

**ISSUE WITH
TRACKING?
FAIL THAT BUILD!**



LAUNCH DAY

***SHIPPED IT!
GO TEAM!***

***DID WE EVER FINISH
THE TRACKING?***



TWO WEEKS LATER

***10 MILLION EVENTS
NAMED 'REPLACEME.'
NOTHING ELSE.***

A black and white photograph of a business meeting. On the left, a woman with curly hair is looking at a laptop. On the right, a man with glasses and a beard is looking at a tablet. In the background, other people are partially visible. A large purple lightning bolt graphic is superimposed over the center of the image, pointing downwards.

THE BIG DIVIDE

***CUE DATA
BREAKING STUFF
IN TAG MANAGER***

***CUE PRODUCT
ROLLING IT OUT
ANYWAY***



STEVE COPPIN-SMITH

**HEAD OF ENGINEERING
@ SNOWFLOW ANALYTICS**

SNOWFLOW

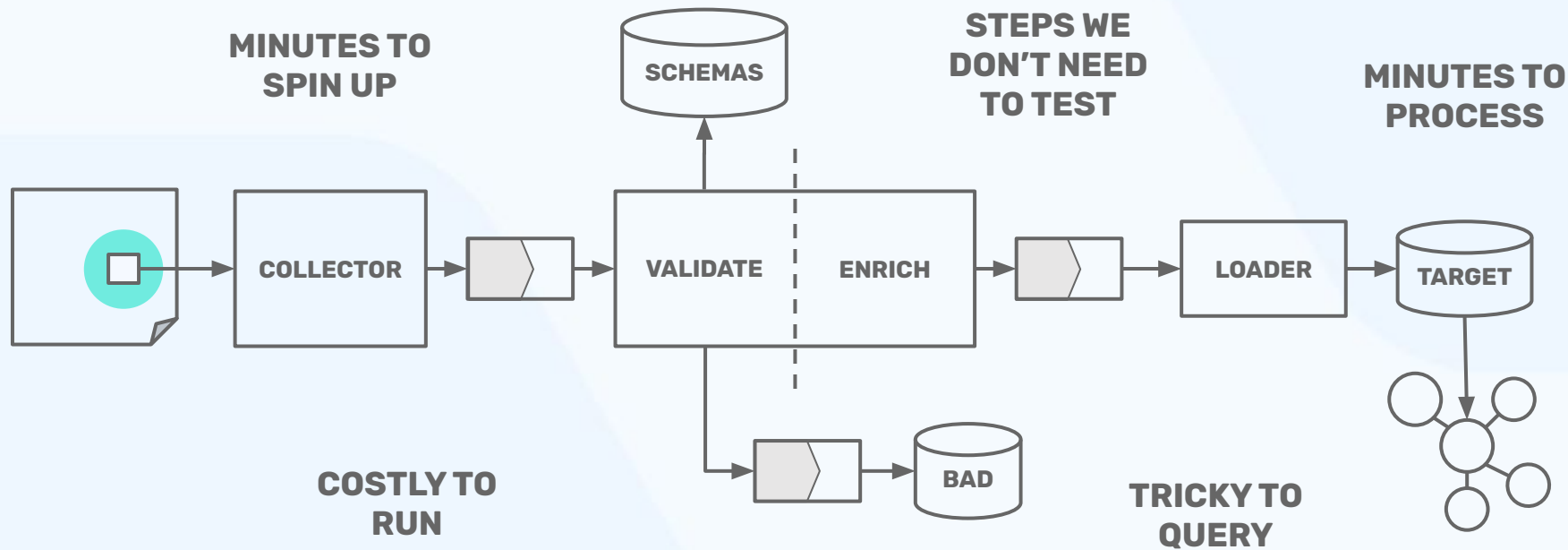
@COPPIN_SMITH

**TRACKING IS AS
IMPORTANT AS THE
FEATURE YOU'RE
SHIPPING.
SO TEST IT.**

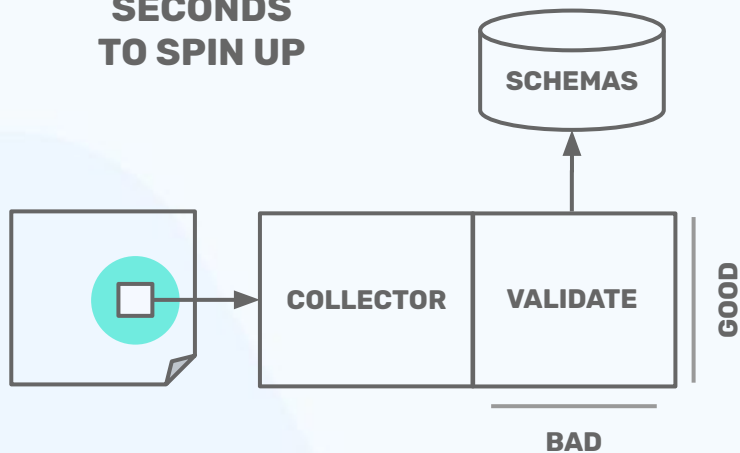
DATA QUALITY

SCHEMAS + BAD ROWS + QA PIPELINES

DATA ACCURACY



**SECONDS
TO SPIN UP**

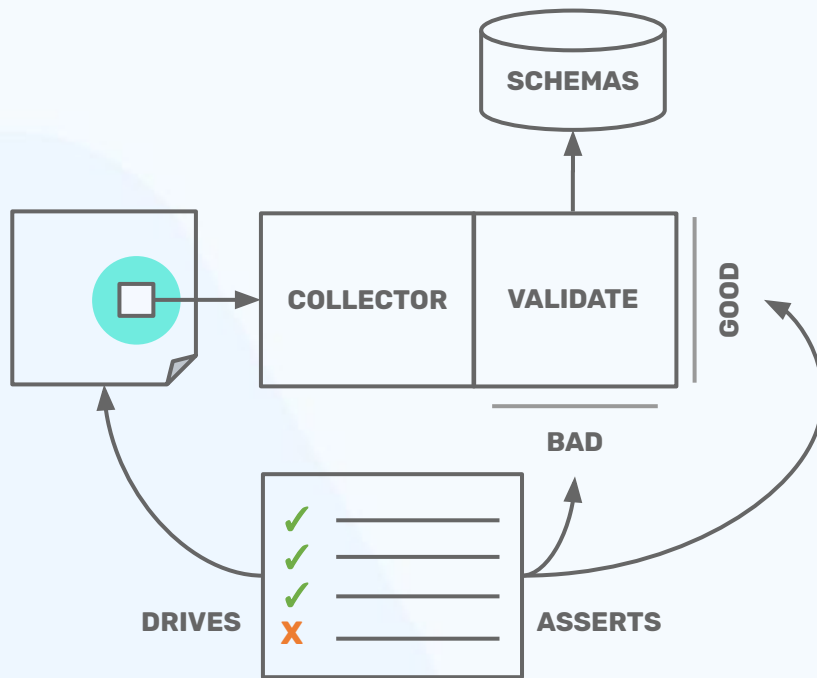


**SPLIT
SECONDS
TO ARRIVE**

**SIMPLE TO
QUERY**

**MINIMAL
COSTS**

**A SINGLE
EXECUTABLE**





NIGHTWATCH.JS

- ✓ **CROSS BROWSER**
- ✓ **EASY TO EXTEND***
- ✓ **CUTEST LOGO**

```
'Add multiple products to cart' : function (browser) {  
  browser  
    .click('//a/span[text()="Shirt"]/..')  
    .click('//*[@id="quantity"]/option[2]')  
    .click('//button[@id="addToCart"]')  
    .assert.containsText('//*[@id="cart"]', 'Shirt')  
}
```

\$ ✓ Testing if element contains text 'Shirt'

```
'Add multiple products to cart' : function (browser) {  
  browser  
    .click('//a/span[text()="Shirt"]/..')  
    .click('//*[@id="quantity"]/option[2]')  
    .click('//button[@id="addToCart"]')  
    .assert.containsText('//*[@id="cart"]', 'Shirt')  
    .assert.noBadEvents()  
}
```

\$ **x** Expected 0 bad events, 10 found.

```
'Add multiple products to cart' : function (browser) {  
  browser  
    .click('//a/span[text()="Shirt"]/..')  
    .click('//*[@id="quantity"]/option[2]')  
    .click('//button[@id="addToCart"]')  
    .assert.containsText('//*[@id="cart"]', 'Shirt')  
    .assert.noBadEvents()  
    .assert.eventWithProperty({  
      'schema': '.../cart_event/jsonschema/1-0-0',  
    }, 2)  
}
```

\$ **x** Expected 2 events with properties, 1 found.

```
'Add multiple products to cart' : function (browser) {  
  browser  
    .click('//a/span[text()="Shirt"]/..')  
    .click('//*[@id="quantity"]/option[2]')  
    .click('//button[@id="addToCart"]')  
    .assert.containsText('//*[@id="cart"]', 'Shirt')  
    .assert.noBadEvents()  
    .assert.eventWithProperty({  
      'schema': '.../cart_event/jsonschema/1-0-0',  
      'values': {'quantity': 2},  
    }, 1)  
}
```

\$  Expected 1 event with properties, 1 found.

```

'Add multiple products to cart' : function (browser) {
  browser
    .click('//a/span[text()="Shirt"]/..')
    .click('//*[@id="quantity"]/option[2]')
    .click('//button[@id="addToCart"]')
    .assert.containsText('//*[@id="cart"]', 'Shirt')
    .assert.noBadEvents()
    .assert.eventWithProperty({
      'schema': '.../cart_event/jsonschema/1-0-0',
      'values': {'quantity': 2},
      'contexts': [{
        'schema': '.../product_entity/jsonschema/1-0-0',
        'values': {'name': 'Shirt', 'color': 'White'}
      }]
    }, 1)
}

```

**THIS GOES BEYOND
BUILDING ASSURANCE
AROUND THE FEATURE
YOU ARE SHIPPING**

YOUR TEST SUITE

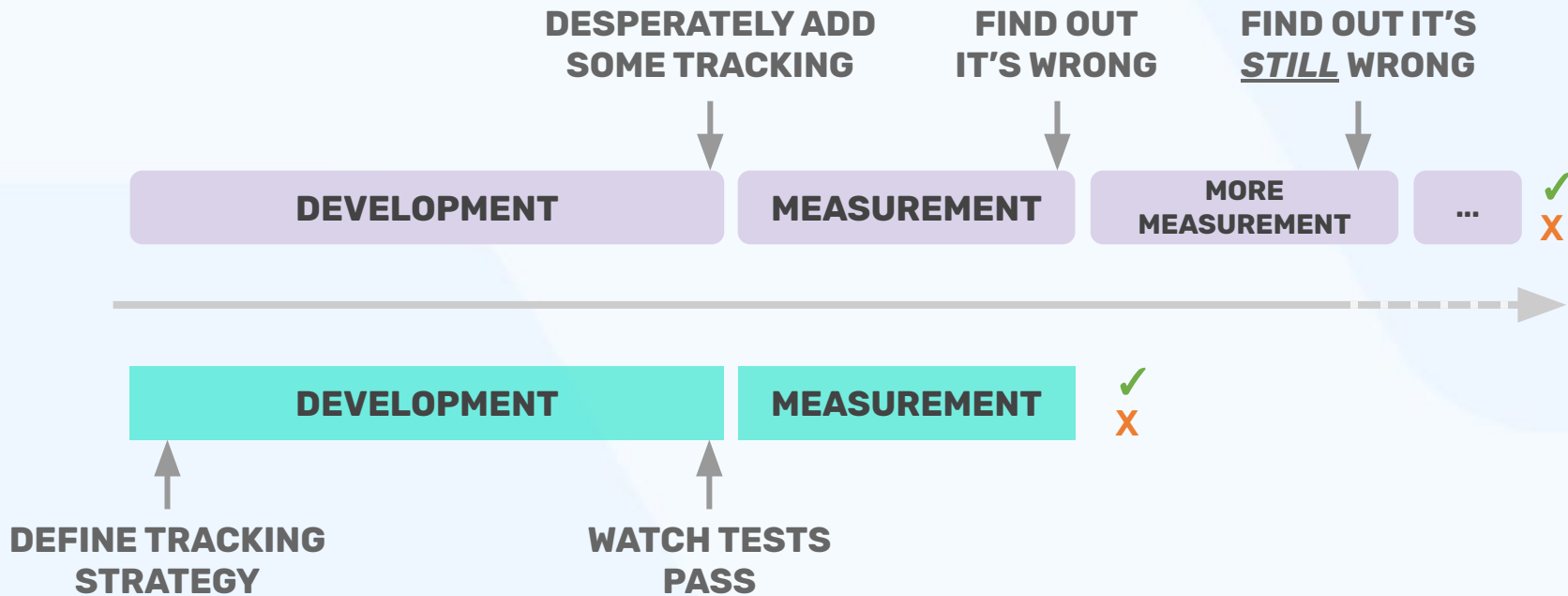
X

✓

✓

✓

✓



A black and white photograph of a business meeting. Four people are gathered around a wooden table. A woman with curly hair stands on the left, gesturing towards a man with glasses and a beard who is seated and holding a document with a line graph. A woman with blonde hair is seated in the foreground, facing away from the camera. A man with a short haircut is seated on the right, looking towards the group. A speech bubble is overlaid on the image, containing the text "BEST. DATA. EVER." in a stylized font.

***BEST. DATA.
EVER.***

LINTING **OK**
UNIT TESTS **OK**
FEATURES **OK**
TRACKING **NOT OK**
BUNDLING
DEPLOYING

**ISSUE WITH
TRACKING?
FAIL THAT BUILD!**