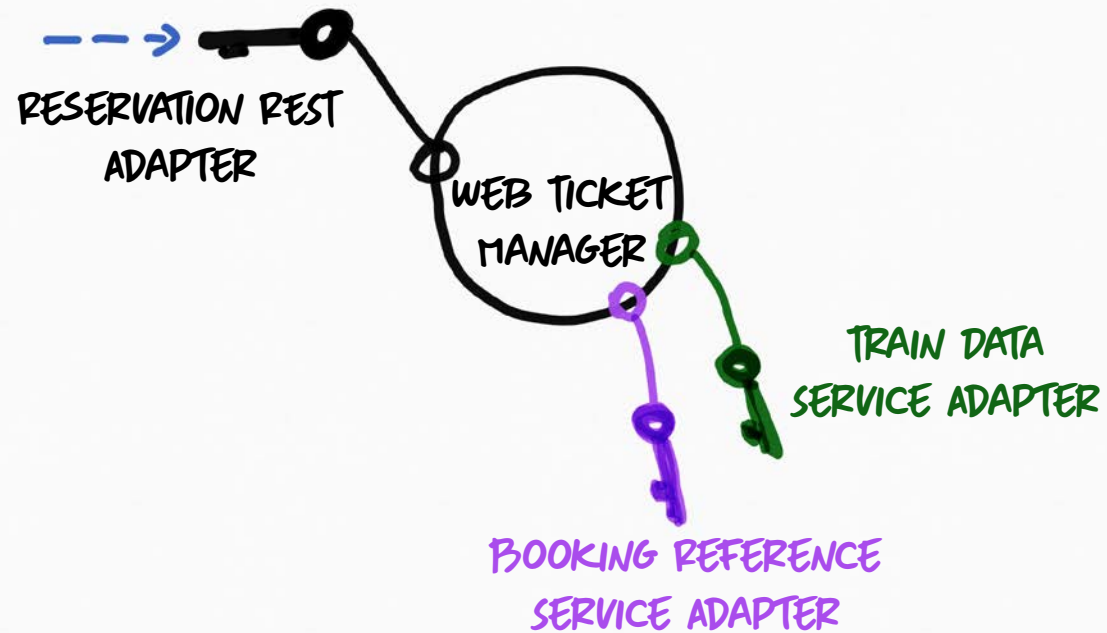
## HEXAGONAL ARCHITECTURE

3. INSTANTIATE THE  
"I NEED TO ENTER"  
ADAPTERS



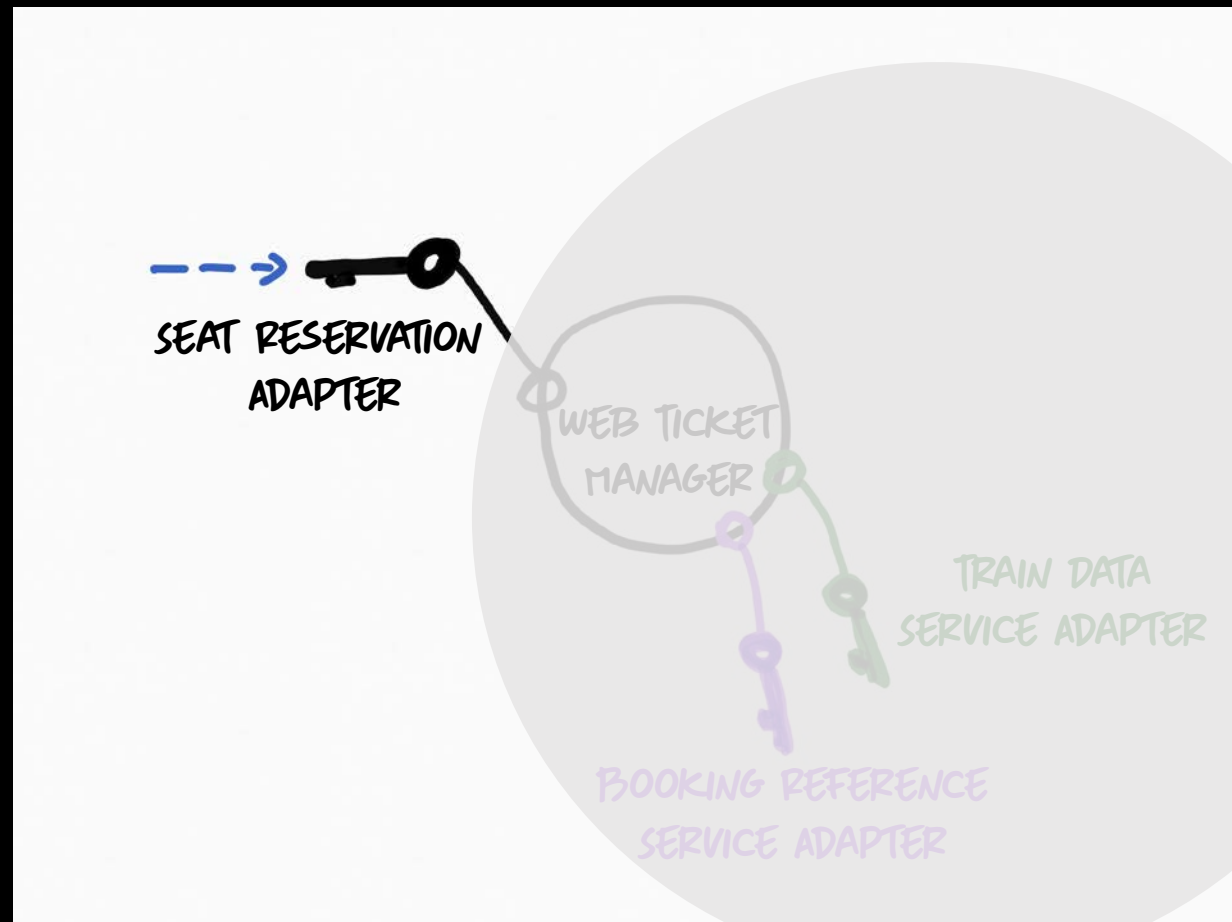
## HEXAGONAL ARCHITECTURE

ALL YOU KEEP IN  
"YOUR HAND"



# HEXAGONAL ARCHITECTURE

ALL YOU KEEP IN  
"YOUR HAND"





## HEXAGONAL ARCHITECTURE

TO PUT IT DIFFERENTLY...

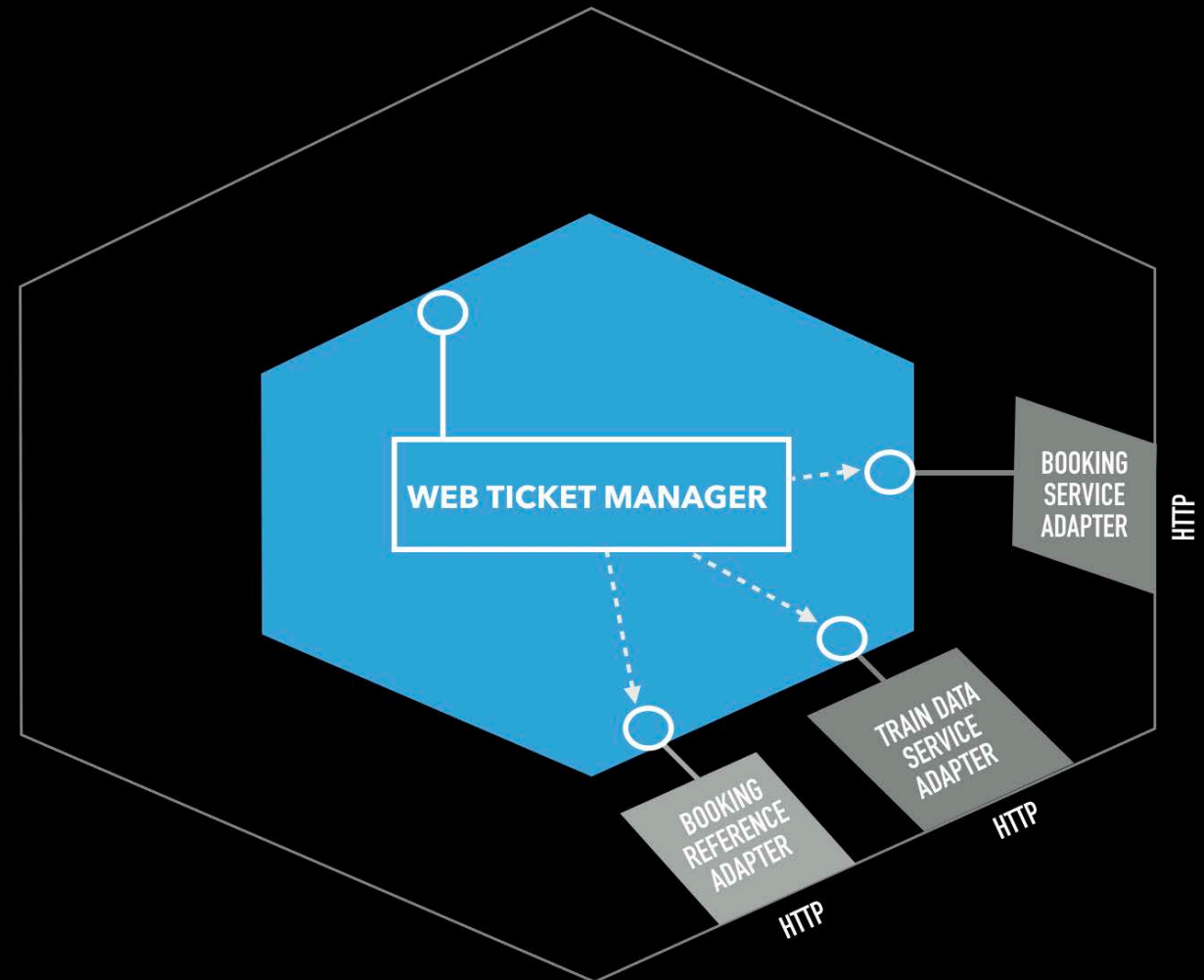
## A 3 STEPS INITIALIZATION

1. INSTANTIATE THE  
"I NEED TO GO  
OUT" ADAPTERS



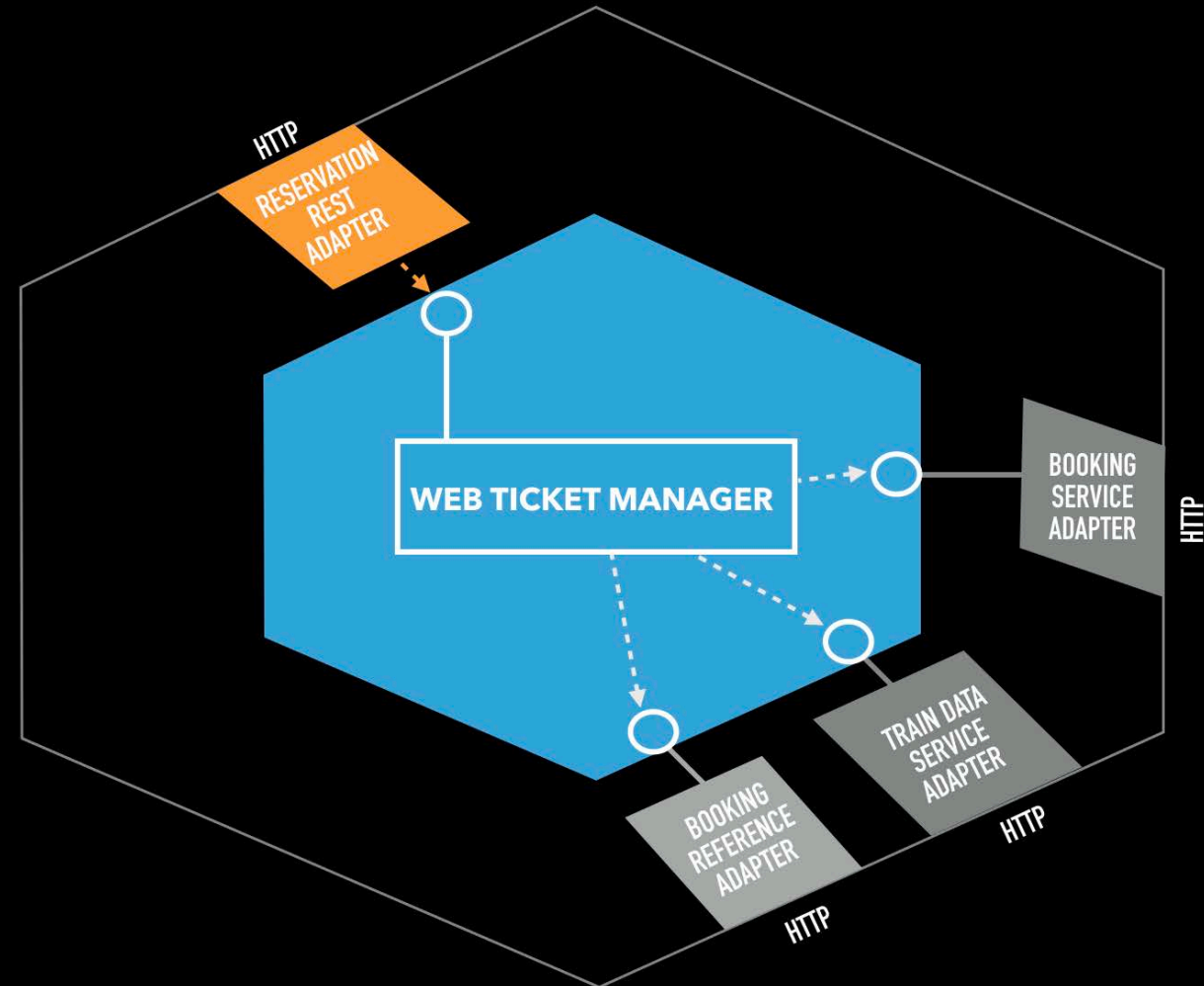
# HEXAGONAL ARCHITECTURE

## 2. INSTANTIATE THE HEXAGON



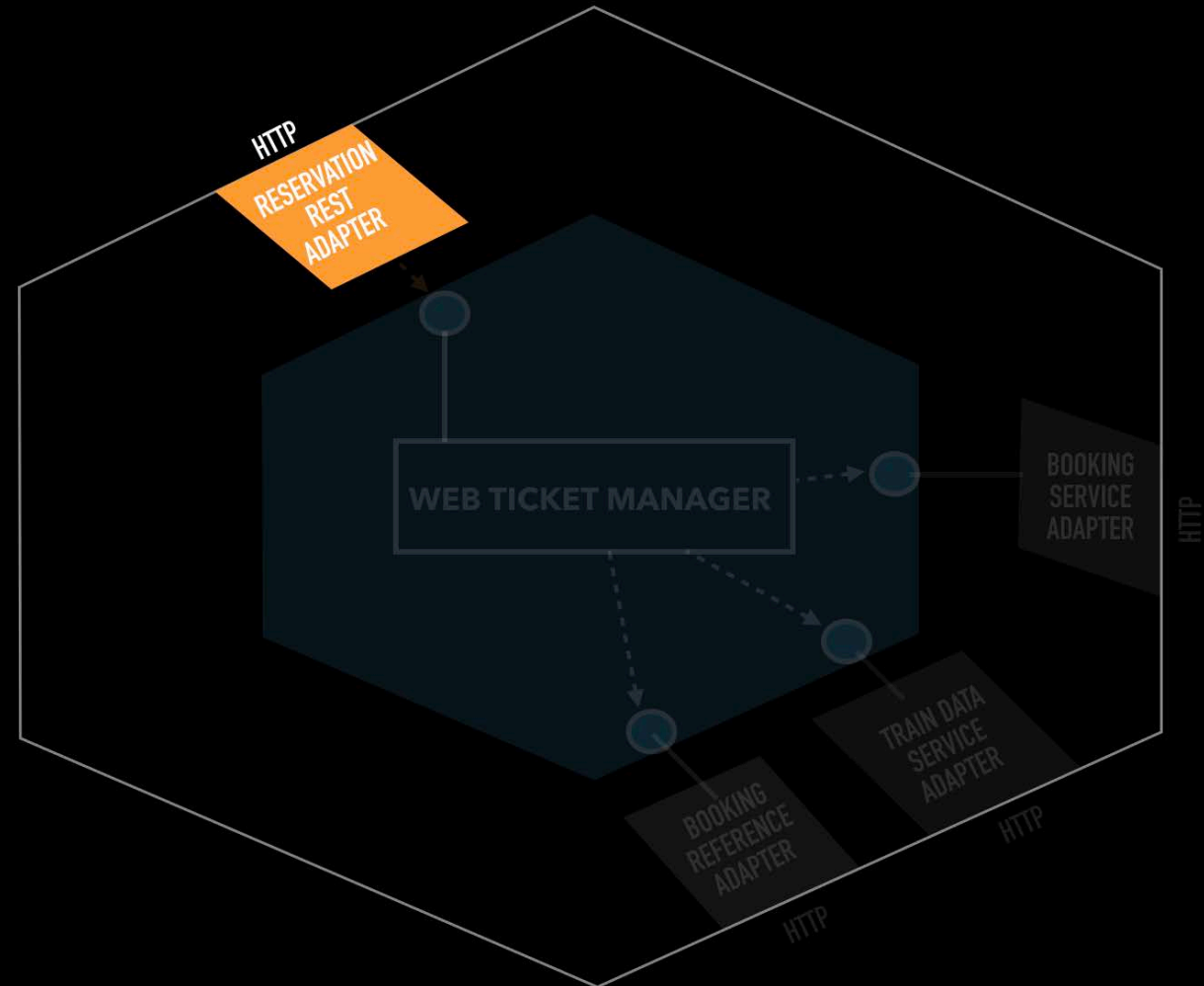
# HEXAGONAL ARCHITECTURE

3. INSTANTIATE THE  
"I NEED TO ENTER"  
ADAPTERS



# HEXAGONAL ARCHITECTURE

ALL YOU KEEP IN  
"YOUR HAND"



HEXAGONAL ARCHITECTURE

LET'S DO IT!



WRAP-UP



# FROM WEB TICKET MANAGER TO SEAT RESERVATION SERVICE

[illegible]

```

1  #!/usr/bin/perl -w
2
3  use strict;
4
5  use Data::Dumper;
6
7  my $url = "http://www.example.com";
8
9  my $response = get($url);
10
11  my $html = $response->content;
12
13  my $doc = HTML::Dom->new($html);
14
15  my $title = $doc->title;
16
17  print "Title: $title\n";
18
19  my $body = $doc->body;
20
21  my $p = $body->p;
22
23  my $text = $p->text;
24
25  print "Text: $text\n";
26
27  my $a = $body->a;
28
29  my $href = $a->href;
30
31  print "Href: $href\n";
32
33  my $img = $body->img;
34
35  my $src = $img->src;
36
37  print "Src: $src\n";
38
39  my $div = $body->div;
40
41  my $class = $div->class;
42
43  print "Class: $class\n";
44
45  my $div2 = $body->div2;
46
47  my $id = $div2->id;
48
49  print "Id: $id\n";
50
51  my $div3 = $body->div3;
52
53  my $style = $div3->style;
54
55  print "Style: $style\n";
56
57  my $div4 = $body->div4;
58
59  my $value = $div4->value;
60
61  print "Value: $value\n";
62
63  my $div5 = $body->div5;
64
65  my $label = $div5->label;
66
67  print "Label: $label\n";
68
69  my $div6 = $body->div6;
70
71  my $checked = $div6->checked;
72
73  print "Checked: $checked\n";
74
75  my $div7 = $body->div7;
76
77  my $disabled = $div7->disabled;
78
79  print "Disabled: $disabled\n";
80
81  my $div8 = $body->div8;
82
83  my $readonly = $div8->readonly;
84
85  print "Readonly: $readonly\n";
86
87  my $div9 = $body->div9;
88
89  my $type = $div9->type;
90
91  print "Type: $type\n";
92
93  my $div10 = $body->div10;
94
95  my $name = $div10->name;
96
97  print "Name: $name\n";
98
99  my $div11 = $body->div11;
100
101  my $value2 = $div11->value;
102
103  print "Value2: $value2\n";
104
105  my $div12 = $body->div12;
106
107  my $label2 = $div12->label;
108
109  print "Label2: $label2\n";
110
111  my $div13 = $body->div13;
112
113  my $checked2 = $div13->checked;
114
115  print "Checked2: $checked2\n";
116
117  my $div14 = $body->div14;
118
119  my $disabled2 = $div14->disabled;
120
121  print "Disabled2: $disabled2\n";
122
123  my $div15 = $body->div15;
124
125  my $readonly2 = $div15->readonly;
126
127  print "Readonly2: $readonly2\n";
128
129  my $div16 = $body->div16;
130
131  my $type2 = $div16->type;
132
133  print "Type2: $type2\n";
134
135  my $div17 = $body->div17;
136
137  my $name2 = $div17->name;
138
139  print "Name2: $name2\n";
140
141  my $div18 = $body->div18;
142
143  my $value3 = $div18->value;
144
145  print "Value3: $value3\n";
146
147  my $div19 = $body->div19;
148
149  my $label3 = $div19->label;
150
151  print "Label3: $label3\n";
152
153  my $div20 = $body->div20;
154
155  my $checked3 = $div20->checked;
156
157  print "Checked3: $checked3\n";
158
159  my $div21 = $body->div21;
160
161  my $disabled3 = $div21->disabled;
162
163  print "Disabled3: $disabled3\n";
164
165  my $div22 = $body->div22;
166
167  my $readonly3 = $div22->readonly;
168
169  print "Readonly3: $readonly3\n";
170
171  my $div23 = $body->div23;
172
173  my $type3 = $div23->type;
174
175  print "Type3: $type3\n";
176
177  my $div24 = $body->div24;
178
179  my $name3 = $div24->name;
180
181  print "Name3: $name3\n";
182
183  my $div25 = $body->div25;
184
185  my $value4 = $div25->value;
186
187  print "Value4: $value4\n";
188
189  my $div26 = $body->div26;
190
191  my $label4 = $div26->label;
192
193  print "Label4: $label4\n";
194
195  my $div27 = $body->div27;
196
197  my $checked4 = $div27->checked;
198
199  print "Checked4: $checked4\n";
200
201  my $div28 = $body->div28;
202
203  my $disabled4 = $div28->disabled;
204
205  print "Disabled4: $disabled4\n";
206
207  my $div29 = $body->div29;
208
209  my $readonly4 = $div29->readonly;
210
211  print "Readonly4: $readonly4\n";
212
213  my $div30 = $body->div30;
214
215  my $type4 = $div30->type;
216
217  print "Type4: $type4\n";
218
219  my $div31 = $body->div31;
220
221  my $name4 = $div31->name;
222
223  print "Name4: $name4\n";
224
225  my $div32 = $body->div32;
226
227  my $value5 = $div32->value;
228
229  print "Value5: $value5\n";
230
231  my $div33 = $body->div33;
232
233  my $label5 = $div33->label;
234
235  print "Label5: $label5\n";
236
237  my $div34 = $body->div34;
238
239  my $checked5 = $div34->checked;
240
241  print "Checked5: $checked5\n";
242
243  my $div35 = $body->div35;
244
245  my $disabled5 = $div35->disabled;
246
247  print "Disabled5: $disabled5\n";
248
249  my $div36 = $body->div36;
250
251  my $readonly5 = $div36->readonly;
252
253  print "Readonly5: $readonly5\n";
254
255  my $div37 = $body->div37;
256
257  my $type5 = $div37->type;
258
259  print "Type5: $type5\n";
260
261  my $div38 = $body->div38;
262
263  my $name5 = $div38->name;
264
265  print "Name5: $name5\n";
266
267  my $div39 = $body->div39;
268
269  my $value6 = $div39->value;
270
271  print "Value6: $value6\n";
272
273  my $div40 = $body->div40;
274
275  my $label6 = $div40->label;
276
277  print "Label6: $label6\n";
278
279  my $div41 = $body->div41;
280
281  my $checked6 = $div41->checked;
282
283  print "Checked6: $checked6\n";
284
285  my $div42 = $body->div42;
286
287  my $disabled6 = $div42->disabled;
288
289  print "Disabled6: $disabled6\n";
290
291  my $div43 = $body->div43;
292
293  my $readonly6 = $div43->readonly;
294
295  print "Readonly6: $readonly6\n";
296
297  my $div44 = $body->div44;
298
299  my $type6 = $div44->type;
300
301  print "Type6: $type6\n";
302
303  my $div45 = $body->div45;
304
305  my $name6 = $div45->name;
306
307  print "Name6: $name6\n";
308
309  my $div46 = $body->div46;
310
311  my $value7 = $div46->value;
312
313  print "Value7: $value7\n";
314
315  my $div47 = $body->div47;
316
317  my $label7 = $div47->label;
318
319  print "Label7: $label7\n";
320
321  my $div48 = $body->div48;
322
323  my $checked7 = $div48->checked;
324
325  print "Checked7: $checked7\n";
326
327  my $div49 = $body->div49;
328
329  my $disabled7 = $div49->disabled;
330
331  print "Disabled7: $disabled7\n";
332
333  my $div50 = $body->div50;
334
335  my $readonly7 = $div50->readonly;
336
337  print "Readonly7: $readonly7\n";
338
339  my $div51 = $body->div51;
340
341  my $type7 = $div51->type;
342
343  print "Type7: $type7\n";
344
345  my $div52 = $body->div52;
346
347  my $name7 = $div52->name;
348
349  print "Name7: $name7\n";
350
351  my $div53 = $body->div53;
352
353  my $value8 = $div53->value;
354
355  print "Value8: $value8\n";
356
357  my $div54 = $body->div54;
358
359  my $label8 = $div54->label;
360
361  print "Label8: $label8\n";
362
363  my $div55 = $body->div55;
364
365  my $checked8 = $div55->checked;
366
367  print "Checked8: $checked8\n";
368
369  my $div56 = $body->div56;
370
371  my $disabled8 = $div56->disabled;
372
373  print "Disabled8: $disabled8\n";
374
375  my $div57 = $body->div57;
376
377  my $readonly8 = $div57->readonly;
378
379  print "Readonly8: $readonly8\n";
380
381  my $div58 = $body->div58;
382
383  my $type8 = $div58->type;
384
385  print "Type8: $type8\n";
386
387  my $div59 = $body->div59;
388
389  my $name8 = $div59->name;
390
391  print "Name8: $name8\n";
392
393  my $div60 = $body
```

# AFTER

BEFORE

## FROM WEB TICKET MANAGER TO SEAT RESERVATION SERVICE

```
public async Task<Reservation> Reserve(string trainId, int seatsRequestedCount)
{
    Train train = await _trainDataService.GetTrain(trainId);

    if (train.DoesNotExceedTrainCapacityLimit(seatsRequestedCount))
    {
        ReservationAttempt reservationAttempt = train.BuildReservationAttempt(trainId, seatsRequestedCount);

        if (reservationAttempt.IsFulfilled)
        {
            var bookingRef = await _bookingReferenceService.GetBookingReference();

            reservationAttempt.AssignBookingReference(bookingRef);

            await _trainDataService.BookSeats(reservationAttempt);

            Reservation reservation = reservationAttempt.Confirm();

            return reservation;
        }
    }

    return new ReservationFailed(trainId, seatsRequestedCount);
}
```

AFTER

## UBIQUITOUS LANGUAGE IN DA HOUSE

```
public async Task<Reservation> Reserve(string trainId, int seatsRequestedCount)
{
    Train train = await _trainDataService.GetTrain(trainId);

    if (train.DoesNotExceedTrainCapacityLimit(seatsRequestedCount))
    {
        ReservationAttempt reservationAttempt = train.BuildReservationAttempt(trainId, seatsRequestedCount);

        if (reservationAttempt.IsFulfilled)
        {
            var bookingRef = await _bookingReferenceService.GetBookingReference();

            reservationAttempt.AssignBookingReference(bookingRef);

            await _trainDataService.BookSeats(reservationAttempt);

            Reservation reservation = reservationAttempt.Confirm();

            return reservation;
        }
    }

    return new ReservationFailed(trainId, seatsRequestedCount);
}
```

## UBIQUITOUS LANGUAGE IN DA HOUSE

```
public async Task<Reservation> Reserve(string trainId, int seatsRequestedCount)
{
    Train train = await _trainDataService.GetTrain(trainId);

    if (train.DoesNotExceedTrainCapacityLimit(seatsRequestedCount))
    {
        ReservationAttempt reservationAttempt = train.BuildReservationAttempt(trainId, seatsRequestedCount);

        if (reservationAttempt.IsFulfilled)
        {
            var bookingRef = await _bookingReferenceService.GetBookingReference();

            reservationAttempt.AssignBookingReference(bookingRef);

            await _trainDataService.BookSeats(reservationAttempt);

            Reservation reservation = reservationAttempt.Confirm();

            return reservation;
        }
    }

    return new ReservationFailed(trainId, seatsRequestedCount);
}
```



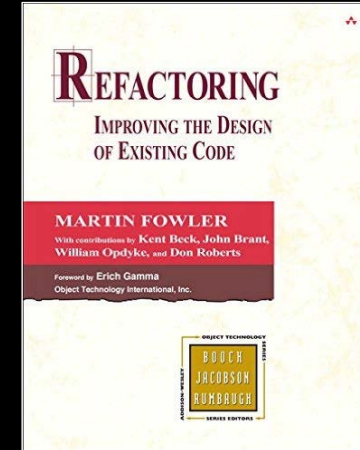
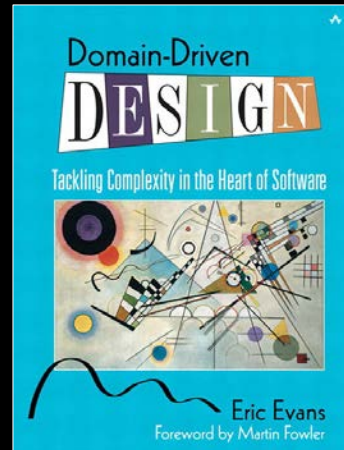
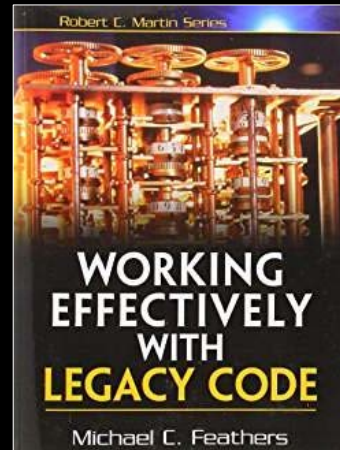
SOON!

FINAL WRAP-UP

## WRAP-UP

FOCUS ON, HIGHLIGHT AND  
PROTECT THE BUSINESS VALUE IN  
YOUR CODE!

# AND THE WINNERS ARE ...





LET'S CONTINUE TO CHAT ON TWITTER



THAT'S ALL FOLKS,  
THANKS!