

++
24

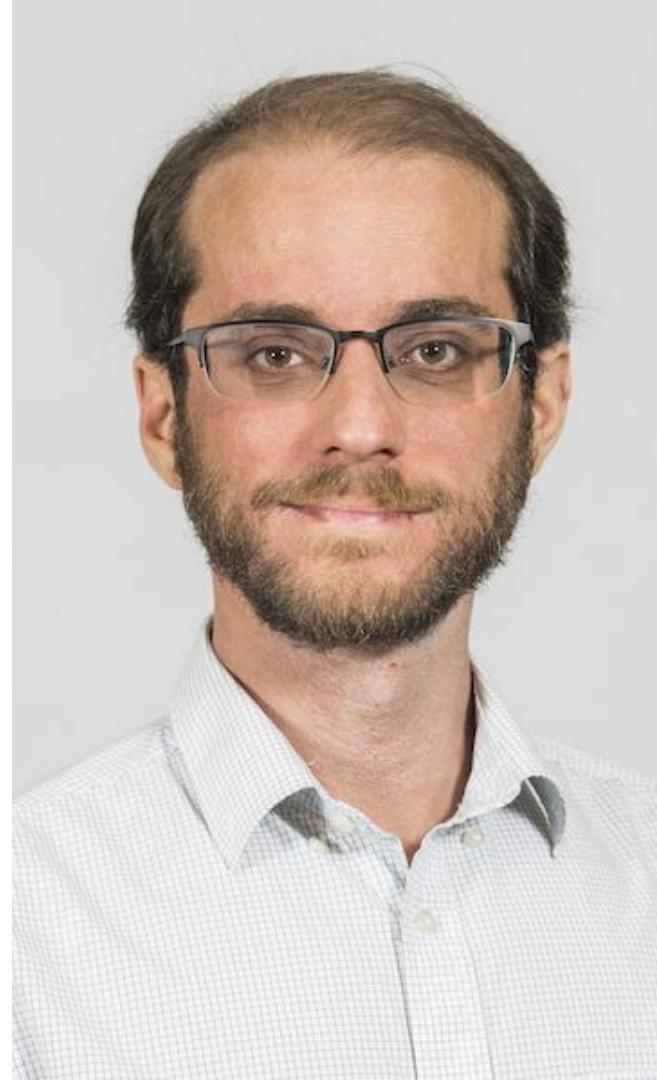
Adventures with Legacy Codebases: Tales of Incremental Improvement

ROTH MICHAELS



20
24 | 
September 15 - 20

Roth Michaels
Principal Software Engineer
Native Instruments



NATIVE INSTRUMENTS[®]



iZOTYPE



Plugin Alliance



BRAINWORX





You can do it!



```
template <typename Derived, typename Ps>
class HasProperties {
private:
    template <typename P>
    void createProperty() {
        auto success = m_propertyHolder.CreateProperty(
            getName<P>(),
            getName<typename P::property_type>(),
            T::defaultValue);
        assert(success);
    }
};
```

<https://www.youtube.com/watch?v=90l0hH5-r5A>

A Case-study in Rewriting a Legacy GUI Library for Real-time Audio Software in Modern C++



Roth Michaels



Legacy Code

...make sure new code is better

What is legacy code?

What is legacy code?

- No tests
- Lot's of code
- Very old
- Authors may be gone
- Many C++ standards
- New/old styles
- New/old paradigms
- Bad decisions from the past that once made sense
- Possibly rewritten by a less skilled eng. org.

adamtornhill / code-maat

Code Issues 15 Pull requests Actions Projects Wiki Security Insights

code-maat Public

master 7 Branches 7 Tags

Go to file

About

A command line tool to mine and analyze data from version-control systems

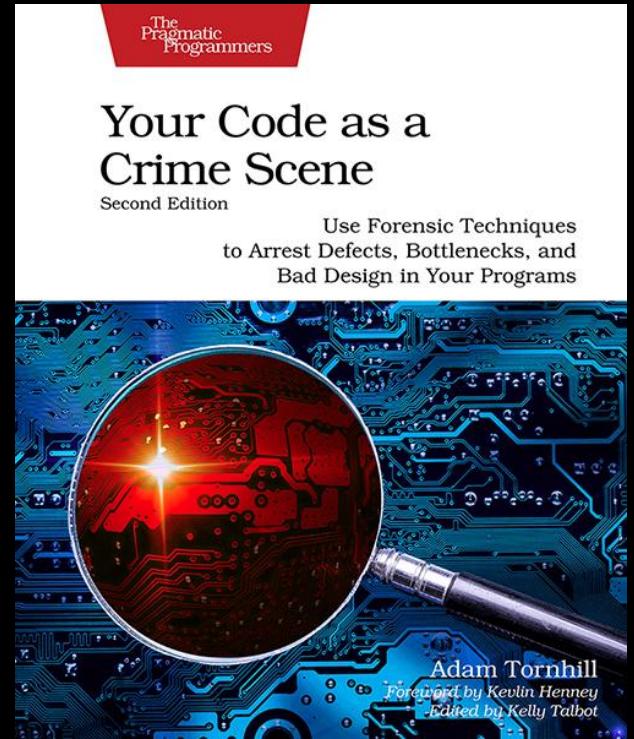
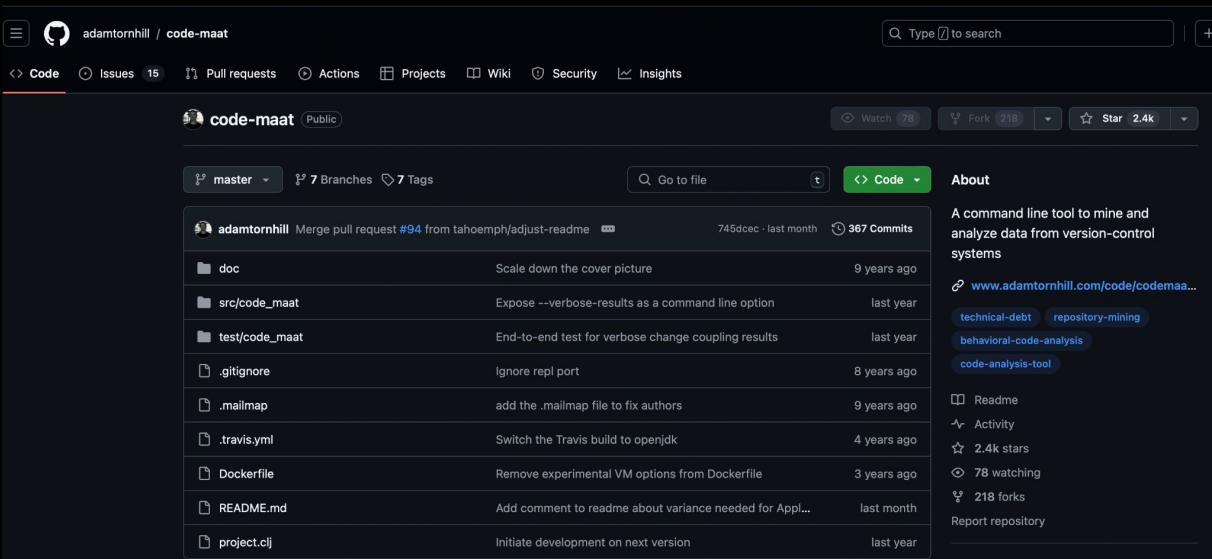
www.adamtornhill.com/code/codemaa...

technical-debt repository-mining behavioral-code-analysis code-analysis-tool

adamtornhill Merge pull request #94 from tahoemph/adjust-readme 745ddec · last month 367 Commits

File	Description	Time Ago
doc	Scale down the cover picture	9 years ago
src/code_maat	Expose --verbose-results as a command line option	last year
test/code_maat	End-to-end test for verbose change coupling results	last year
.gitignore	Ignore repl port	8 years ago
.mailmap	add the .mailmap file to fix authors	9 years ago
.travis.yml	Switch the Travis build to openjdk	4 years ago
Dockerfile	Remove experimental VM options from Dockerfile	3 years ago
README.md	Add comment to readme about variance needed for Appl...	last month
project.clj	Initiate development on next version	last year

Readme Activity 2.4k stars 78 watching 218 forks Report repository



Your Code as a Crime Scene + code-maat

How big is the universe?

~15 Millions Lines of C/C++/Objective-C(++)

~670,000 Lines

Product Code

- Ozone
- RX
- Neutron
- Nectar
- etc.

~1.33 Million Lines

Shared Code

- iZBase
- iZDSPBase
- Glass
- EqualizerIIR
- etc.

~13 Million Lines

Open Source

- Boost
- Skia
- libPNG
- libXML2
- etc.

A cautionary tale...

...of linting, typechecking, and unit tests

Story of maintaining some Python tools

- Adopt new Poetry + pyproject.toml project templates
 - New linters
 - New type-checking
 - Code changes needed to adopt
- A goal to increase test coverage
 - “Refactoring” to make code more testable

**"X% of your code will be
rewritten in 5 years."**



Kubrick, Stanley. *2001: A Space Odyssey*. (1968)



Lang, Fritz. *Metropolis*. (1927)

Don't change code without
user/business value

Evolving styleguides...

...with clang-format

Improving with static analysis

- Always provide a clang-format file
- git-hooks to automatically apply formatting to changes
- Formatting verified in CI

Improving with static analysis

...with ASAN, UBSAN, and TSAN



Roth Michaels

```
unsigned random() {  
    unsigned x  
    return x;  
}
```

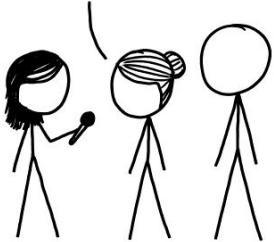
<https://www.youtube.com/watch?v=vEtGtphI3lc>

Purging Undefined Behavior & Intel
Assumptions in a Legacy C++
Codebase



ASKING AIRCRAFT DESIGNERS
ABOUT AIRPLANE SAFETY:

NOTHING IS EVER FOOLPROOF,
BUT MODERN AIRLINERS ARE
INCREDIBLY RESILIENT. FLYING IS
THE SAFEST WAY TO TRAVEL.



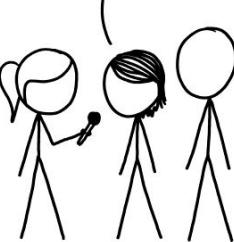
ASKING BUILDING ENGINEERS
ABOUT ELEVATOR SAFETY:

ELEVATORS ARE PROTECTED BY
MULTIPLE TRIED-AND-TESTED
FAILSAFE MECHANISMS. THEY'RE
NEARLY INCAPABLE OF FALLING.



ASKING SOFTWARE
ENGINEERS ABOUT
COMPUTERIZED VOTING:

THAT'S TERRIFYING.

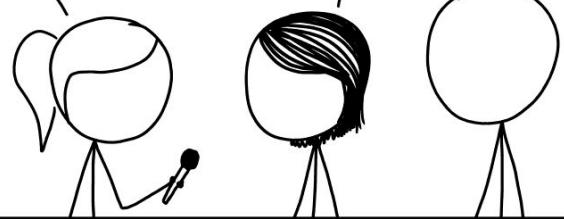


WAIT, REALLY?

| DON'T TRUST VOTING SOFTWARE AND DON'T
LISTEN TO ANYONE WHO TELLS YOU IT'S SAFE.

WHY?

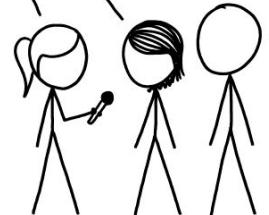
| I DON'T QUITE KNOW HOW TO PUT THIS, BUT
OUR ENTIRE FIELD IS BAD AT WHAT WE DO,
AND IF YOU RELY ON US, EVERYONE WILL DIE.



THEY SAY THEY'VE FIXED IT WITH
SOMETHING CALLED "BLOCKCHAIN."

| AAAAA!!!

| WHATEVER THEY SOLD
YOU, DON'T TOUCH IT.
BURY IT IN THE DESERT.)
WEAR GLOVES.



Improving with static analysis

Success

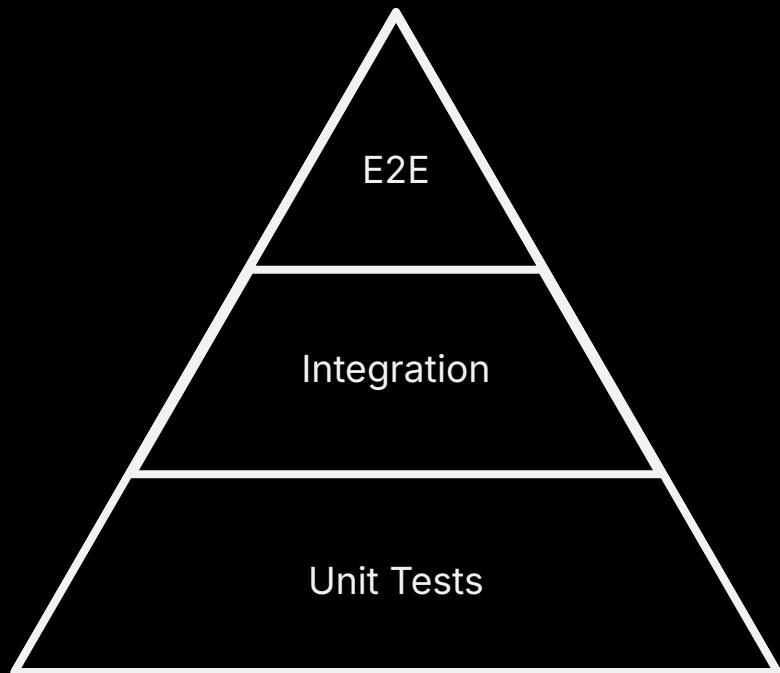
- Incremental rollout of ASAN / UBSAN

Failure?

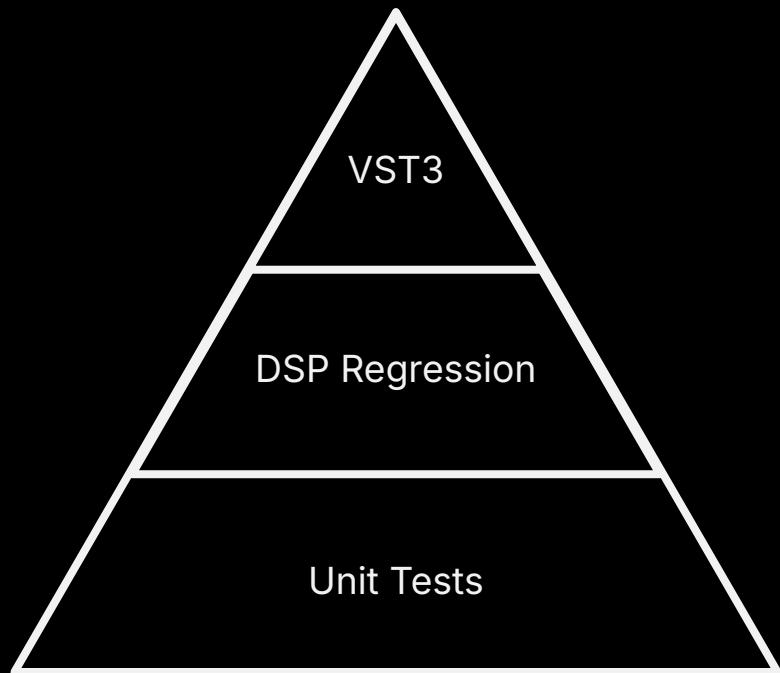
- Full product testing with TSAN

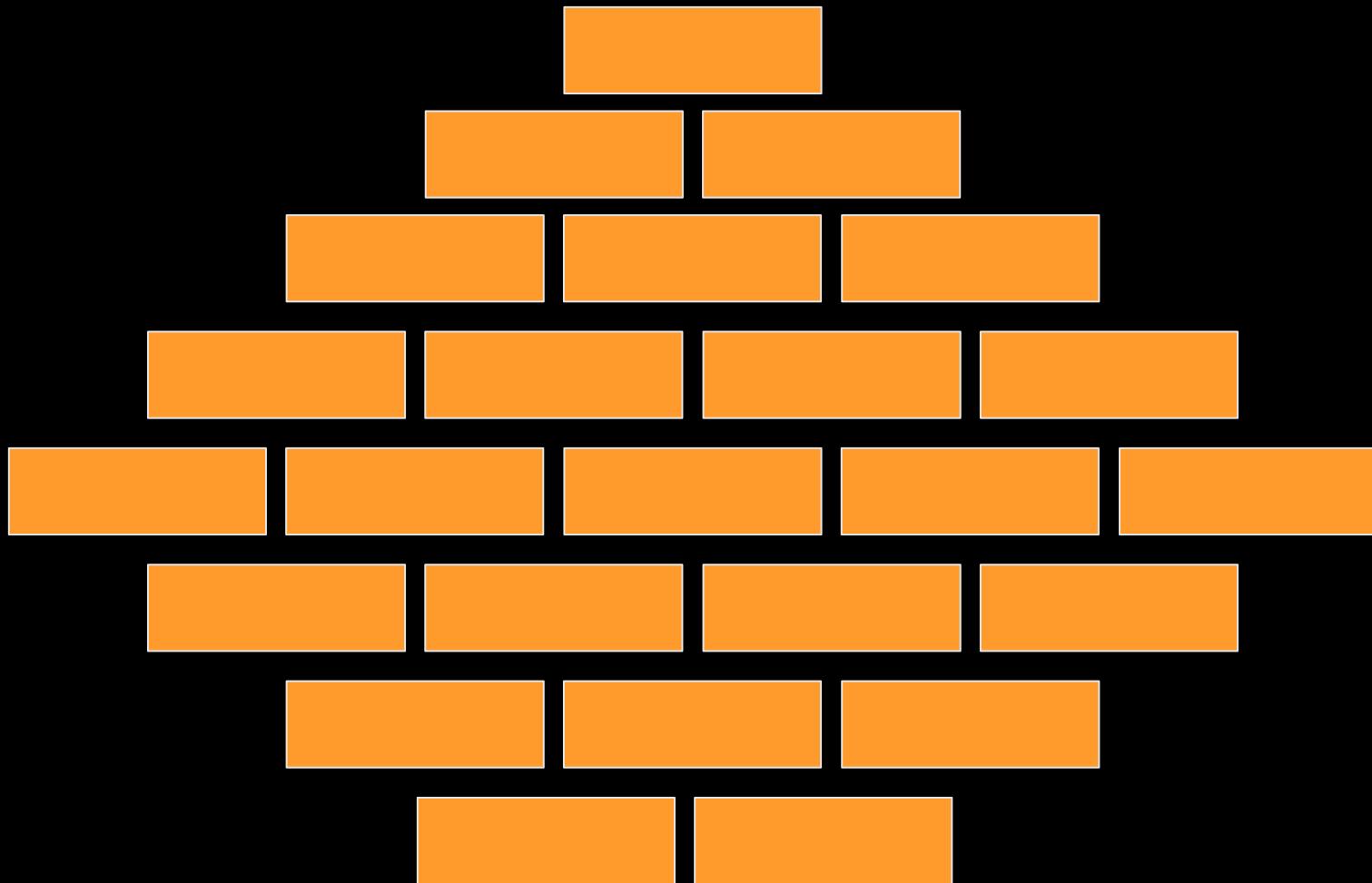
Write tests!

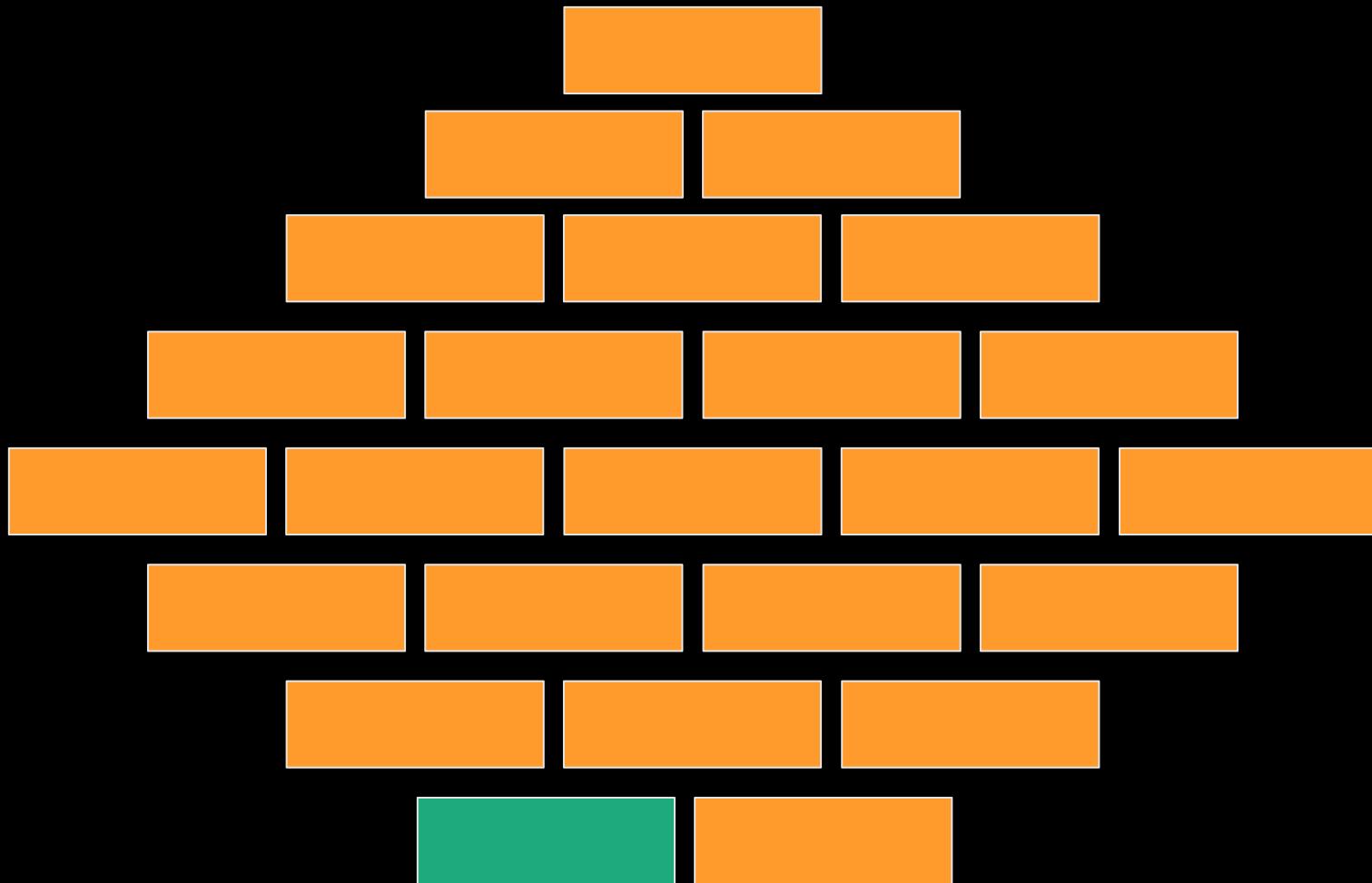
Moving up the testing pyramid

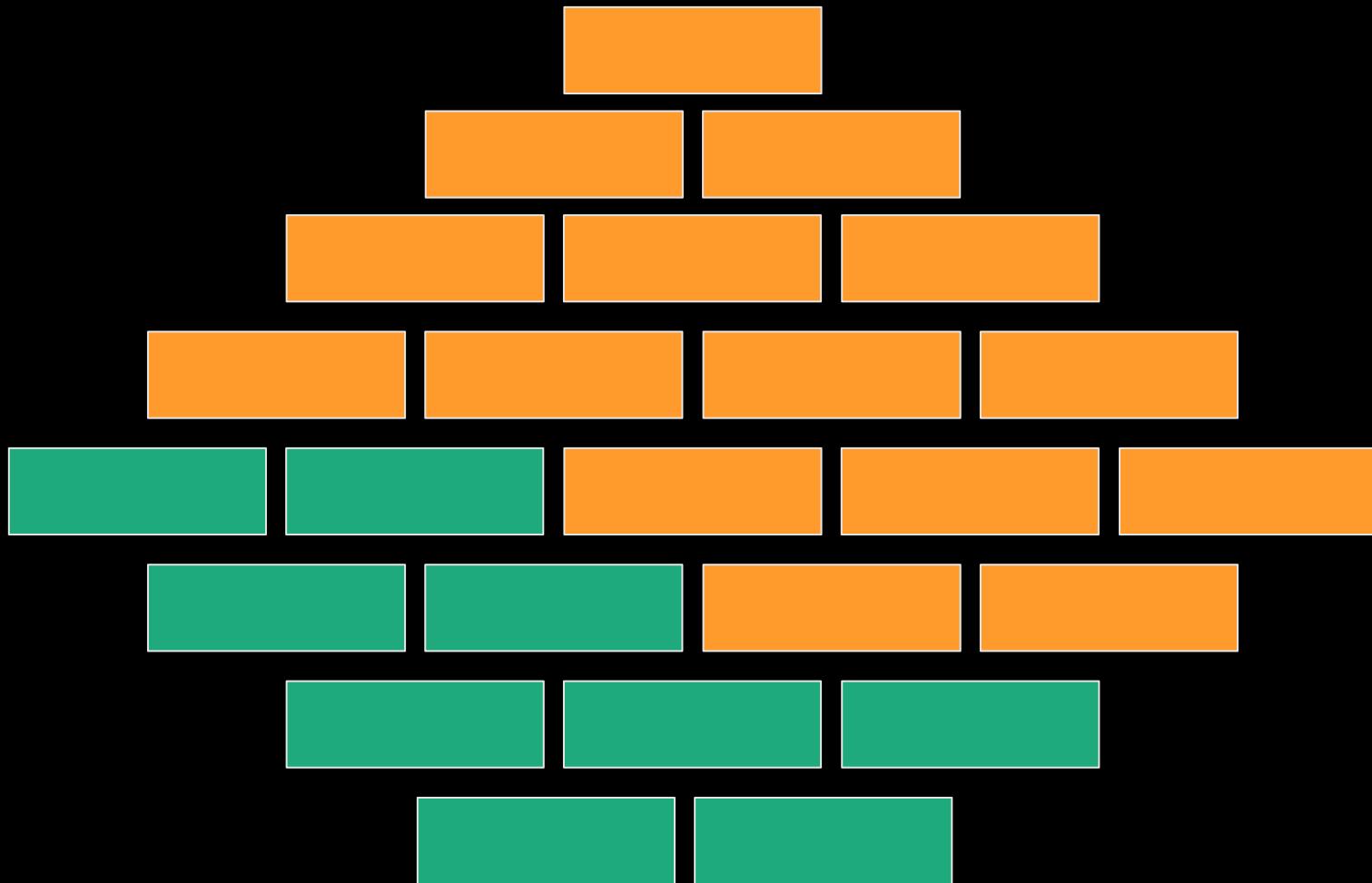


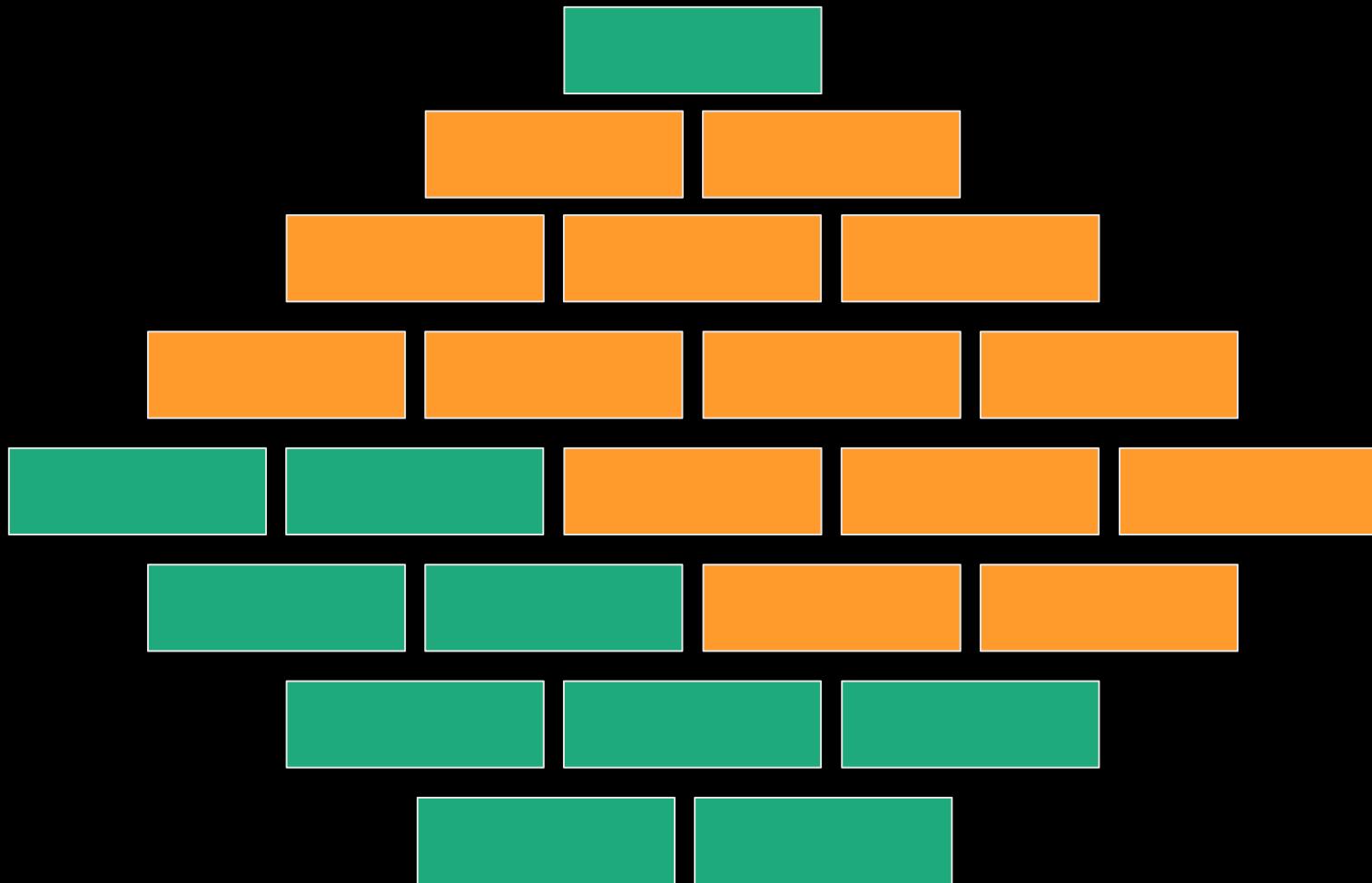
Moving up the testing pyramid











Changing legacy APIs

should you?

Breaking changes

- `class template <class T> checked_value`
- `class template <class T> boost::optional`
- Platform transition:
 - `using Optional = std::experimental::optional`
 - `using Optional = boost::optional`
- `std::optional`

LATEST: 10.17

UPDATE

CHANGES IN VERSION 10.17: THE CPU NO LONGER OVERHEATS WHEN YOU HOLD DOWN SPACEBAR.

COMMENTS:

LONGTIMEUSER4 WRITES:

THIS UPDATE BROKE MY WORKFLOW!
MY CONTROL KEY IS HARD TO REACH,
SO I HOLD SPACEBAR INSTEAD, AND I
CONFIGURED EMACS TO INTERPRET A
RAPID TEMPERATURE RISE AS "CONTROL".

ADMIN WRITES:

THAT'S HORRIFYING.

LONGTIMEUSER4 WRITES:

LOOK, MY SETUP WORKS FOR ME.
JUST ADD AN OPTION TO REENABLE
SPACEBAR HEATING.

EVERY CHANGE BREAKS SOMEONE'S WORKFLOW.

xkcd.com/1172/

No breaking changes

No breaking changes
(or use small libraries)



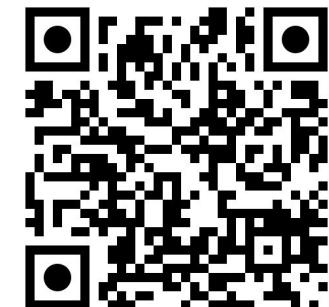
TITUS WINTERS

C++ as a

Engineering is programming
integrated over time.

<https://www.youtube.com/watch?v=tISy7EJQPzI>

C++ as a "Live at Head" Language



Can we change usage of "compatible" types?

```
void f() {  
    // Can we make this an absl::node_hash_map?  
    std::unordered_map<std::string, int> counts;  
}  
  
TEST(OrderTest, StringHashing) {  
    // Fails if the iteration order changes.  
    EXPECT_EQ(counts, ElementsAre({"foo", 5}, {"bar", 3}));  
}
```

https://www.youtube.com/watch?v=v_yzLe-wnfk

Maintainability and Refactoring Impact of Higher-Level Design Features





Kristen Shaker

<https://www.youtube.com/watch?v=torqlZnu9Ag>

How to Build Your First C++ Automated Refactoring Tool

A screenshot of the Compiler Explorer interface. On the left, there is a code editor window titled "C++ source #1" containing the following C++ code:

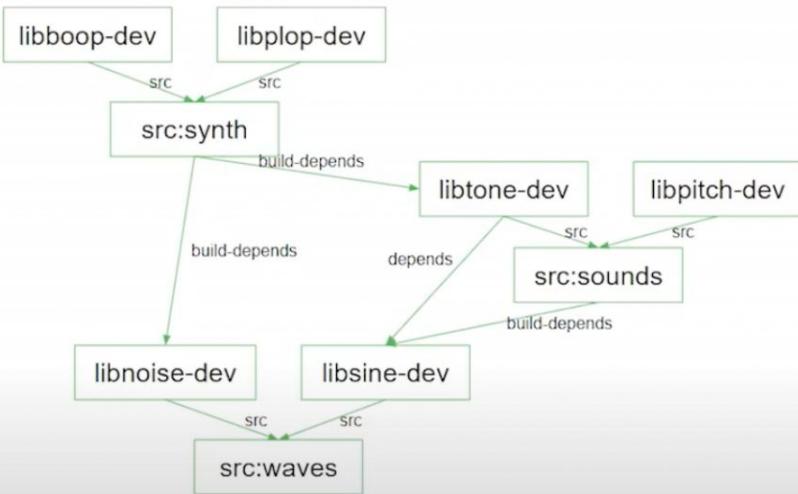
```
int main() {
    int a;
    int b;
    int c = a + b;
}
```

On the right, there is an "Ast Viewer x86-64 clang (trunk) (Editor #1, Compiler #1)" window showing the Abstract Syntax Tree (AST) for the same code. The tree structure is as follows:

```
1 TranslationUnitDecl
2   ^-FunctionDecl <line:1:1, line:5:1> line:1:5 main 'int ()'
3     ^-CompoundStmt <col:12, line:5:1>
4       |-DeclStmt <line:2:5, col:10>
5         | `~VarDecl <col:5, col:9> col:9 used a 'int'
6         |-DeclStmt <line:3:5, col:10>
7           | `~VarDecl <col:5, col:9> col:9 used b 'int'
8           ^-DeclStmt <line:4:5, col:18>
9             ^-VarDecl <col:5, col:17> col:9 c 'int' cinit
10               ^-BinaryOperator <col:13, col:17> 'int' '+' 
11                 |-ImplicitCastExpr <col:13> 'int' <LValueToRValue>
12                   |-DeclRefExpr <col:13> 'int' lvalue Var 0xbfadfa8 'a' 'int'
13                   ^-ImplicitCastExpr <col:17> 'int' <LValueToRValue>
14                     ^-DeclRefExpr <col:17> 'int' lvalue Var 0xbfae040 'b' 'int'
```



Support Multiple Build Systems



TechAtBloomberg.com

Bloomberg

<https://www.youtube.com/watch?v=R1E1tmeqxBY>

Lessons Learned from Packaging
10,000+ C++ Projects



Bret Brown
Daniel Ruoso



Static analysis and handling deprecations

...with clang-tidy

Improving with runtime analysis

- clang-tidy-diff lets you run only on changed lines
- clang-compilation database may need modifications
 - replace PCH with real header
 - compile header as Objective-C++
- Turn on deprecation warnings as errors
 - can be used to incrementally adopt warnings
- verify in CI
- have a way to disable this check in CI to deprecate

#red-diff-appreciation-society

Sharing Legacy Code

The desire to standardize

We have built the same things...

- Lots of similar and unique DSP algorithms
- 7 UI technologies
- 4 installer technologies
- etc....

HOW STANDARDS PROLIFERATE:

(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)

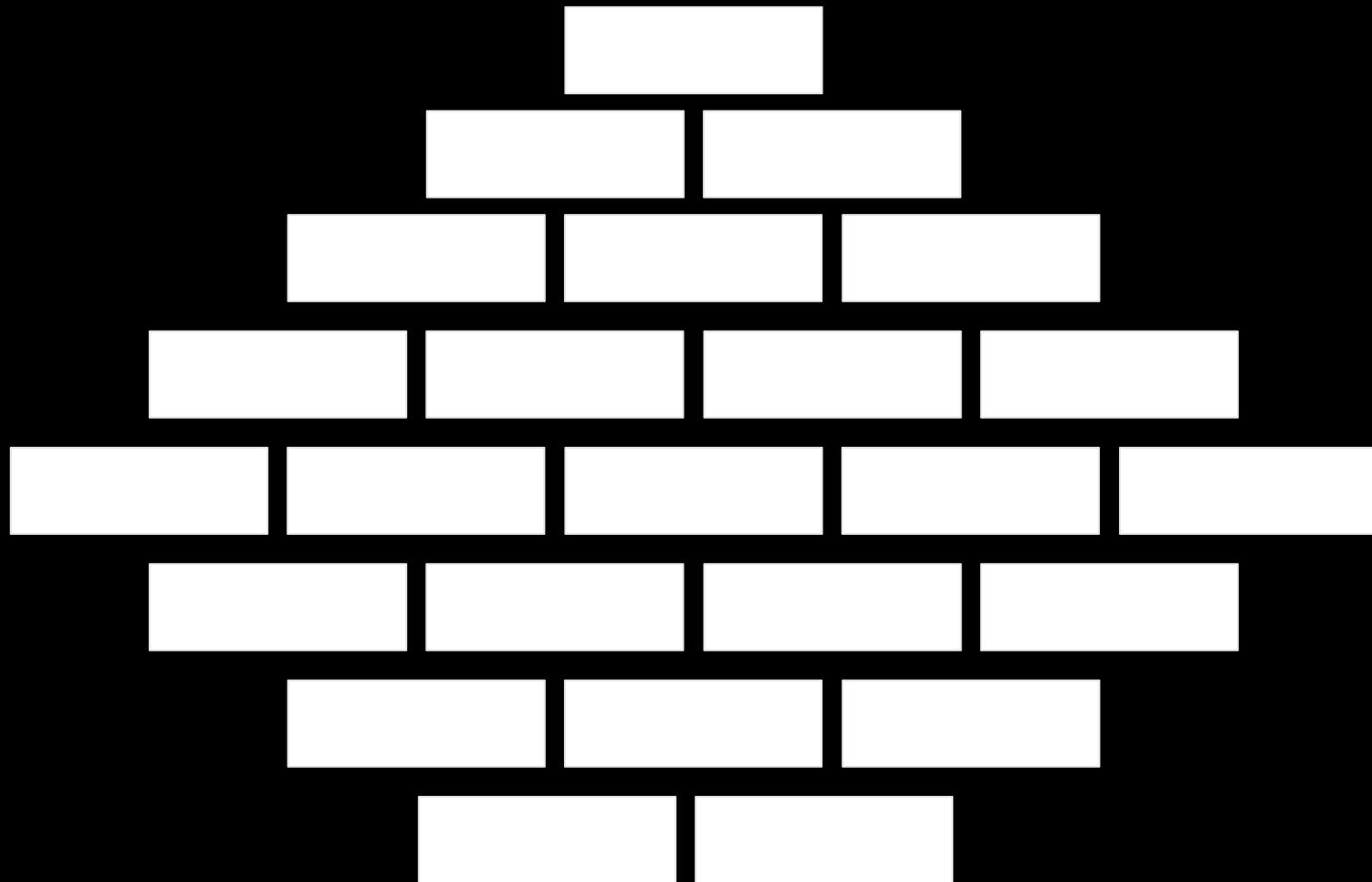
SITUATION:
THERE ARE
14 COMPETING
STANDARDS.

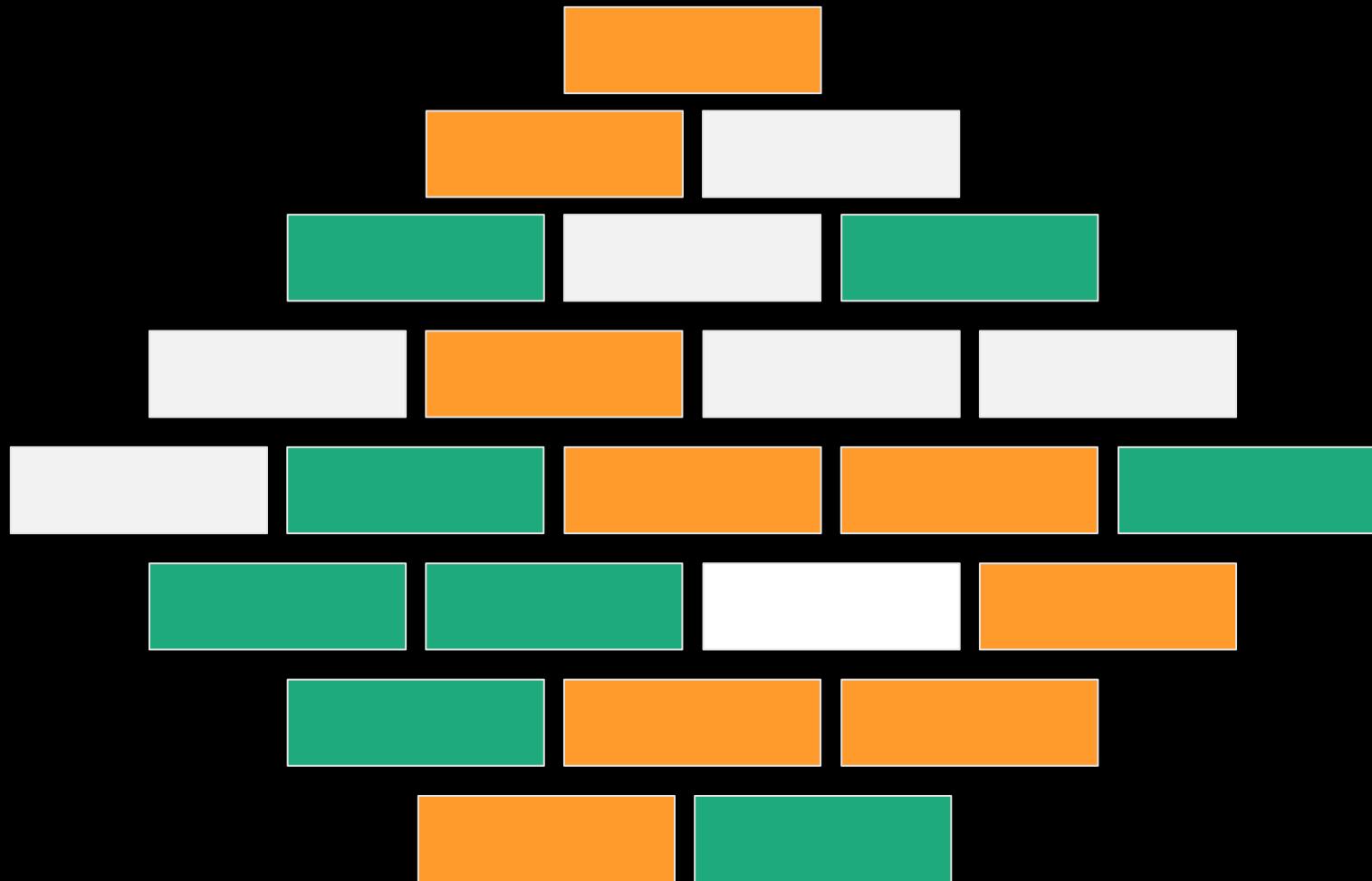
14?! RIDICULOUS!
WE NEED TO DEVELOP
ONE UNIVERSAL STANDARD
THAT COVERS EVERYONE'S
USE CASES.

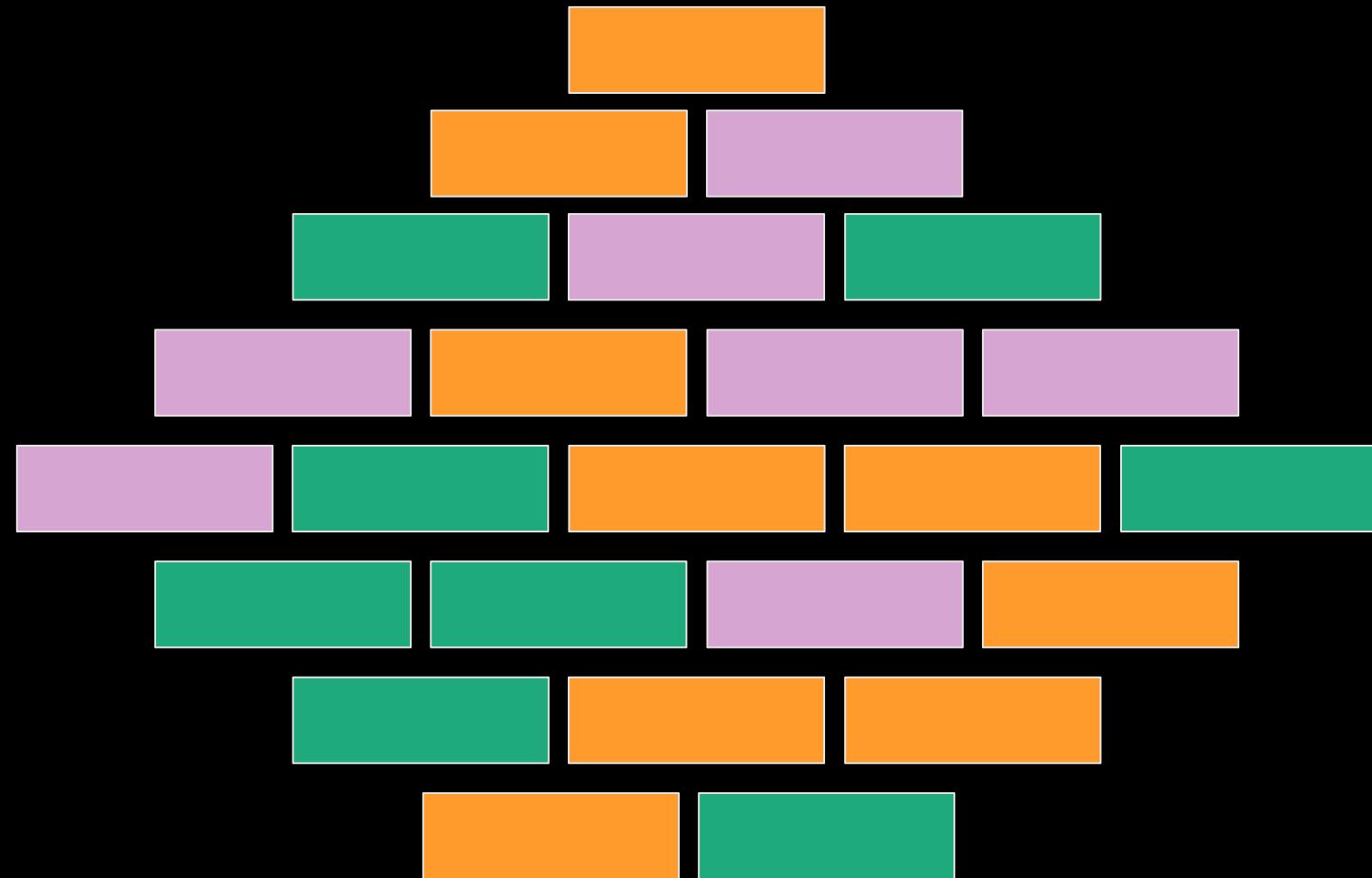


SOON:

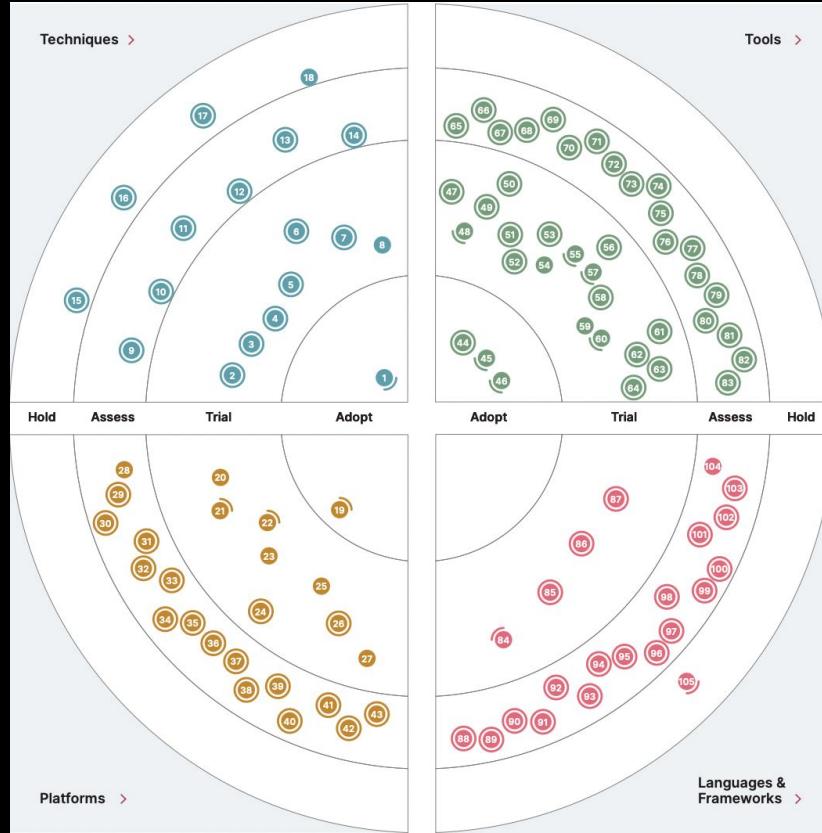
SITUATION:
THERE ARE
15 COMPETING
STANDARDS.







Discover your hidden gems



<https://www.thoughtworks.com/en-us/radar>

What challenges come with sharing legacy code?

What challenges come with sharing legacy code?

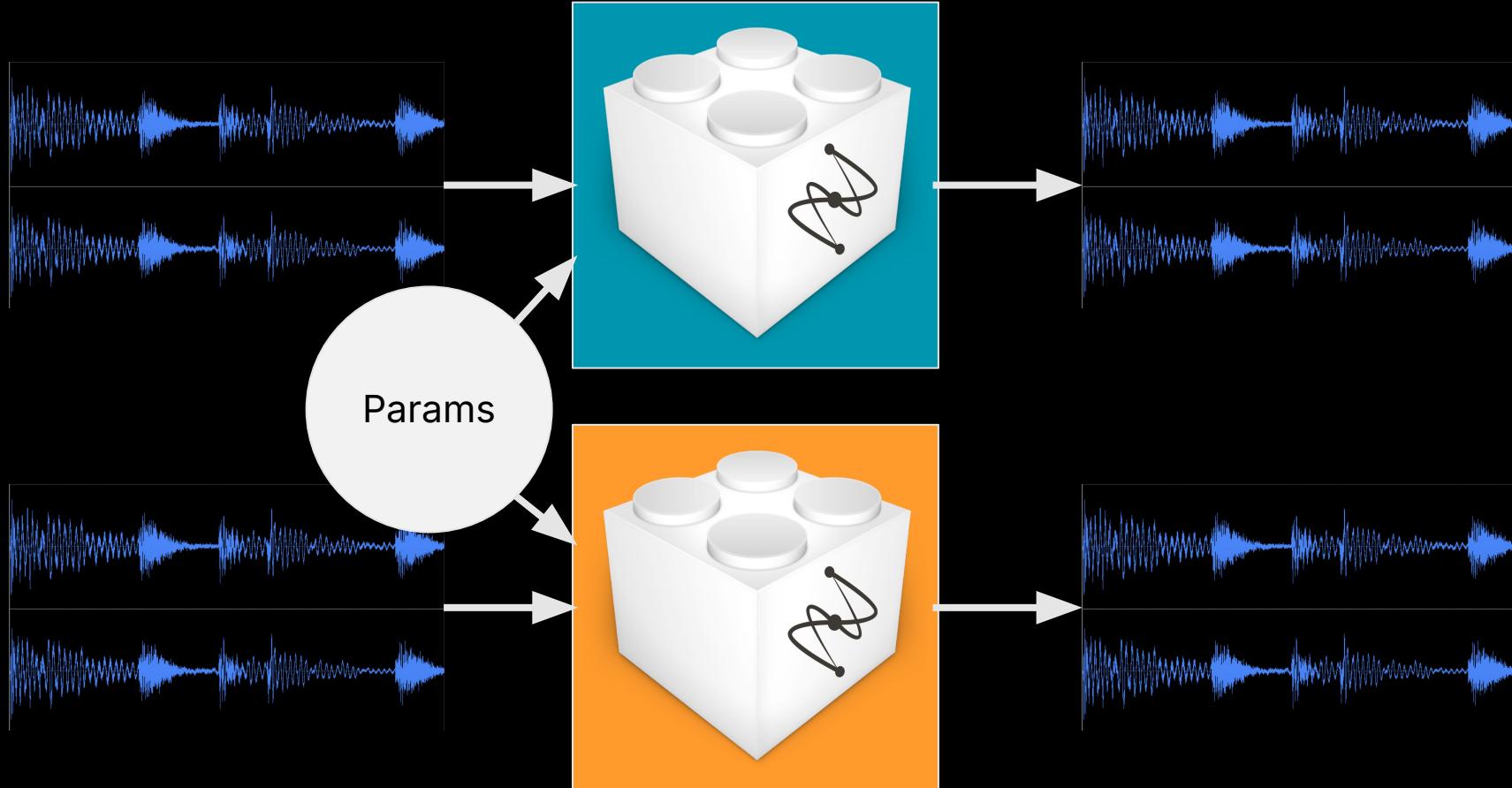
- Different build systems
- Possibly different languages standards
- Different versions of 3rd-party libraries
- Precompiled headers
- Weird legacy vocabulary types
- Global namespace solutions
- Different error handling strategies
- Different code styles

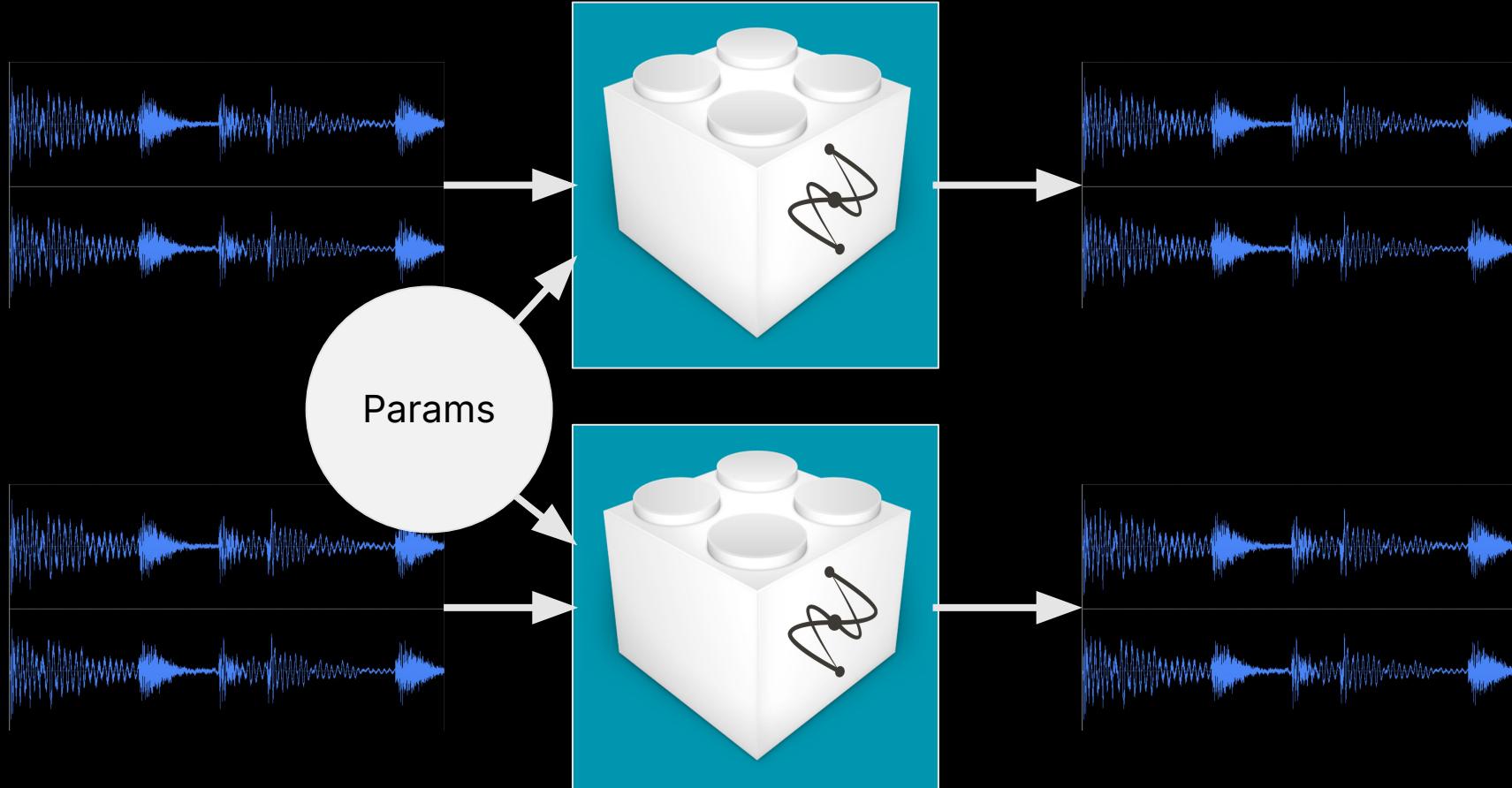
Please provide a
.clang-format

Dynamic library sharing

...with TestableDSP (TDSP)







TestableDSP Interface

```
TestableDSP* TestableDSP_Create(  
    unsigned instance,  
    unsigned cargs,  
    const char* vargs[]);  
  
void TestableDSP_Destroy(TestableDSP* tdsp);
```

TestableDSP Interface

```
class TestableDSP {
public:
    virtual void SetBool(size_t index, bool value) = 0;
    virtual void SetFloat(size_t index, float value) = 0;
    virtual void SetEnum(size_t index, unsigned value) = 0;

    virtual unsigned GetNumChannels() = 0;

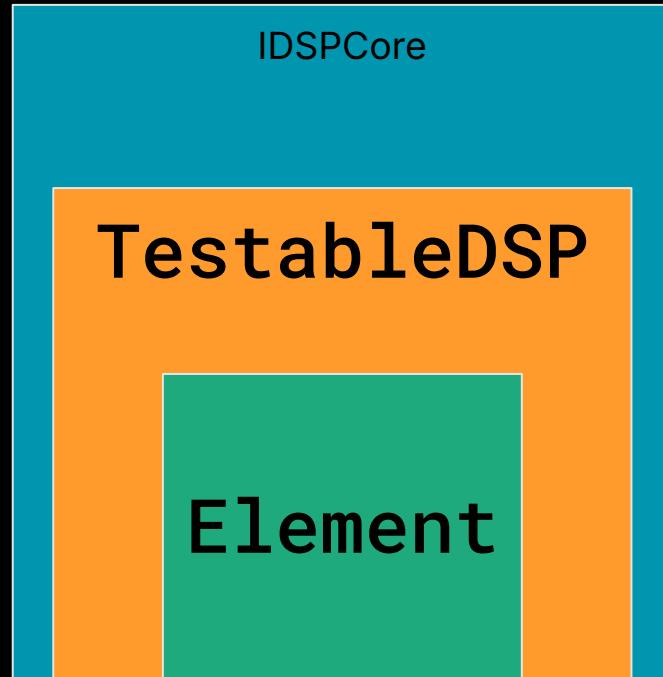
    virtual float GetSamplingRate() = 0;

    virtual void Reset() = 0;

    virtual void Process(size_t n, float* const* buf) = 0;
};
```



A successful pattern!



TestableDSP Interface

```
class __attribute__((visibility("default"))) TestableDSP {
public:
    virtual void SetBool(size_t index, bool value) = 0;
    virtual void SetFloat(size_t index, float value) = 0;
    virtual void SetEnum(size_t index, unsigned value) = 0;

    virtual unsigned GetNumChannels() = 0;

    virtual float GetSamplingRate() = 0;

    virtual void Reset() = 0;

    virtual void Process(size_t n, float* const* buf) = 0;
};
```

TestableDSP Interface

```
class __declspec(dllexport) TestableDSP {
public:
    virtual void SetBool(size_t index, bool value) = 0;
    virtual void SetFloat(size_t index, float value) = 0;
    virtual void SetEnum(size_t index, unsigned value) = 0;

    virtual unsigned GetNumChannels() = 0;

    virtual float GetSamplingRate() = 0;

    virtual void Reset() = 0;

    virtual void Process(size_t n, float* const* buf) = 0;
};
```

Source code sharing

Two stories of source code sharing

- Simple ML guitar amp modeling (not production)
 - Can be converted to header only
 - Depends on eigen (we already use)
- Adopting Native Instruments' auth library
 - Split out from monolithic nilibs library
 - Replace custom vocab types with std::
 - Align on 3rd-party libraries/versions
 - Boost, base64, nlohmann/json
 - Build scripts for both build systems

Copy+Paste sharing



Static library sharing

...sharing an analytics library



- Reactive programming event dispatch

- Event dispatch
- local caching
- opt-in management
- user-id management

Copy+Paste sharing

Source code sharing blockers

- Util::ConfigFile (plist/registry abstraction)
 - String

Dynamic library sharing

- We know it works
- Decided on layer for abstract API for virtual interface
- Would require installer tech investment to deliver
- Long term goal is source code sharing

Copy+Paste sharing gotchas

macOS

- -Qunused-arguments
- -fno-objc-msgsend-selector-stubs

Windows

- Agree on iterator debug level
- Static or Dynamic RT
- Enable MultiThreadDLL
- Other flag dances
-

Dummy dynamic library to generate static libs (NO LTO)

Summary

Incremental improvement

- Focus on improving new code; don't fix everything
- Incrementally adopt clang-format (clang-format-diff)
- Incrementally adopt sanitizers
- Incrementally adopt clang-tidy (clang-tidy-diff)



Thank you!

Roth Michaels

Principal Software Engineer

roth.michaels@native-instruments.com

@thevibesman