

# iowa code camp

11.01.2014

# TESTS

**WHY TESTS ARE MORE THAN  
CODE COVERAGE.**

# WHY TESTS ARE MORE THAN CODE COVERAGE. the business value behind a test

**dj daugherty**

dj daugherty

**[dj@pillartechology.com](mailto:dj@pillartechology.com)**

dj daugherty

[dj@pillartechology.com](mailto:dj@pillartechology.com)

**@ddaugher**

point air

**pointair**  
**we are hiring !**

pitair

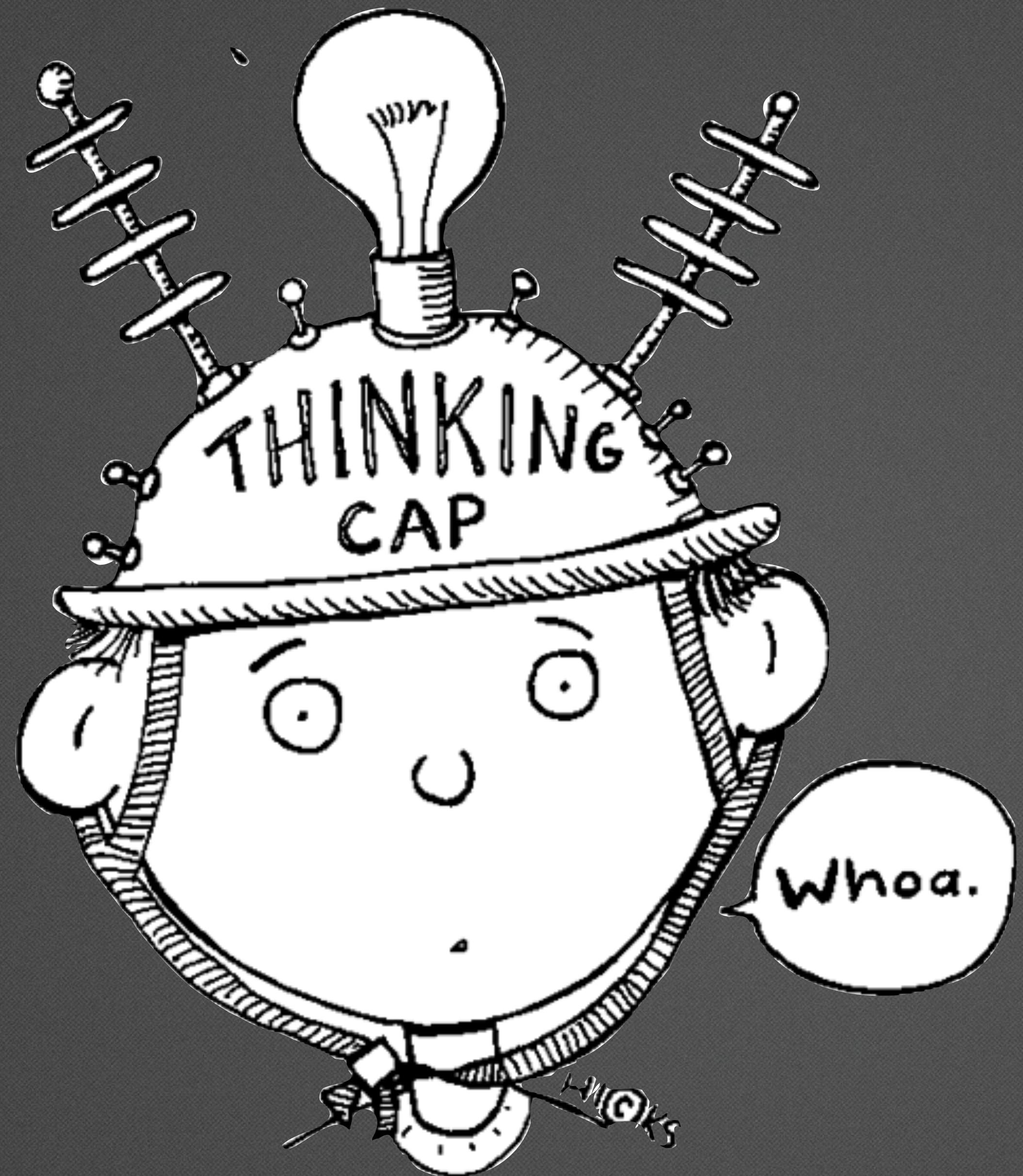


i am developer.

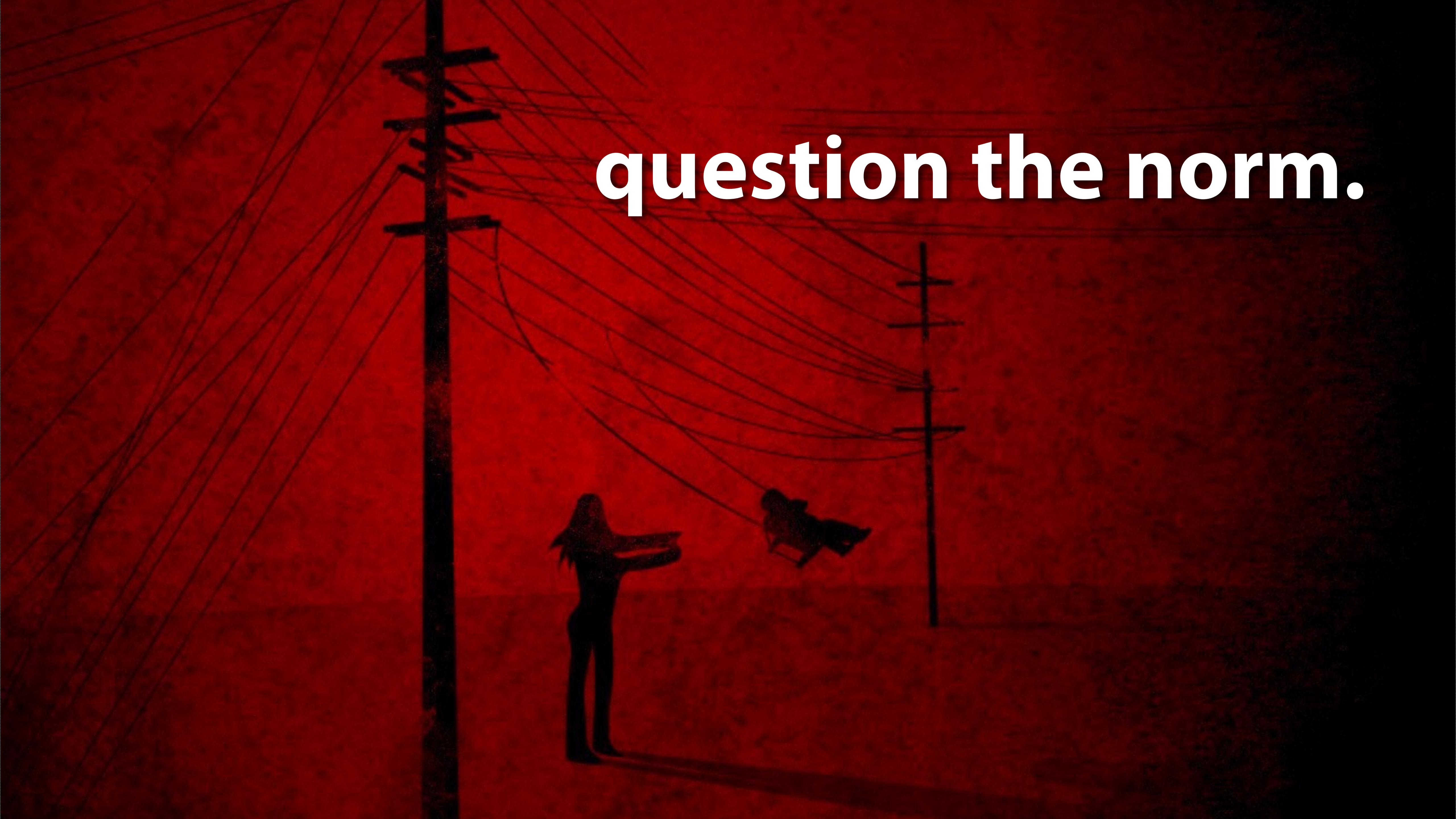








think.

A red-tinted photograph showing two silhouetted figures against a dark background. On the left, a woman stands facing right, her hair flowing. On the right, a man sits in a chair, looking towards the woman. They are positioned beneath a network of power lines supported by two utility poles. The scene has a dramatic, almost theatrical feel due to the color and lighting.

question the norm.

# DISCLOSURE!

---

i struggle with these ideas every day.

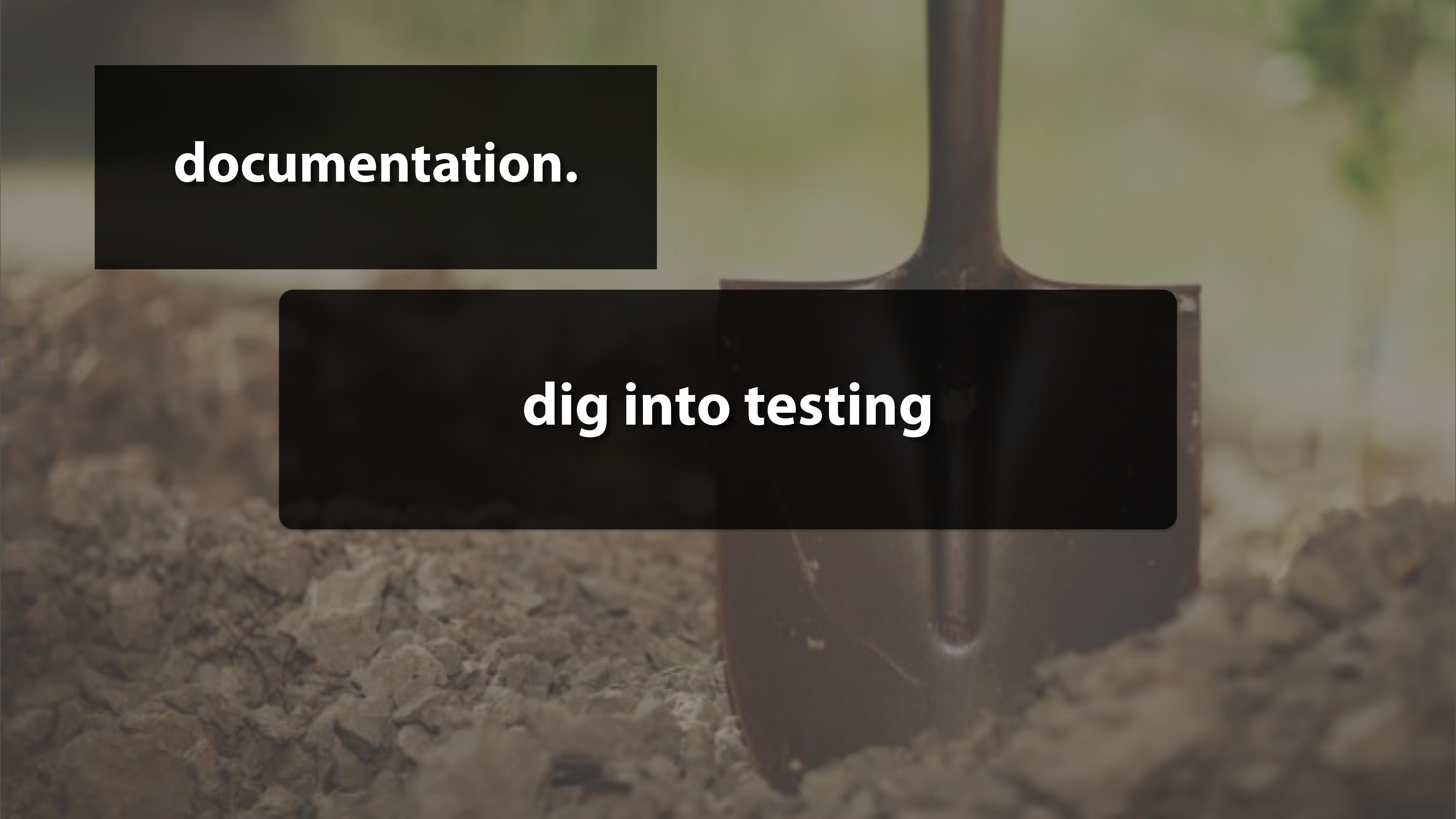
# WHY TESTS ARE MORE THAN CODE COVERAGE. the business value behind a test

**how many developers?**

**how many business?**



dig into testing

A close-up photograph of a dark brown wooden shovel handle and head, partially buried in light brown soil. The background is blurred.

documentation.

dig into testing



A close-up photograph of a person's arm and hand. The person is wearing a dark long-sleeved shirt. Their hand is positioned as if they are about to grip or are gripping a metal shovel handle. The background is blurred, showing some greenery and earth.

documentation.

dig into testing

engage developers into  
business process.

A person wearing a white hard hat and a light blue safety vest over a dark shirt is looking down at a tablet device held in their hands. They are standing outdoors, with a blurred background of green trees and a building.

**documentation.**

**compare requirements  
over time.**

**dig into testing**

**engage developers into  
business process.**

A photograph of a person's hands resting on a laptop keyboard, serving as the background for the slide. The hands are positioned centrally, with the fingers on the keys.

**documentation.**

**compare requirements  
over time.**

**dig into testing**

**engage developers into  
business process.**

**remove fear  
of refactoring.**

**WHAT IS  
CODE COVERAGE?**

Please comment on a proposed amendment regarding undisclosed paid editing.

[ Help with translations! ]

# Code coverage

From Wikipedia, the free encyclopedia

In computer science, **code coverage** is a measure used to describe the degree to which the [source code](#) of a [program](#) is tested by a particular [test suite](#). A program with high code coverage has been more thoroughly tested and has a lower chance of containing [software bugs](#) than a program with low code coverage. Many different metrics can be used to calculate code coverage; some of the most basic are the percent of program [subroutines](#) and the percent of program [statements](#) called during execution of the test suite.

Please comment on a proposed amendment regarding undisclosed paid editing.

[ Help with translations! ]

# Code coverage

From Wikipedia, the free encyclopedia

In computer science, **code coverage** is a measure used to describe the degree to which the [source code](#) of a [program](#) is tested by a particular [test suite](#). A program with high code coverage has been more thoroughly tested and has a lower chance of containing [software bugs](#) than a program with low code coverage. Many different metrics can be used to calculate code coverage; some of the most basic are the percent of program [subroutines](#) and the percent of program [statements](#) called during execution of the test suite.

Please comment on a proposed amendment regarding undisclosed paid editing.

[ Help with translations! ]

## Code coverage

From Wikipedia, the free encyclopedia

In computer science, **code coverage** is a measure used to describe the degree to which the source code of a program is tested by a particular test suite. A program with high code coverage has been more thoroughly tested and has a lower chance of containing software bugs than a program with low code coverage. Many different metrics can be used to calculate code coverage; some of the most basic are the percent of program subroutines and the percent of program statements called during execution of the test suite.

**WHAT IS A  
TEST?**

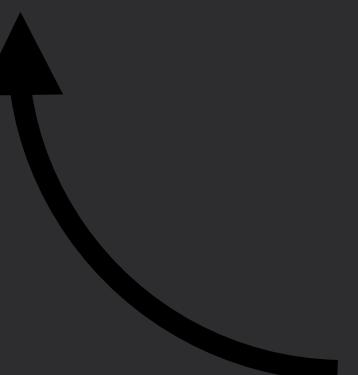
**test<sup>1</sup>**  [test]  [Show IPA](#)

**noun**

1. the means by which the presence, quality, or genuineness of anything is determined; a means of trial.
2. the trial of the quality of something: *to put to the test*.
3. a particular process or method for trying or assessing.
4. a set of questions, problems, or the like, used as a means of evaluating the abilities, aptitudes, skills, or performance of an individual or group; examination.
5. *Psychology* . a set of standardized questions, problems, or tasks designed to elicit responses for use in measuring the traits, capacities, or achievements of an individual.

4. a set of questions, problems, or the like, used as a means of evaluating the abilities, aptitudes, skills, or performance of an individual or group; examination.

4. a set of questions, problems, or the like, used as a means of evaluating the abilities, aptitudes, skills, or performance of ~~an individual or group~~; examination.



**code or an application**

**WHY DO I  
CARE?**

# WHY DO I CARE?

deliver features.  
eliminate defects.  
business value.

# WORKING CODE.

WHY DO I  
CARE.

deliver features.  
eliminate defects.  
business value.

@ A HIGH LEVEL  
**BUSINESS**

@ A HIGH LEVEL  
**BUSINESS**

features that work.  
faster development.

@ A HIGH LEVEL  
**DEVELOPMENT**

**understand business.  
understand codebase.  
refactor without fear.  
commit without fear.**

# **@ A HIGH LEVEL DEVELOPMENT**

**INTERESTINGLY...**

# **INTERESTINGLY...**



## **DEVELOPMENT = BUSINESS**

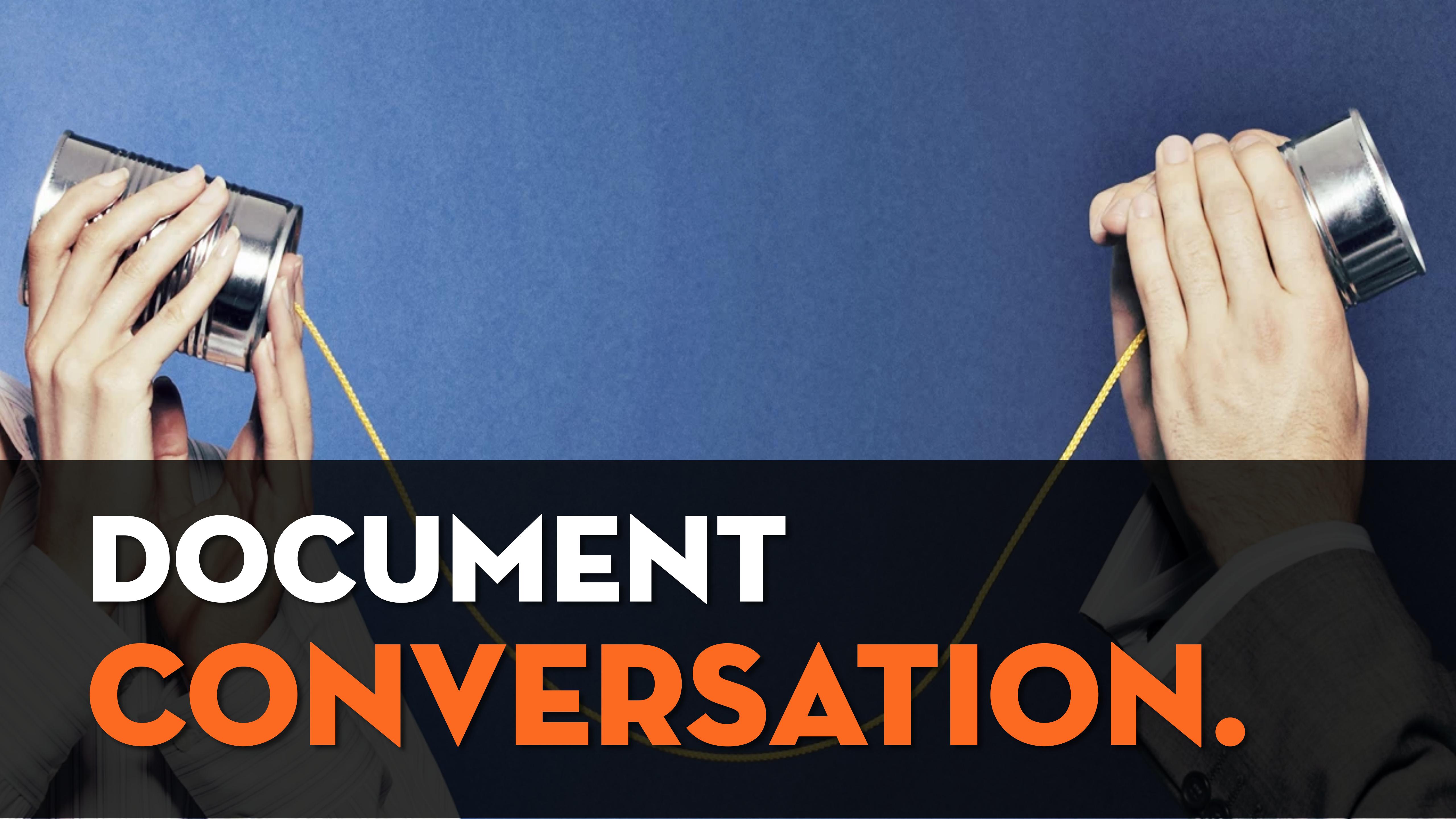
# **INTERESTINGLY...**



## **BUSINESS = DEVELOPMENT**

# BUSINESS VALUE

the  
**BUSINESS VALUE**  
of testing



**DOCUMENT  
CONVERSATION.**



**CLARIFY  
INTENT.**



**REPEATABLE  
SPECIFICATIONS.**

**FIRST  
CONSUMER.**



# EXECUTABLE CODE EXAMPLES.

**CHANGE**

**PROTECT FROM**



**UNDERSTAND CODE  
FASTER.**



MERCILESS  
REFACTORING.



**IMPROVE  
CODE REUSE.**

**GOOD CODE  
BAD CODE.**

**“imperfect tests,  
run frequently, are much  
better than perfect  
tests that are never  
written at all.”**

**- M. Fowler**

# BE CAREFUL



watch  
your  
tests



tests...  
do not create  
**AWESOMENESS!**



poor architecture... is just poor.



bad tests... are just bad.



too vague

too  
implementation  
specific

too granular

too broad

bad tests... are just bad.

A close-up photograph of a person's hand emerging from a dark, textured glove. The hand is holding a cluster of various colored crayons, including red, blue, green, yellow, and purple. The background is a dark, textured surface, possibly leather or cloth.

**TESTING IS AN  
ART.**

**if tests are hard  
to write...  
code may  
need work!**

**ASSET OF THE  
PRODUCT.**

**example**

**STAY AWAKE**

so,...



**TESTING IS NOT  
FAIRY DUST AND  
UNICORN FARTS.**

**JUST  
START.**

**JUST  
START.**

it will suck at first.

**JUST  
START.**

it will suck at first.  
that's ok.

**JUST  
START.**

it will suck at first.  
that's ok.  
have a ritual.

**JUST  
START.**

it will suck at first.  
that's ok.  
have a ritual.  
it is ok to fail.

**JUST  
START.**

it will suck at first.  
that's ok.  
have a ritual.  
it is ok to fail.  
beer helps!

**JUST  
START.**

it will suck at first.  
that's ok.  
have a ritual.  
it is ok to fail.  
beer helps!  
eventually it will suck less.

# THANKS.

dj daugherty  
[dj@pillartechology.com](mailto:dj@pillartechology.com)  
[@ddauher](https://twitter.com/ddaughher)

# Q&A.

dj daugherty  
[dj@pillartechology.com](mailto:dj@pillartechology.com)  
[@ddaugher](https://twitter.com/@ddaugher)