TESTING SOFTWARE ARCHITECTURE

@vladikk

vladikk.com

DoiT International

@vladikk

vladikk.com

DoiT International

INTERNOVUS
The Ultimate Acquisition Solution

Domain-Driven DESIGN
Tackling Complexity in the Heart of Software
Eric Evans
Foreword by Martin Fowler
The Addison-Wesley
ENTERPR
INTEGRA
PATTERN
DESIGNING, BUILDING
DEPLOYING MESSAGIN
GREGOR HOHPE
BOBBY WOOLF
WITH CONTRIBUTIONS BY
KYLE BROWN
CONRAD F. D'CRUZ
MARTIN FOWLER
SEAN NEVILLE
MICHAEL J. RETTIG
JONATHAN SIMON
Forewords by John Crupi and Martin Fowler
PATTERNS OF ENTERPRISE APPLICATION ARCHITECTURE
A MARTIN FOWLER SIGNATURE BOOK
MARTIN FOWLER
WITH CONTRIBUTIONS BY
DAVID RICE,
MATTHEW FOEMMEL,
EDWARD HIEATT,
ROBERT MEE, AND
RANDY STAFFORD
REFACTORIN
IMPROVING THE DESI
OF EXISTING CODE
MARTIN FOWLER
With contributions by Kent Beck, John Br
William Opdyke, and Don Roberts
Foreword by Erich Gamma
Object Technology International, Inc.
BOOCH
JACOBSON
RUMBAUGH
Microsoft .NET: Architecting Applications for th Enterprise
Dino Es
Andrea Sal
Robert C. Martin Series
Agile Principles, Patterns, and Practices in C#
Robert C. Martin
Design Patterns
Elements of Reusable Object-Oriented Software
Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides
Copyrighted Material
Robert C. Martin Series
PRENTICE HALL
Clean Code
A Handbook of Agile Software Craftsmanshi
Copyrighted Material
Jimmy Nil
Applying Domain
Design and Patter
Copyrighted Material
The Addison-Wesley Signat
TEST-DRIVE
DEVELOPME
BY EXAMPLE
KENT BECK

Design Patterns

Libraries / Frameworks

Infrastructural components

Programming languages

Adapting to changes in business

Adapting to changes in technology

Testing out assumptions

Evolutionary architecture

Adapting to changes in business

Adapting to changes in technology

Testing out assumptions

Evolutionary architecture

Software Architecture

Testing Software Architecture

Automation Architectural Tests

SOFTWARE ARCHITECTURE

*"**Architecture** is about the **important** stuff, whatever that is"*

- Ralph Johnson

*"**Architecture** is about the **important** stuff, whatever that is"*

- Ralph Johnson

*"**Software architecture** is those decisions which are both **important** and hard to change"*

- Martin Fowler

SOFTWARE SOLVES PROBLEMS

SOFTWARE SOLVES PROBLEMS

GOOD DESIGN SUPPORTS THE BUSINESS

SOFTWARE SOLVES PROBLEMS

GOOD DESIGN SUPPORTS THE BUSINESS

…but it’s hard!!!

Infrastructure

Databases

Architectural styles

Frameworks

Languages

Methodologies

Infrastructure

Databases

Architectural styles

Frameworks

Languages

Methodologies

Infrastructure

Databases

Architectural styles

Frameworks

Languages

Methodologies

THE GOAL OF SOFTWARE ARCHITECTURE

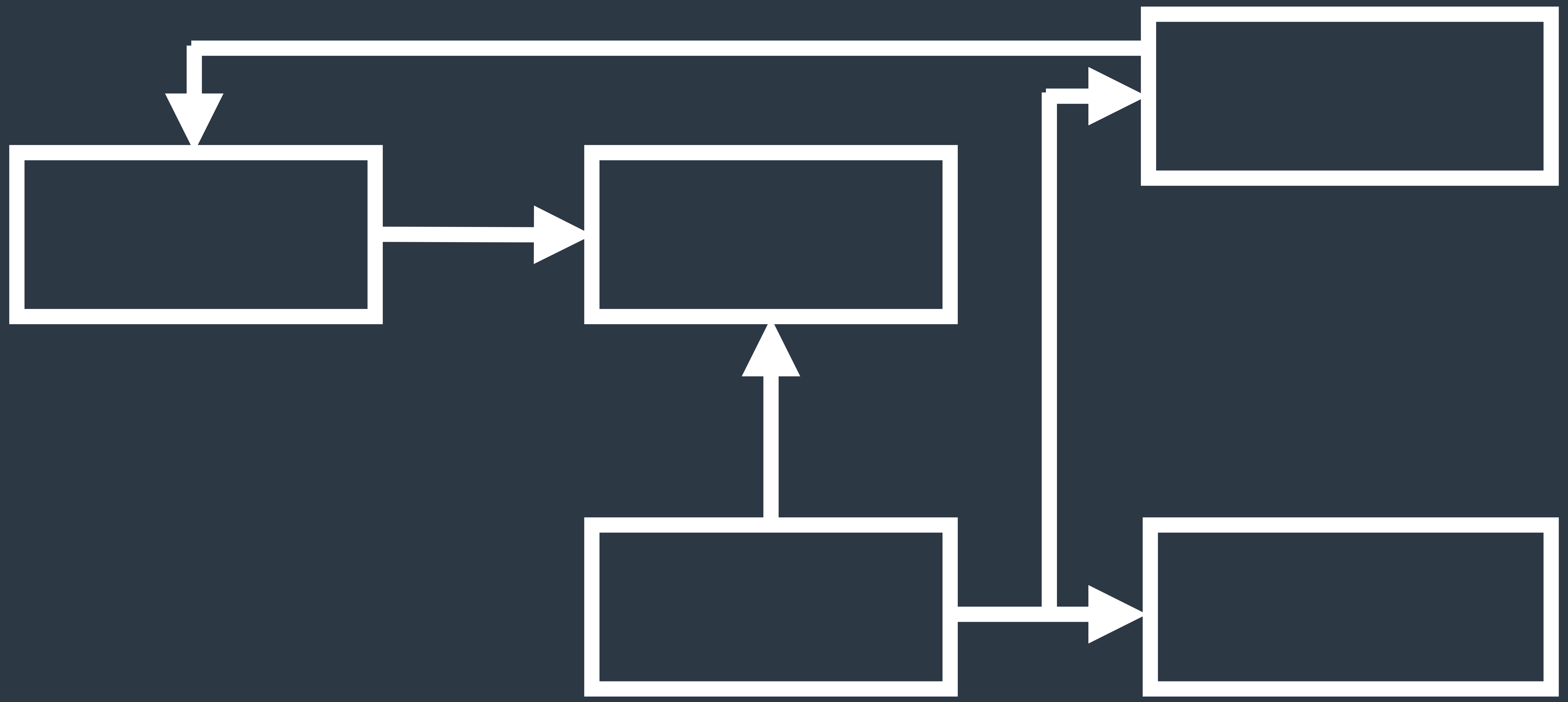
IS TO MANAGE COMPLEXITY!

Complexity Management

1. Component level

2. System level

3. Implementation level

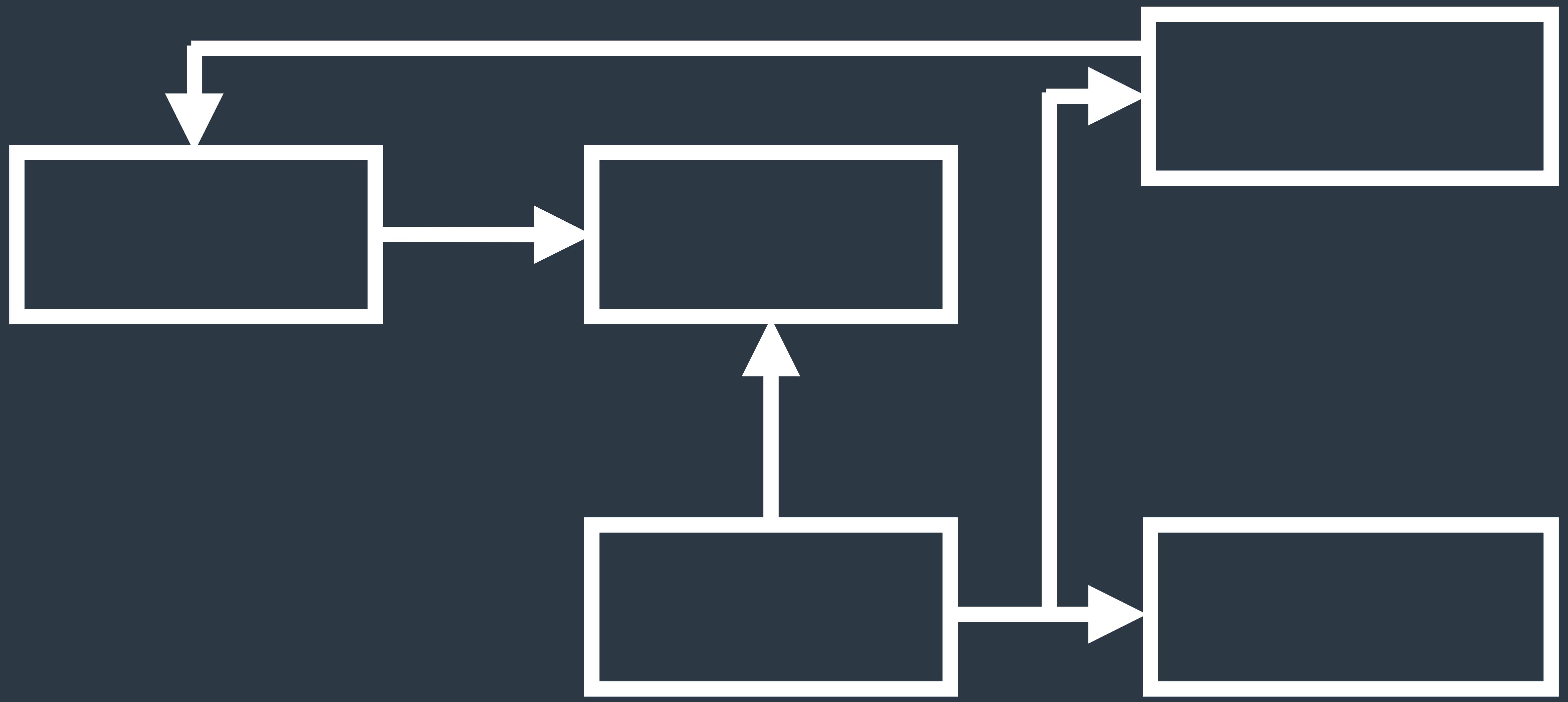


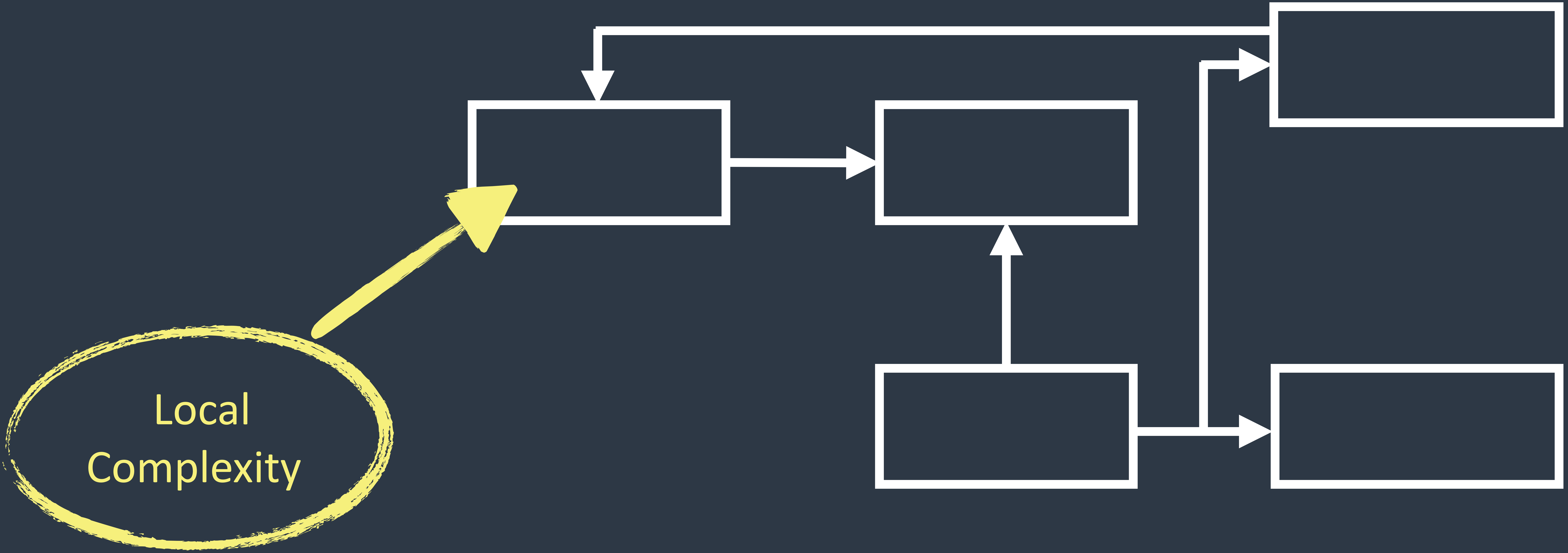
Complexity Management

1. Component level

2. System level

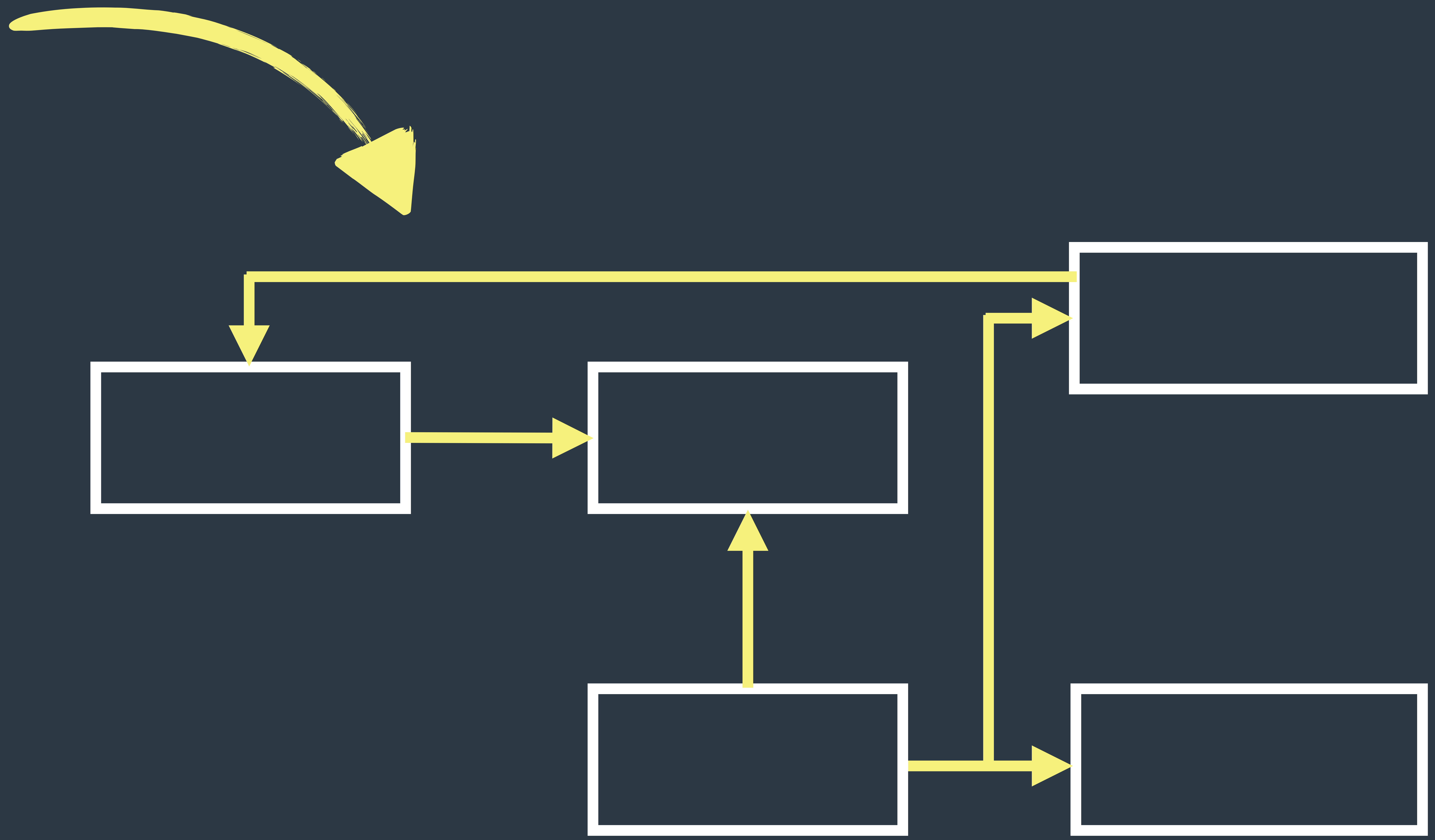
3. Implementation level



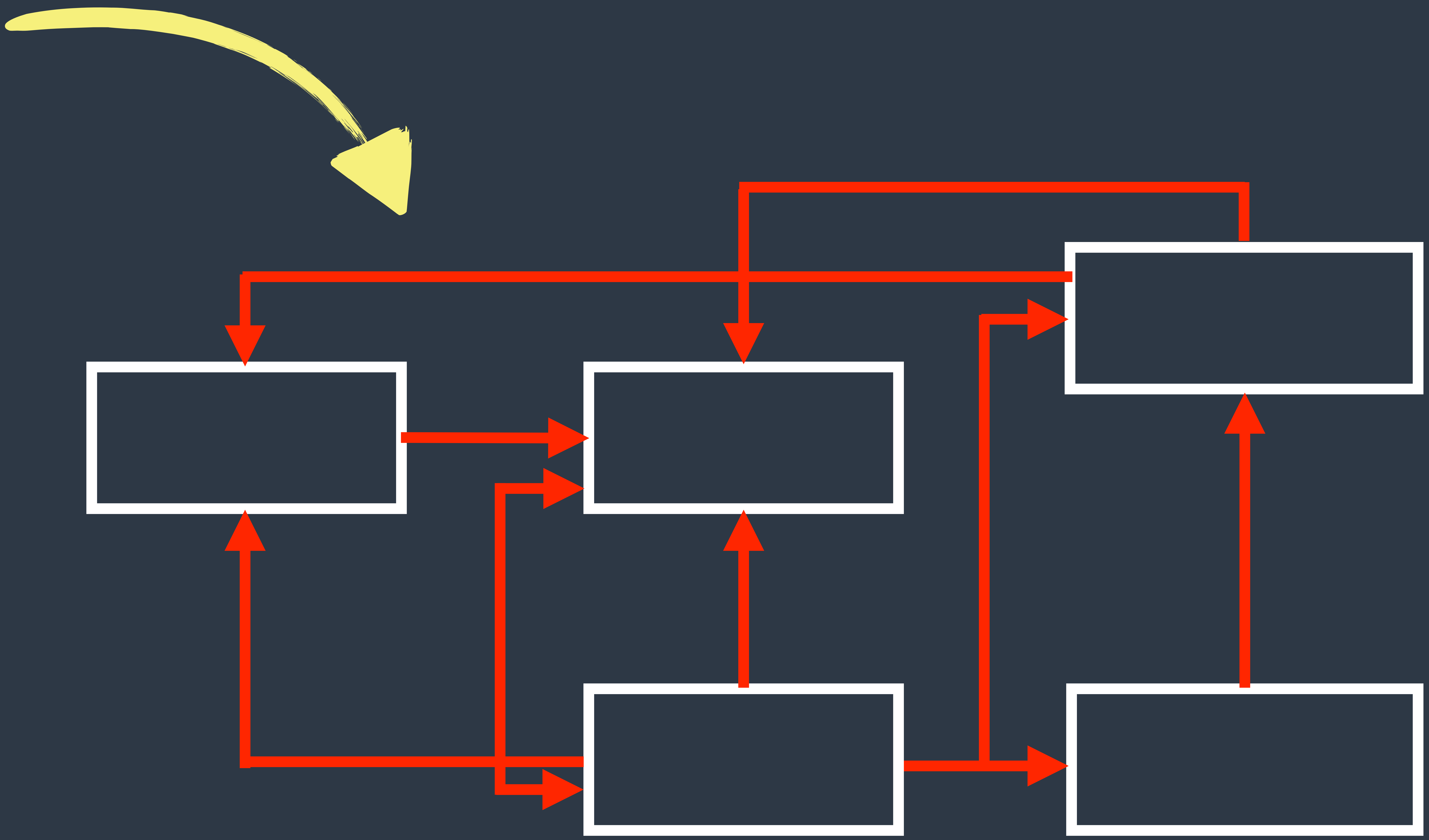


Local
Complexity

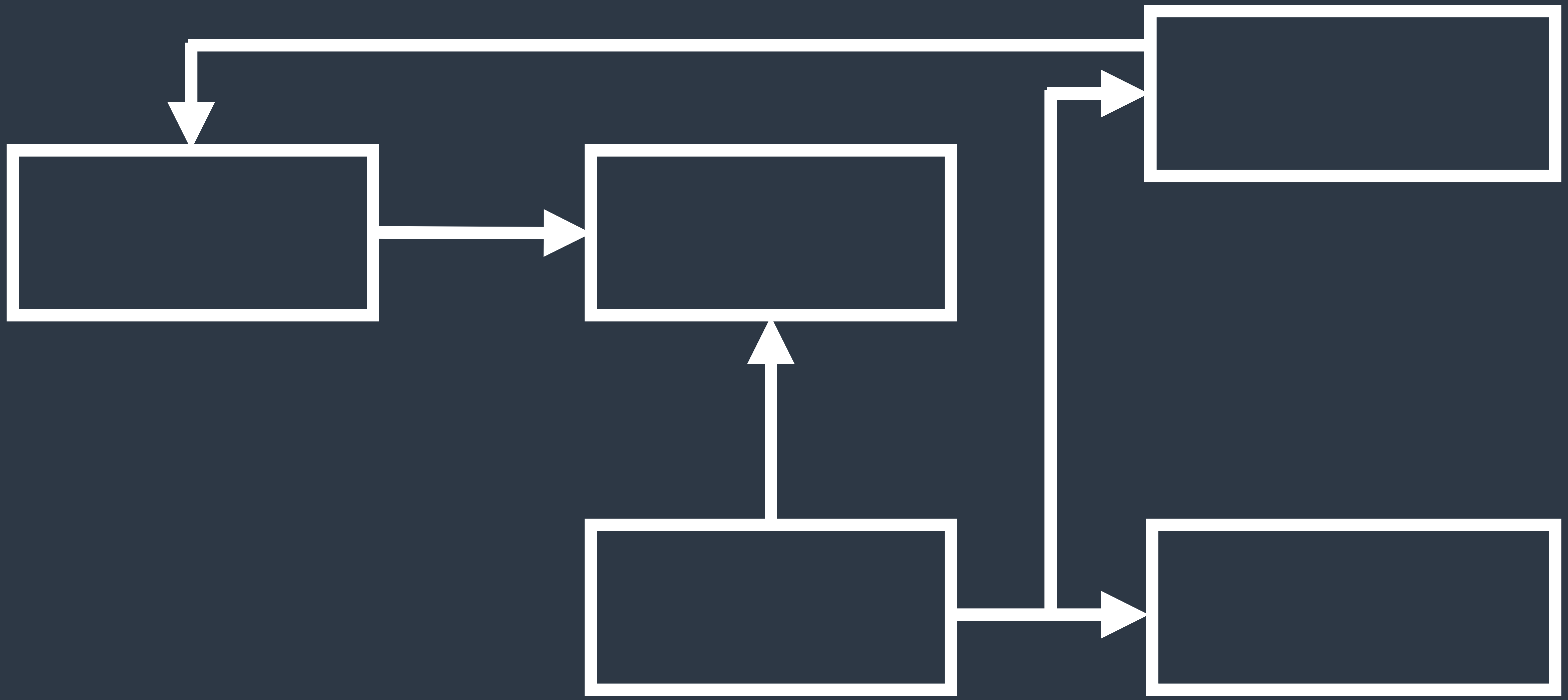
Global Complexity



Global Complexity



ARCHITECTURE = BOUNDARIES DESIGN



> ***"System design … is inherently about boundaries - what’s in, what's out, what spans, what moves between, and about tradeoffs.*** *It reshapes what is outside, just as it shapes what is inside.”*
>
> *- Ruth Malan*

"System design … is inherently about boundaries - what’s in, what's out, what spans, what moves between, and about tradeoffs. *It reshapes what is outside, just as it shapes what is inside.”*

- Ruth Malan

Big design upfront

No design

Architect

Evolutionary architecture

Big design upfront

No design

Architect

Evolutionary architecture

Big design upfront

No design

Architect

Evolutionary architecture

Big design upfront

No design

Architect

Evolutionary architecture

Big design upfront

No design

Architect

Evolutionary architecture

ENTROPY -
GRADUAL DECLINE INTO DISORDER

TESTING SOFTWARE ARCHITECTURE

FUNCTIONAL TESTING:

UNIT, INTEGRATION, END-TO-END

FUNCTIONAL TESTING:

UNIT, INTEGRATION, END-TO-END

NON-FUNCTIONAL TESTING:

PERFORMANCE, STRESS, SECURITY, USABILITY, ETC

FUNCTIONAL TESTING:

UNIT, INTEGRATION, END-TO-END

NON-FUNCTIONAL TESTING:

PERFORMANCE, STRESS, SECURITY, USABILITY, ETC

DESIGN TESTING:

LOCAL AND GLOBAL COMPLEXITIES OF THE DESIGN

Good Architecture

Easy to navigate

Easy to contribute

Easy to change

Delivers business goals

Bad Architecture

Hard to navigate

Hard to contribute

Hard to change

Under/over engineered

Big ball of mud

"A Big Ball of Mud is a ***haphazardly structured, sprawling, sloppy, duct-tape-and-baling-wire, spaghetti-code jungle****. These systems show unmistakable signs of* ***unregulated growth****, and repeated,* ***expedient repair****.* ***Information is shared promiscuously among distant elements of the system****, often to the point where nearly all the important information becomes global or duplicated.”*

- Joseph Yoder

ARCHITECTURE = BOUNDARIES DESIGN

GOOD ARCHITECTURE:

GOOD BOUNDARIES

BAD ARCHITECTURE:

BAD BOUNDARIES

BAD BOUNDARIES = EVERYBODY SUFFERS

Software Engineers

Hard to add new functionality

Hard to change existing functionality

Impossible to unit test

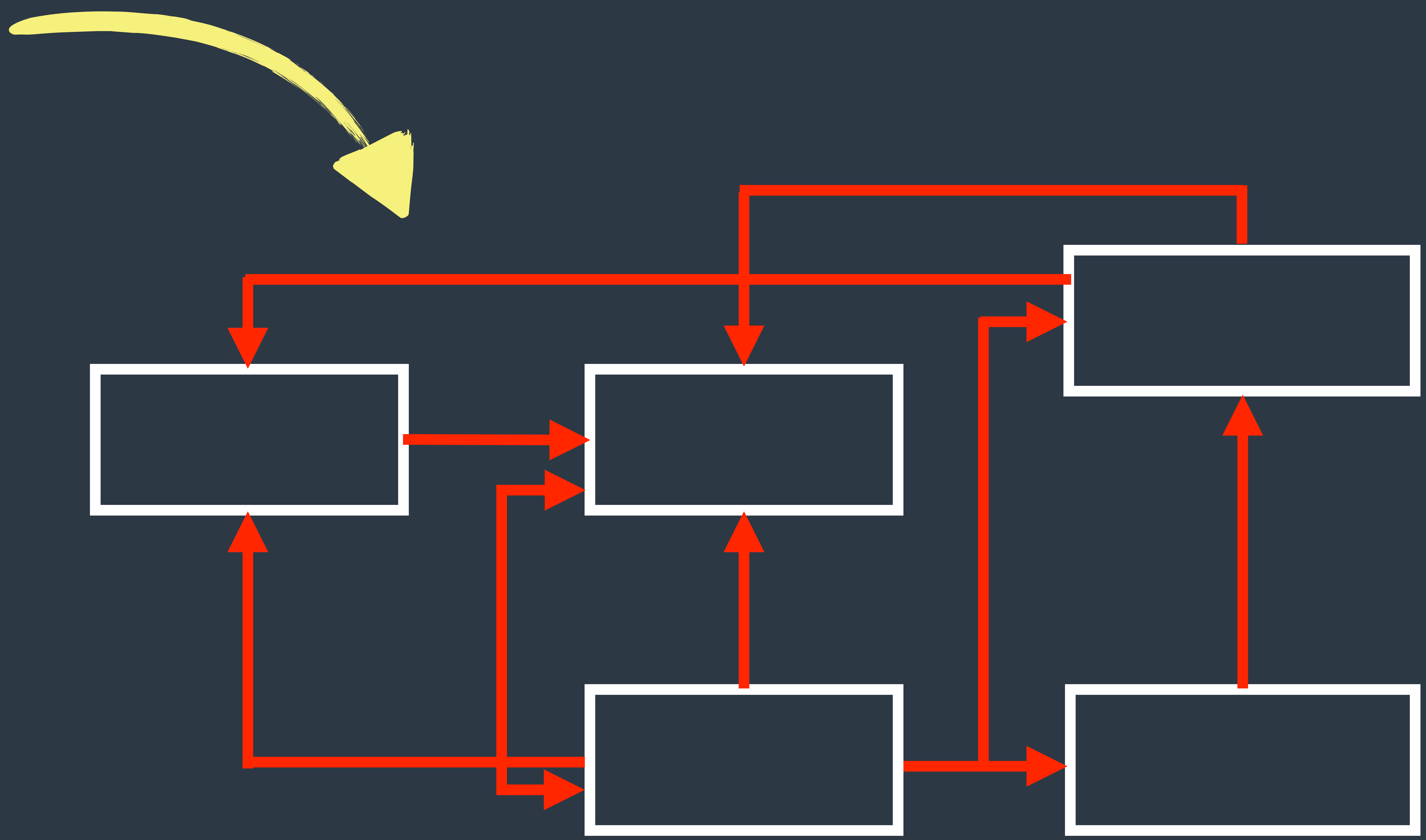
Software Engineers

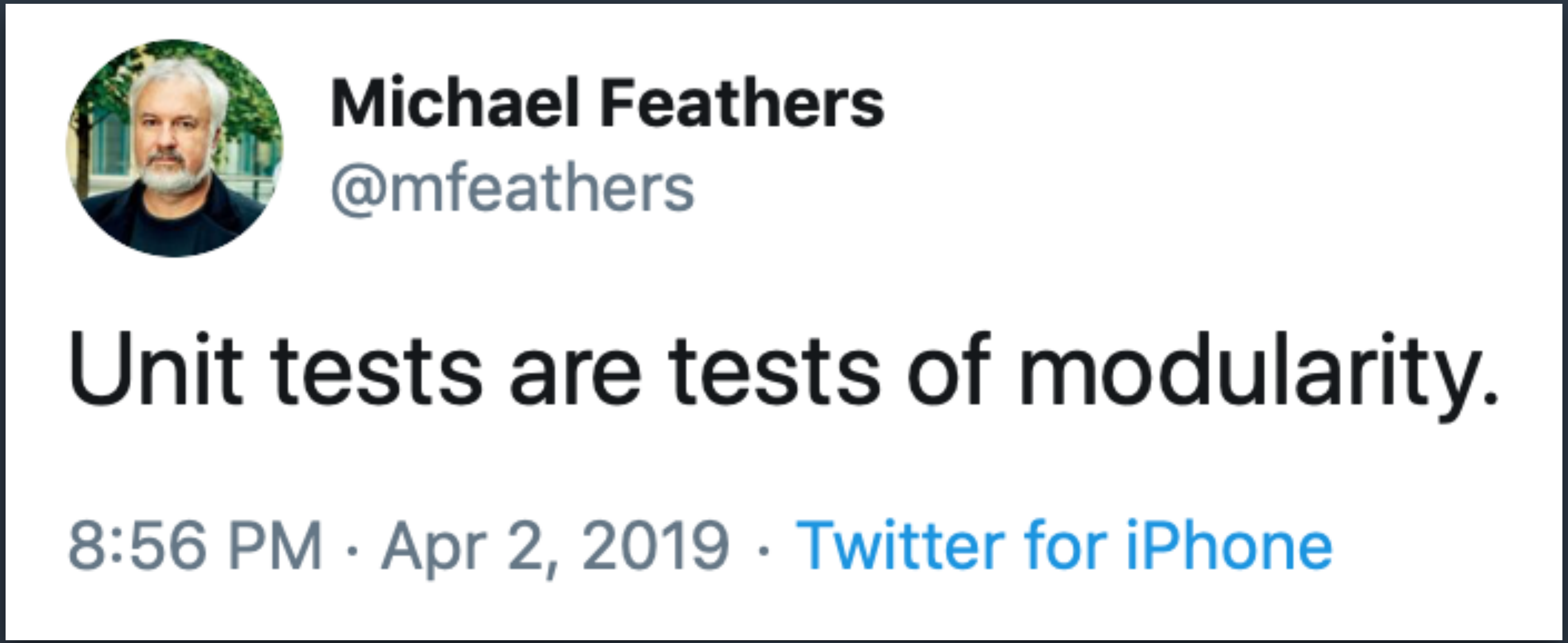
Hard to add new functionality

Hard to change existing functionality

Impossible to unit test

Global Complexity





Michael Feathers
@mfeathers

Unit tests are tests of modularity.

8:56 PM · Apr 2, 2019 · Twitter for iPhone

QA Engineers

Defects caused by logic duplication

Regression testing in-proportionally long

QA Engineers

Defects caused by logic duplication

Regression testing in-proportionally long

Database Administrators

Multiple sources of truth

Inconsistent data that has to be fixed manually

Lack of transactions due to design constraints

Database Administrators

Multiple sources of truth

Inconsistent data that has to be fixed manually

Lack of transactions due to design constraints

Operations

No DevOps culture

Big releases

Rare deployments

Complex configurations for components

Software Engineers

Hard to add new functionality

Hard to change existing functionality

Impossible to unit test

QA Engineers

Defects caused by logic duplication

Regression testing in-proportionally long

BAD BOUNDARIES = EVERYBODY SUFFERS

Operations

No DevOps culture

Big releases

Rare deployments

Complex configurations for components

Database Administrators

Multiple sources of truth

Inconsistent data that has to be fixed manually

Lack of transactions due to design constraints

GOOD ARCHITECTURE
REQUIRES SHARED OWNERSHIP

AUTOMATING ARCHITECTURAL TESTS

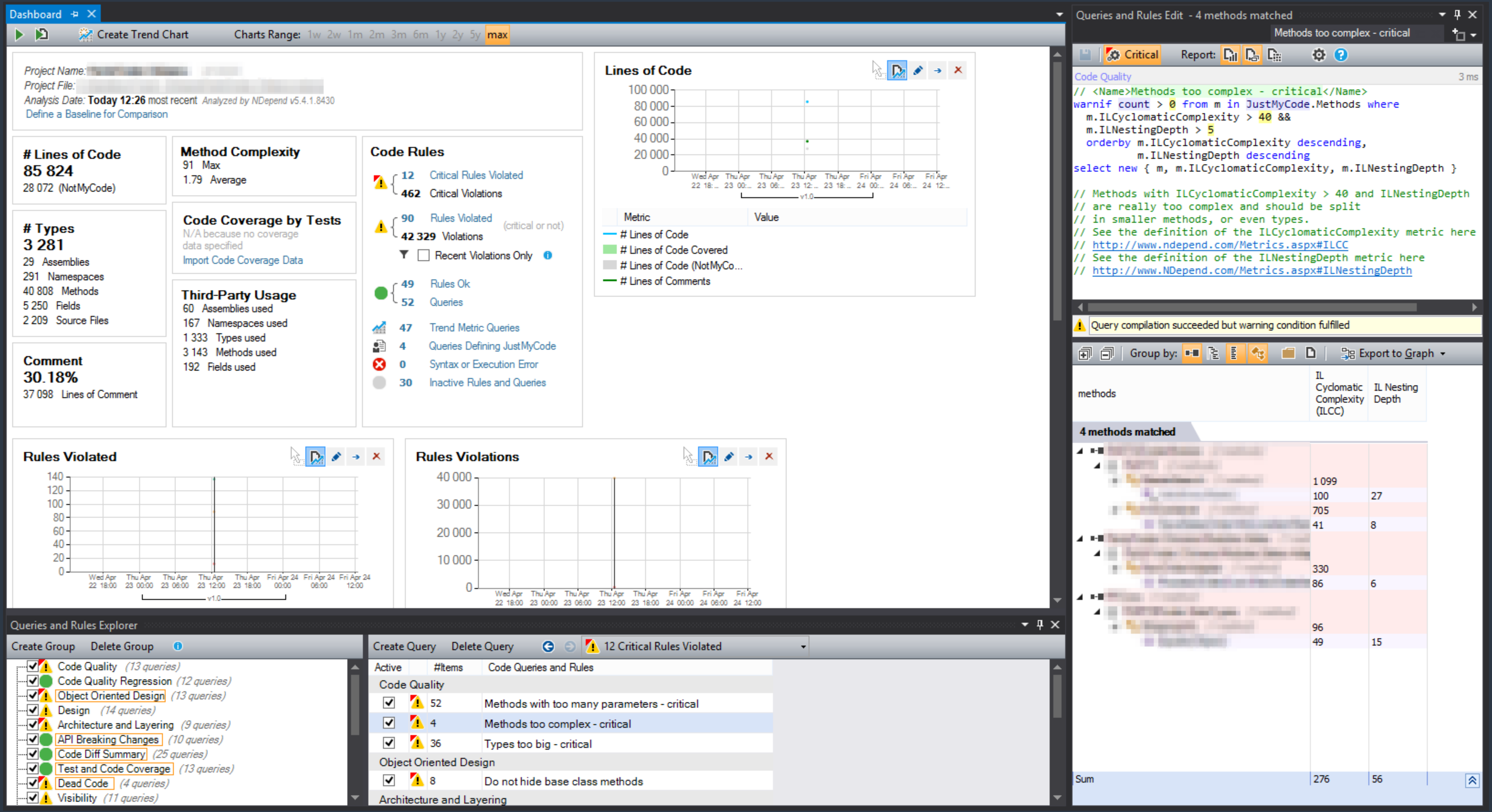
Static Code Analysis

Analyze the codebase

Query the code's health

Flexible health rules definitions

Build pipeline integrations



Dashboard

Create Trend Chart   Charts Range: 1w 2m 1m 2m 3m 6m 1y 2y 5y max

Project Name: [redacted]
Project File: [redacted]
Analysis Date: **Today 12:26** most recent *Analyzed by NDepend v5.4.1.8430*
Define a Baseline for Comparison

# Lines of Code
85 824
28 072 (NotMyCode)

# Types
3 281

29	Assemblies
291	Namespaces
40 808	Methods
5 250	Fields
2 209	Source Files

Comment
30.18%
37 098 Lines of Comment

Method Complexity

91	Max
1.79	Average

Code Coverage by Tests
N/A because no coverage data specified
Import Code Coverage Data

Third-Party Usage

60	Assemblies used
167	Namespaces used
1 333	Types used
3 143	Methods used
192	Fields used

Code Rules

12	Critical Rules Violated
462	Critical Violations
90	Rules Violated (critical or not)
42 329	Violations

☐ Recent Violations Only

49	Rules Ok
52	Queries
47	Trend Metric Queries
4	Queries Defining JustMyCode
0	Syntax or Execution Error
30	Inactive Rules and Queries

Lines of Code

Metric	Value
# Lines of Code	
# Lines of Code Covered	
# Lines of Code (NotMyCo...	
# Lines of Comments	

Rules Violated

Rules Violations

Queries and Rules Explorer

Create Group   Delete Group

- ☑ Code Quality (13 queries)
- ☑ Code Quality Regression (12 queries)
- ☑ Object Oriented Design (13 queries)
- ☑ Design (14 queries)
- ☑ Architecture and Layering (9 queries)
- ☑ API Breaking Changes (10 queries)
- ☑ Code Diff Summary (25 queries)
- ☑ Test and Code Coverage (13 queries)
- ☑ Dead Code (4 queries)
- ☑ Visibility (11 queries)

Create Query   Delete Query   12 Critical Rules Violated

Active	#Items	Code Queries and Rules
Code Quality		
☑	52	Methods with too many parameters - critical
☑	4	Methods too complex - critical
☑	36	Types too big - critical
Object Oriented Design		
☑	8	Do not hide base class methods
Architecture and Layering		

Queries and Rules Edit - 4 methods matched

Methods too complex - critical

Critical   Report:

Code Quality   3 ms

```
// <Name>Methods too complex - critical</Name>
warnif count > 0 from m in JustMyCode.Methods where
  m.ILCyclomaticComplexity > 40 &&
  m.ILNestingDepth > 5
  orderby m.ILCyclomaticComplexity descending,
          m.ILNestingDepth descending
select new { m, m.ILCyclomaticComplexity, m.ILNestingDepth }

// Methods with ILCyclomaticComplexity > 40 and ILNestingDepth
// are really too complex and should be split
// in smaller methods, or even types.
// See the definition of the ILCyclomaticComplexity metric here
// http://www.ndepend.com/Metrics.aspx#ILCC
// See the definition of the ILNestingDepth metric here
// http://www.NDepend.com/Metrics.aspx#ILNestingDepth
```

Query compilation succeeded but warning condition fulfilled

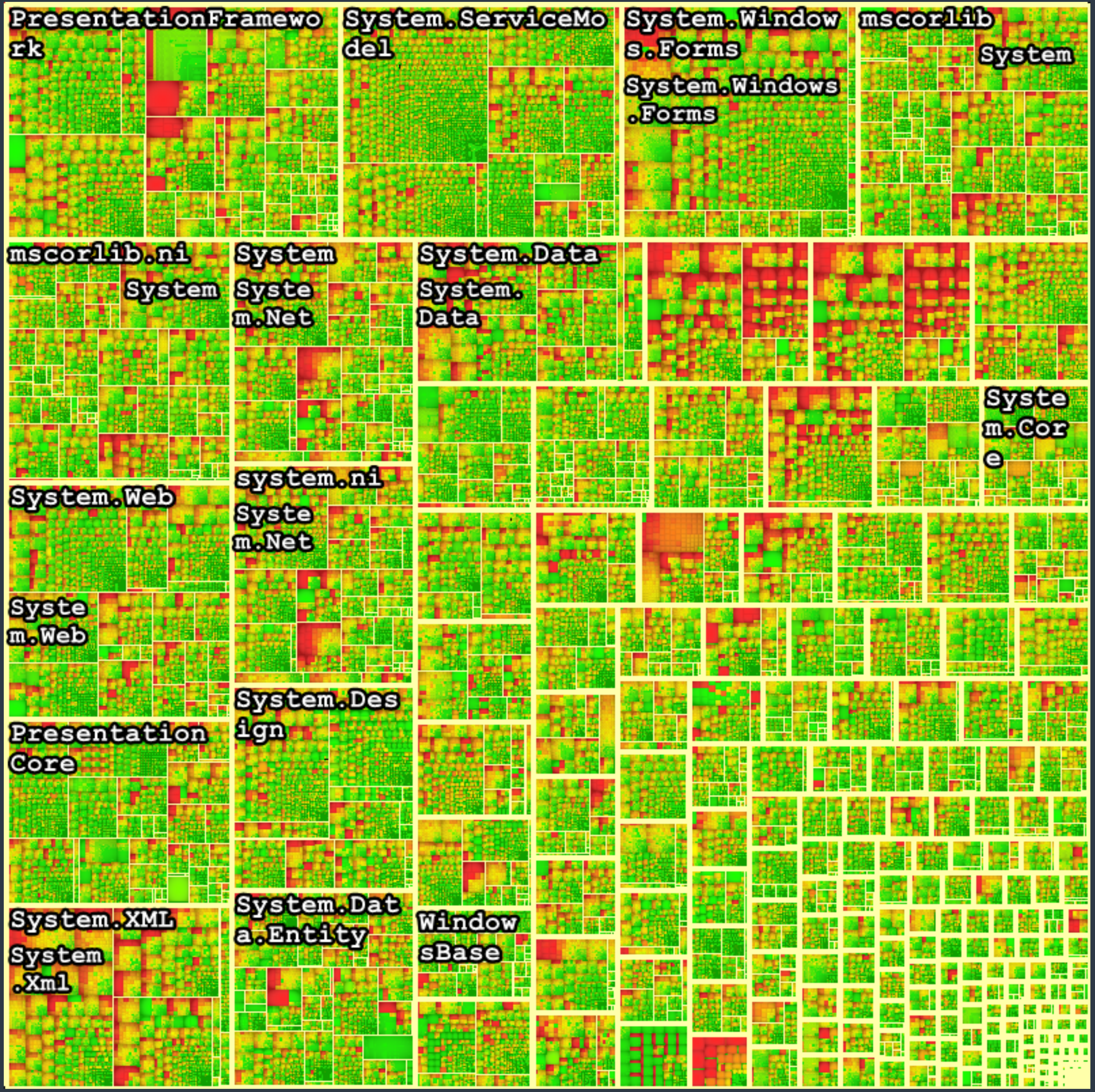
Group by: Export to Graph

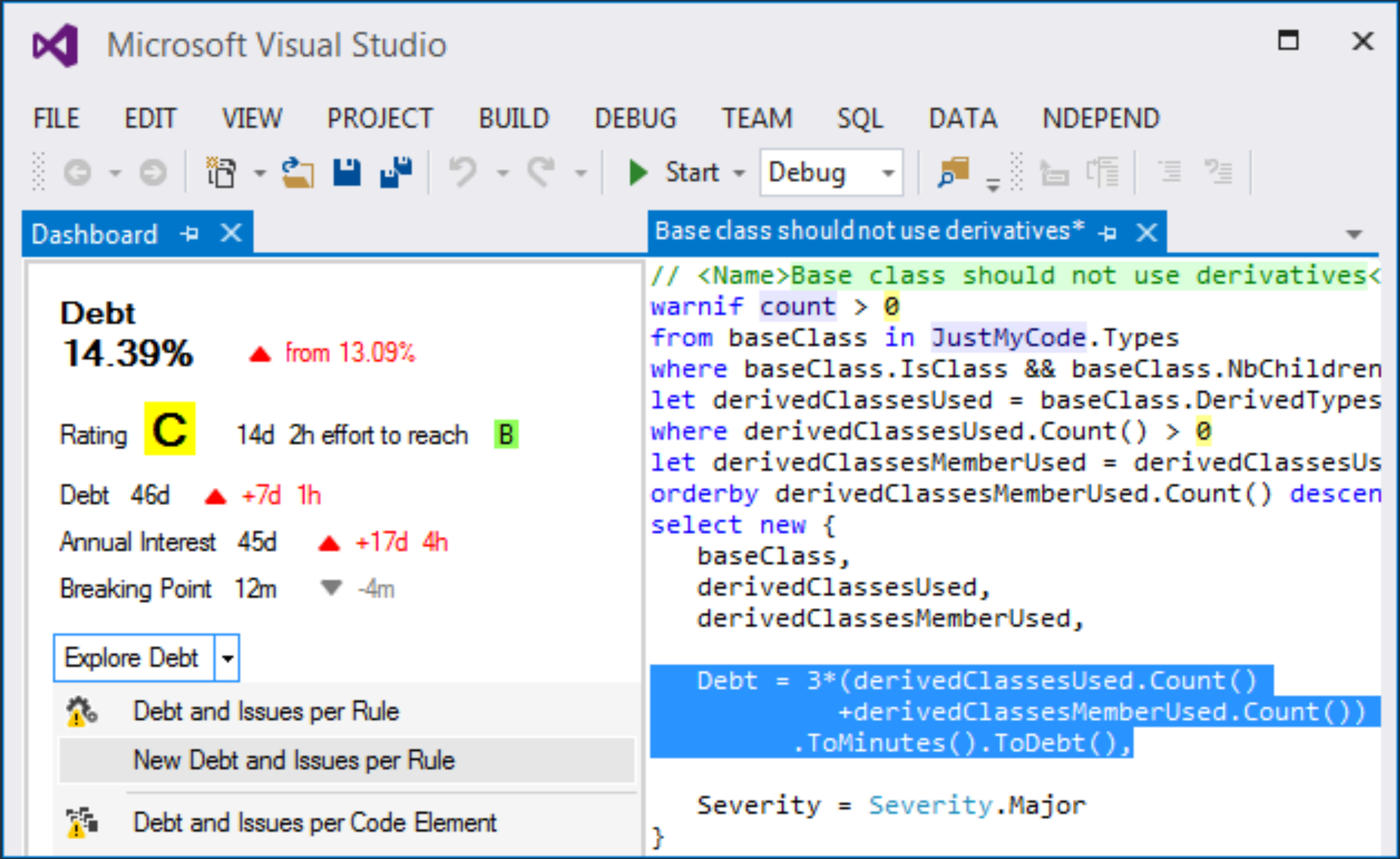
methods	IL Cyclomatic Complexity (ILCC)	IL Nesting Depth
4 methods matched		
[redacted]		
[redacted]		
[redacted]	1 099	
[redacted]	100	27
[redacted]	705	
[redacted]	41	8
[redacted]		
[redacted]		
[redacted]	330	
[redacted]	86	6
[redacted]		
[redacted]		
[redacted]	96	
[redacted]	49	15
Sum	276	56

vladikk

Join at slido.com with #testcon2019

doit-intl.com


PresentationFramework
System.ServiceModel
System.Windows.Forms
System.Windows.Forms
mscorlib
System
mscorlib.ni
System
System
System.Net
System.Data
System.Data
System.Core
system.ni
System.Net
System.Web
System.Web
System.Design
PresentationCore
System.XML
System.Xml
System.Data.Entity
WindowsBase



Debt
14.39% ▲ from 13.09%

Rating **C** 14d 2h effort to reach B

Debt 46d ▲ +7d 1h

Annual Interest 45d ▲ +17d 4h

Breaking Point 12m ▼ -4m

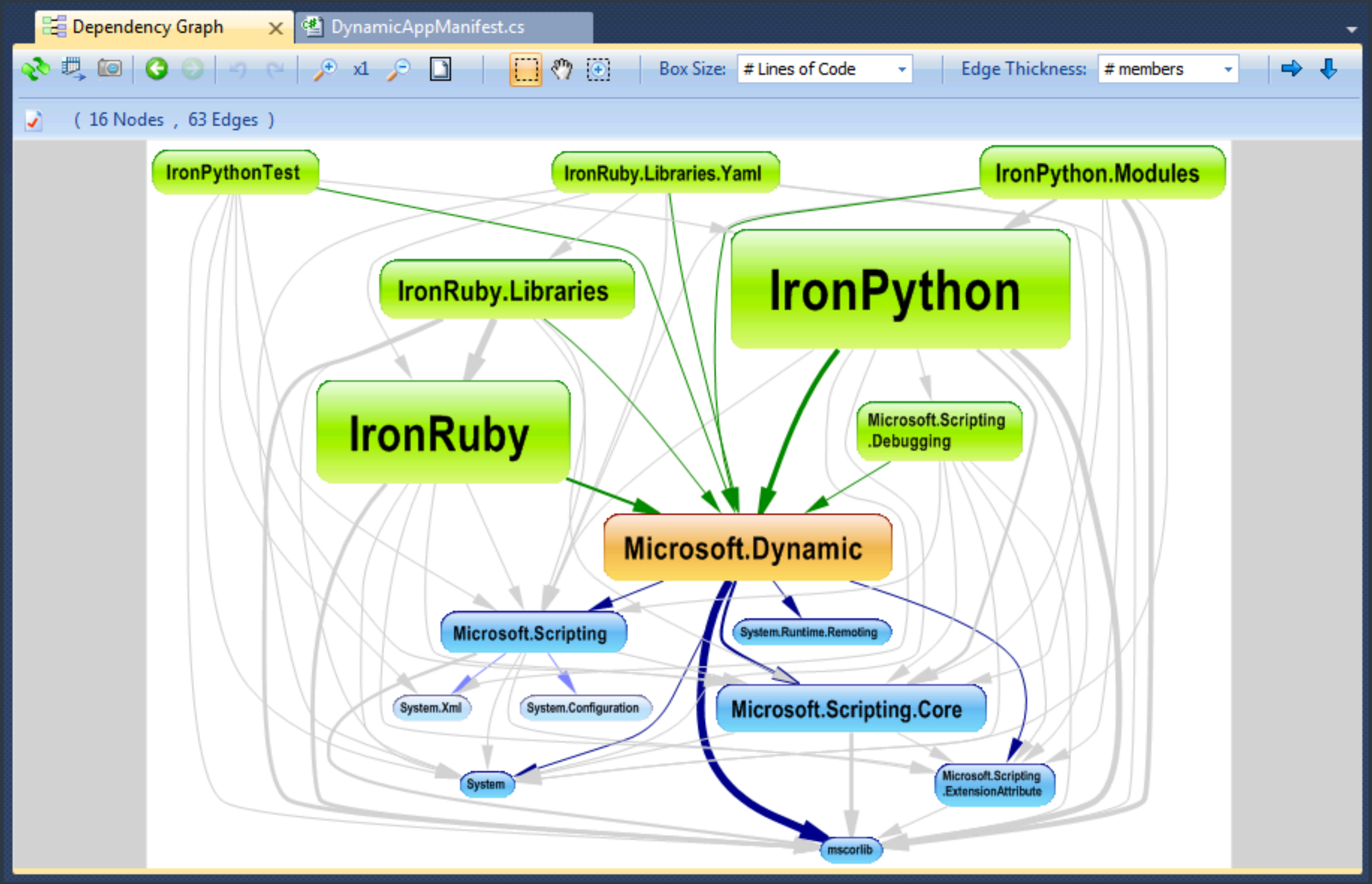
Explore Debt

- Debt and Issues per Rule
- New Debt and Issues per Rule
- Debt and Issues per Code Element

```
// <Name>Base class should not use derivatives<
warnif count > 0
from baseClass in JustMyCode.Types
where baseClass.IsClass && baseClass.NbChildren
let derivedClassesUsed = baseClass.DerivedTypes
where derivedClassesUsed.Count() > 0
let derivedClassesMemberUsed = derivedClassesUs
orderby derivedClassesMemberUsed.Count() descen
select new {
   baseClass,
   derivedClassesUsed,
   derivedClassesMemberUsed,

   Debt = 3*(derivedClassesUsed.Count()
            +derivedClassesMemberUsed.Count())
         .ToMinutes().ToDebt(),

   Severity = Severity.Major
}
```

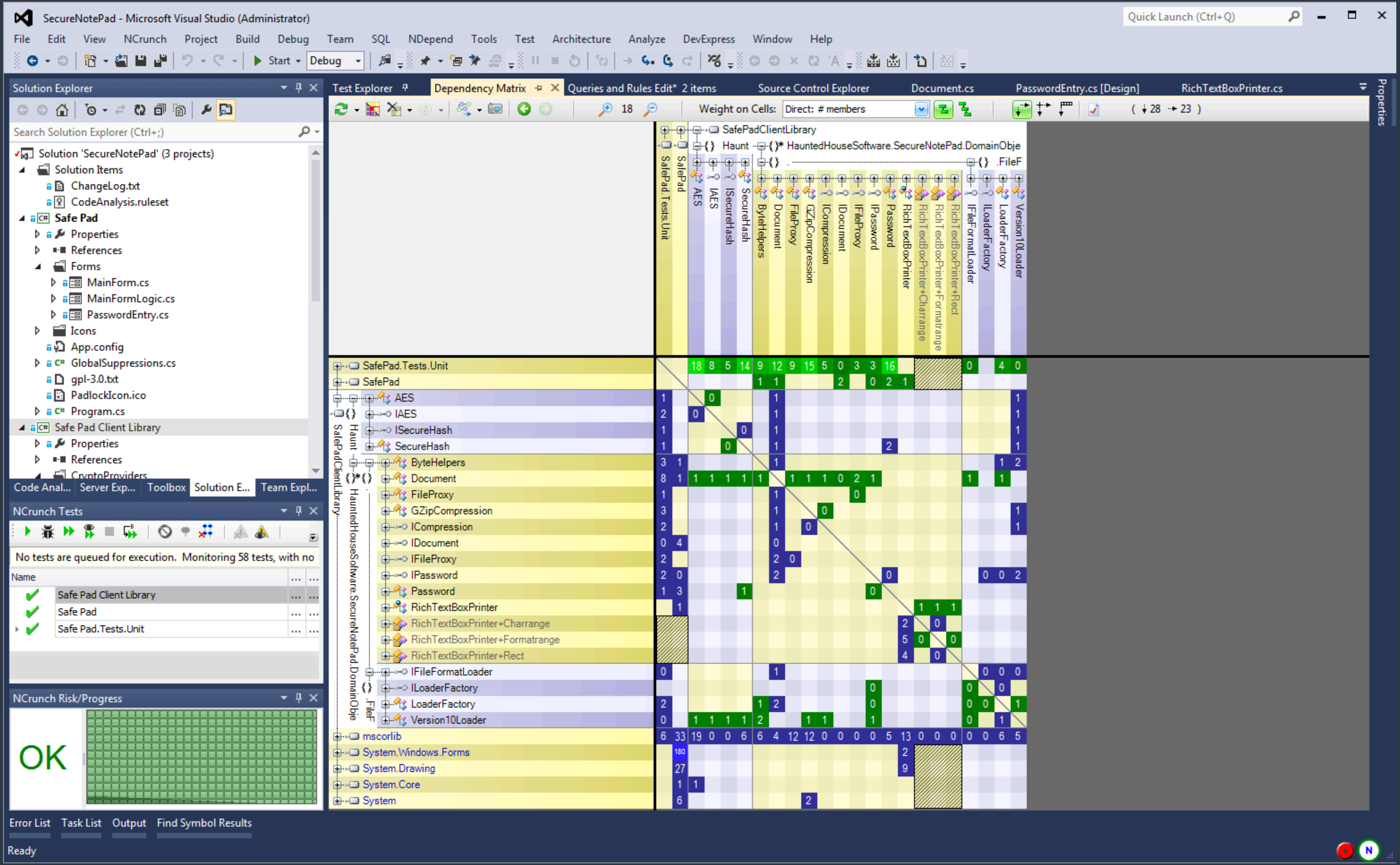


Box Size: # Lines of Code

Edge Thickness: # members

(16 Nodes , 63 Edges)

Class1 .Main(String[])
AppEntry .Main(String[])
NUnitForm .ctor (NUnitForm +CommandLineOptions)
NUnitForm .InitializeComponent()
ProjectEditor .InitializeComponent()
RecentFileMenuHandlerTests .LoadMenuItems()
RecentFileMenuHandlerTests .EnableOnLoadWhenNotEmpty()
RecentFileMenuHandler .Load()
RecentFileMenuHandler .OnRecentFileClick (Object,EventArgs)
NUnitForm.fileMenu_Popup (Object,EventArgs)
RecentFileMenuHandlerTests .DisableOnLoadWhenEmpty()
NUnitForm .openMenuItem_Click (Object,EventArgs)
NUnitForm .NUnitForm_Load (Object,EventArgs)
NUnitForm .newMenuItem_Click (Object,EventArgs)
NUnitForm .NUnitForm_Closing (Object,CancelEventArgs)
NUnitForm .closeMenuItem_Click (Object,EventArgs)
TestLoaderUI .NewProject (Form)
TestLoaderUI .CloseProject(Form)
TestLoaderUI .OpenProject (Form,Boolean)
TestLoaderUI .OpenProject (Form,String)
TestLoaderUI.OpenProject (Form,String,String)
TestLoaderAssemblyTests .UnloadTest()
TestLoaderAssemblyTests .AssemblyWithNoTests()
TestLoaderAssemblyTests .LoadTest()
TestLoaderAssemblyTests .FileNotFound()
TestLoaderAssemblyTests .InvalidAssembly()
TestLoaderAssemblyTests .LoadProject()
TestLoaderAssemblyTests .UnloadProject()
TestLoaderAssemblyTests .LoadTest(String)
TestLoaderAssemblyTests .RunTest()
TestLoaderAssemblyTests .TearDown()
TestLoader .RunLoadedTest()
TestLoader .RunTest(ITest)
TestLoader .RunTests (ITest[])
TestLoader .NewProject (String)
TestLoader .LoadProject (String)
TestLoader .set_TestProject (NUnitProject)
TestLoader .LoadProject (String,String)
TestLoader .NewProject()
TestLoader.LoadProject (String[])
TestLoader .OnProjectLoad (NUnitProject)
TestLoader .OnProjectChanged(Object ,ProjectEventArgs)
TestLoader .UnloadProject()
TestLoader .LoadTest()
TestLoader .LoadTest (String)
TestLoader .ReloadTest()
TestLoader .InstallWatcher()
TestLoader .OnTestChanged (String)
NUnitForm .addVSProjectMenuItem_Click(Object,EventArgs)
TestLoaderUI .AddVSProject(Form)
ProjectEditor .addVSProjectButton_Click (Object,EventArgs)
NUnitForm .editProjectMenuItem_Click(Object,EventArgs)
ProjectEditorTests .CreateFixtureObjects()
ProjectEditor .ctor (NUnitProject)
NUnitForm .reloadMenuItem_Click (Object,EventArgs)
ConsoleRunnerTest .XmlToConsole()
ConsoleRunnerTest .XmlResult()
ConsoleRunnerTest .FailureFixture()
ConsoleRunnerTest .Bug1073539Test()
ConsoleRunnerTest .SuccessFixture()
ConsoleRunnerTest .Bug1311644Test()
ConsoleRunnerTest .InvalidFixture()
ConsoleRunnerTest .runFixture(Type ,String[])
ConsoleRunnerTest .InvalidAssembly()
ConsoleRunnerTest .runFixture(Type)
ConsoleRunnerTest .executeConsole(String[])
Class1 .Main(String[])
ConsoleUi .Main(String[])
ConsoleUi .Execute (ConsoleOptions)
ConsoleUi .MakeRunnerFromCommandLine(ConsoleOptions)
TestDomainTests .ProjectBinPathOverridesHonored()
TestDomain .Load (NUnitProject)
TestDomain .Load (NUnitProject,String)
TestDomain .Load(String ,String)
TestDomain.Load(String ,String[],String)
TestDomain .CreateDomain (String,String,String ,String)
TestDomain .MakeRemoteTestRunner (AppDomain)
TestDomain .CreateDomain (String)
TestDomainTests .ProjectConfigFileOverridesHonored()
TestLoaderAssemblyTests .SetUp()
TestLoader .ctor()
SimpleTestRunner .get_TestFrameworks()
TestFramework .GetLoadedFrameworks()
TestLoader .ctor (TestEventDispatcher)
TestDomain .Unload()
TestExceptionHandler .Dispose()
TestDomain.MakeAppDomain (String,String,String ,String)
AssemblyResolver .Dispose()
Addins .cctor()
AssemblyResolver .ctor()
TestDomainTests .TurnOffShadowCopy()
AssemblyVersionFixture .MakeDynamicAssembly (AssemblyName,String)
TestDomainFixture .AppDomainIsSetUpCorrectly()
AssemblyVersionFixture .Version()
AssemblyVersionFixture .FindAssemblyByName (String)
TestExceptionHandler .ctor (UnhandledExceptionEventHandler)
TestExceptionHandler .Finalize()
AppDomain
SuiteBuilderTests_Multiple .LoadSuite()
SimpleTestRunner .Load (String,String[])
SimpleTestRunner .Load (String)
TestSuiteBuilder .Build (String,String[])
SimpleTestRunner .Load (String,String[],String)
SuiteBuilderTests_Multiple .LoadFixture()
TestSuiteBuilder .Build (String,String[],String)
TestAssemblyBuilder .Build(String)
TestAssemblyBuilder .Build()
TestAssemblyBuilder.Load (String)
AssemblyTests .AppSettingsLoaded()
TestRunnerThreadTests .CreateRunnerThread()
ThreadedTestRunner .BeginRun(EventListener ,String[])
TestRunnerThread .ctor (TestRunner)
NamespaceAssemblyTests .Hierarchy()
TestSuiteTreeViewFixture .SetUp()
SuiteBuilderTests .LoadAssembly()
AssemblyTests .LoadAssemblyWithoutTestFixtures()
SuiteBuilderTests .FileNotFound()
XmlTest .runSchemaValidatorTest (TextWriter,CultureInfo)
XmlTest .TestStream()
NamespaceAssemblyTests .NoNamespaceInAssembly()
XmlTest .TestSchemaValidatorFrenchCulture()
NamespaceAssemblyTests .TestRoot()
XmlTest .TestSchemaValidatorInvariantCulture()
XmlTest .runSchemaValidatorTest (String,CultureInfo)
XmlTest .TestSchemaValidatorUnitedStatesCulture()
SuiteBuilderTests .InvalidAssembly()
EventTestFixture .CheckEventListening()
XmlResultVisitorTest .RunMockTests()
AssemblyTests .LoadAssembly()
AssemblyTests .LoadAssemblyNotFound()
ProgressBarTests .Setup()
StatusBarTests .Setup()
TestFixtureBuilderTests .InvalidSignatureTest (String,String)
TestFixtureBuilderTests .LoadCategories()
TestFixtureBuilderTests .GoodSignature()
NamespaceAssemblyTests .LoadTestFixtureFromAssembly()
SimpleTestRunner .Load (String,String)
TestSuiteBuilder .Build (String)
SuiteBuilderTests .LoadSuite()
SuiteBuilderTests .WrongReturnTypeSuite()
TestFixtureExtension .LoadFixture()
TestFixtureBuilderTests .loadFixture(Type)
SuiteBuilderTests .DiscoverSuite()
AssemblyTests .LoadTestFixtureFromAssembly()
SuiteBuilderTests .LoadFixture()
SuiteBuilderTests .LoadNamespaceAsSuite()
SuiteBuilderTests .FixtureNotFound()
TestSuiteBuilder .Build (String,String)



SecureNotePad - Microsoft Visual Studio (Administrator)

Solution 'SecureNotePad' (3 projects)
- Solution Items
  - ChangeLog.txt
  - CodeAnalysis.ruleset
- Safe Pad
  - Properties
  - References
  - Forms
    - MainForm.cs
    - MainFormLogic.cs
    - PasswordEntry.cs
  - Icons
  - App.config
  - GlobalSuppressions.cs
  - gpl-3.0.txt
  - PadlockIcon.ico
  - Program.cs
- Safe Pad Client Library
  - Properties
  - References
  - CryptoProviders

NCrunch Tests

No tests are queued for execution. Monitoring 58 tests, with no

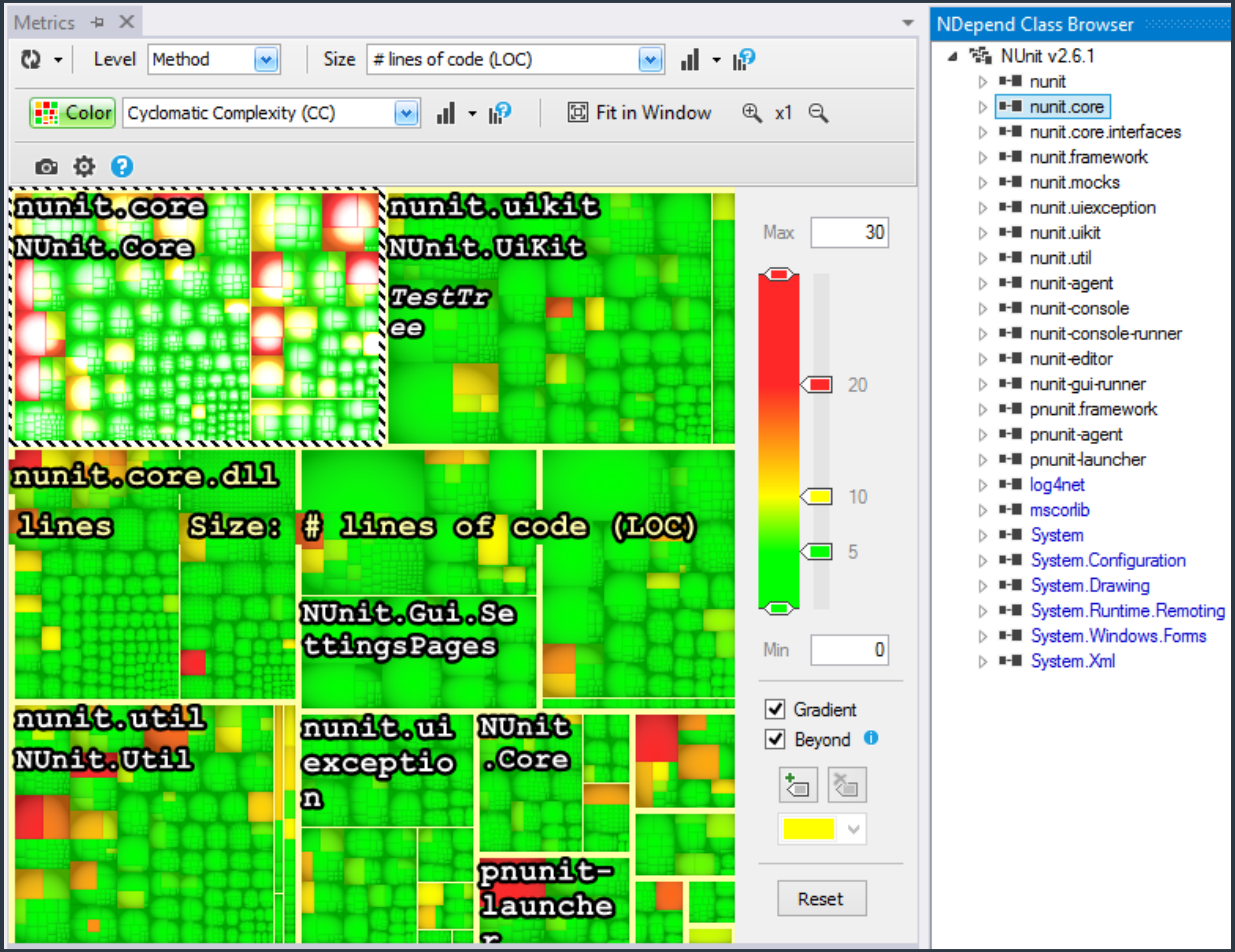
Name
✓ Safe Pad Client Library
✓ Safe Pad
✓ Safe Pad.Tests.Unit

NCrunch Risk/Progress: OK

Weight on Cells: Direct: # members

Dependency Matrix rows: SafePad.Tests.Unit, SafePad, AES, IAES, ISecureHash, SecureHash, ByteHelpers, Document, FileProxy, GZipCompression, ICompression, IDocument, IFileProxy, IPassword, Password, RichTextBoxPrinter, RichTextBoxPrinter+Charrange, RichTextBoxPrinter+Formatrange, RichTextBoxPrinter+Rect, IFileFormatLoader, ILoaderFactory, LoaderFactory, Version10Loader, mscorlib, System.Windows.Forms, System.Drawing, System.Core, System

Ready



Metrics

Level: Method

Size: # lines of code (LOC)

Color: Cyclomatic Complexity (CC)

Fit in Window x1

Max: 30

Min: 0

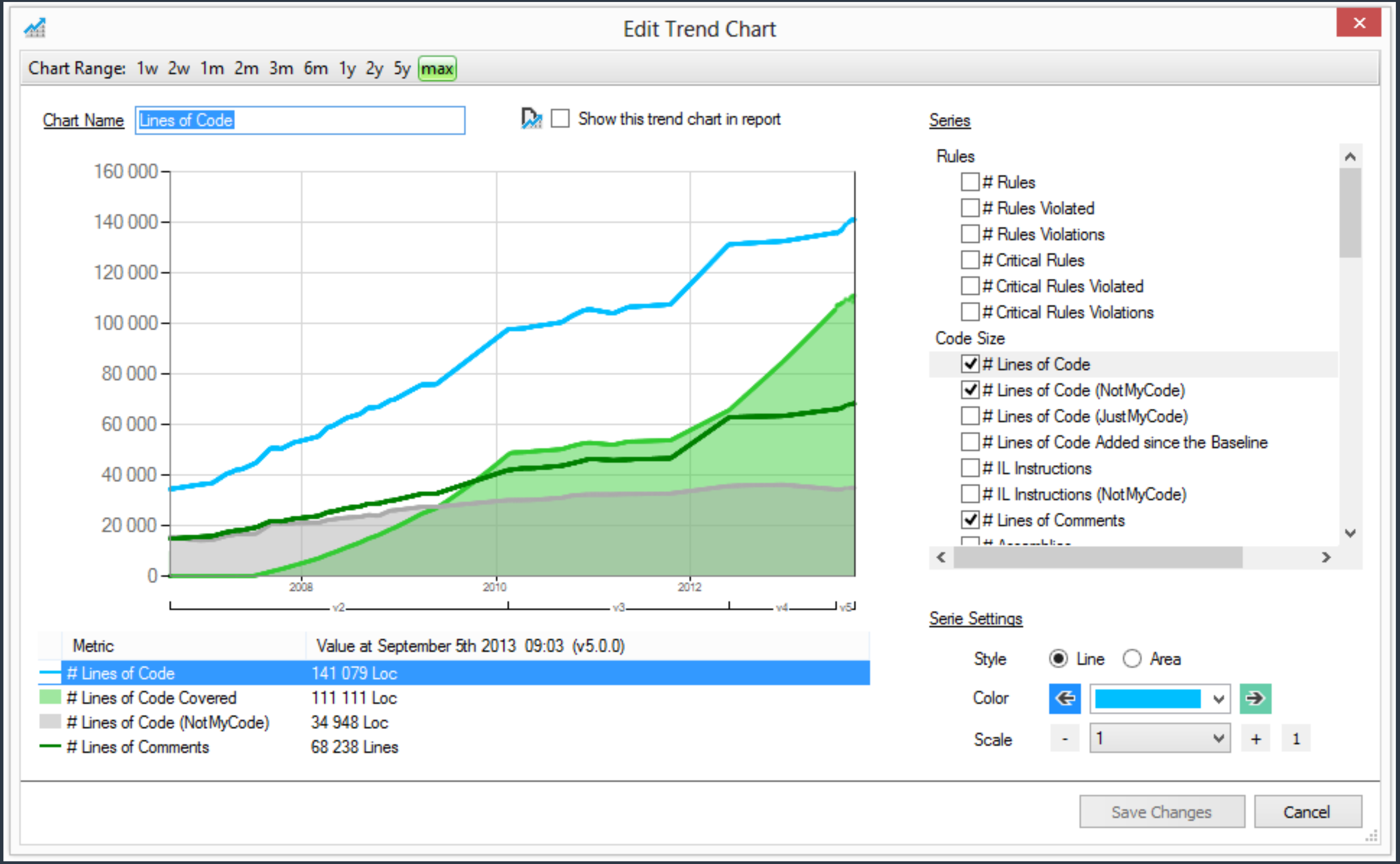
☑ Gradient

☑ Beyond

Reset

NDepend Class Browser

- NUnit v2.6.1
  - nunit
  - nunit.core
  - nunit.core.interfaces
  - nunit.framework
  - nunit.mocks
  - nunit.uiexception
  - nunit.uikit
  - nunit.util
  - nunit-agent
  - nunit-console
  - nunit-console-runner
  - nunit-editor
  - nunit-gui-runner
  - pnunit.framework
  - pnunit-agent
  - pnunit-launcher
  - log4net
  - mscorlib
  - System
  - System.Configuration
  - System.Drawing
  - System.Runtime.Remoting
  - System.Windows.Forms
  - System.Xml



Edit Trend Chart

Chart Range: 1w 2w 1m 2m 3m 6m 1y 2y 5y [✓ max]

Chart Name: Lines of Code

□ Show this trend chart in report

	Metric	Value at September 5th 2013 09:03 (v5.0.0)
━	# Lines of Code	141 079 Loc
🟩	# Lines of Code Covered	111 111 Loc
⬜	# Lines of Code (NotMyCode)	34 948 Loc
━	# Lines of Comments	68 238 Lines

Series

Rules
- □ # Rules
- □ # Rules Violated
- □ # Rules Violations
- □ # Critical Rules
- □ # Critical Rules Violated
- □ # Critical Rules Violations

Code Size
- ☑ # Lines of Code
- ☑ # Lines of Code (NotMyCode)
- □ # Lines of Code (JustMyCode)
- □ # Lines of Code Added since the Baseline
- □ # IL Instructions
- □ # IL Instructions (NotMyCode)
- ☑ # Lines of Comments
- □ # Assemblies

Serie Settings

Style ● Line ○ Area

Color

Scale - 1 + 1

EVOLUTION DRIVERS

ORGANIZATIONAL CHANGE DRIVERS

SMALL:

AD-HOC INTEGRATION

→

LARGE:

FORMAL INTEGRATION

BUSINESS STRATEGY CHANGE DRIVERS

SUPPORTING → CORE BUSINESS

WRAP UP

Software architecture manages complexity

Local and global complexities

Everyone suffers if module boundaries are ineffective

Look for symptoms of bad boundaries. And react!

Use static code analysis tools to control codebase evolution

Look for chages in the organization structure and business strategies that affect software design

Software architecture manages complexity

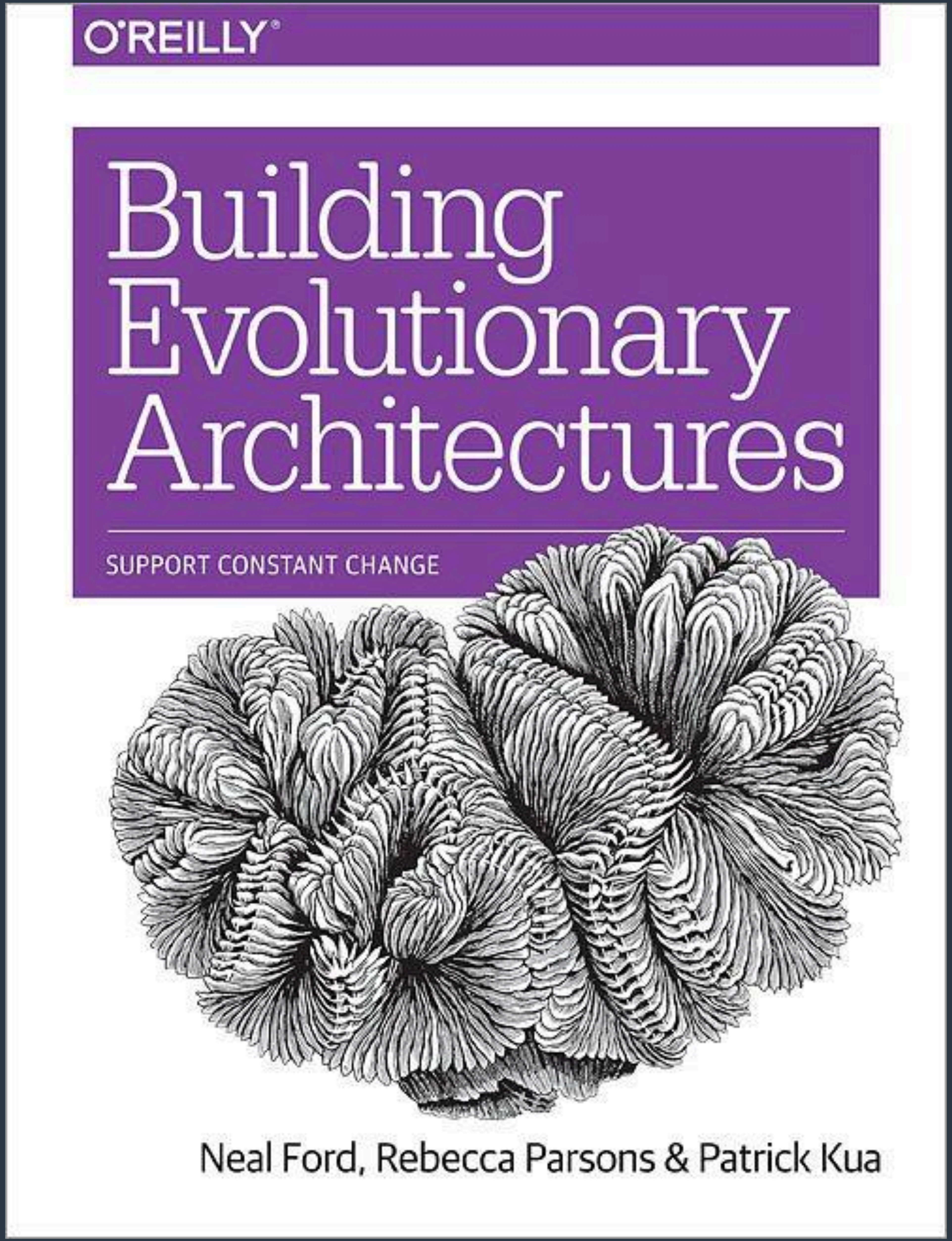
Local and global complexities

Everyone suffers if module boundaries are ineffective

Look for symptoms of bad boundaries. And react!

Use static code analysis tools to control codebase evolution

Look for chages in the organization structure and business strategies that affect software design

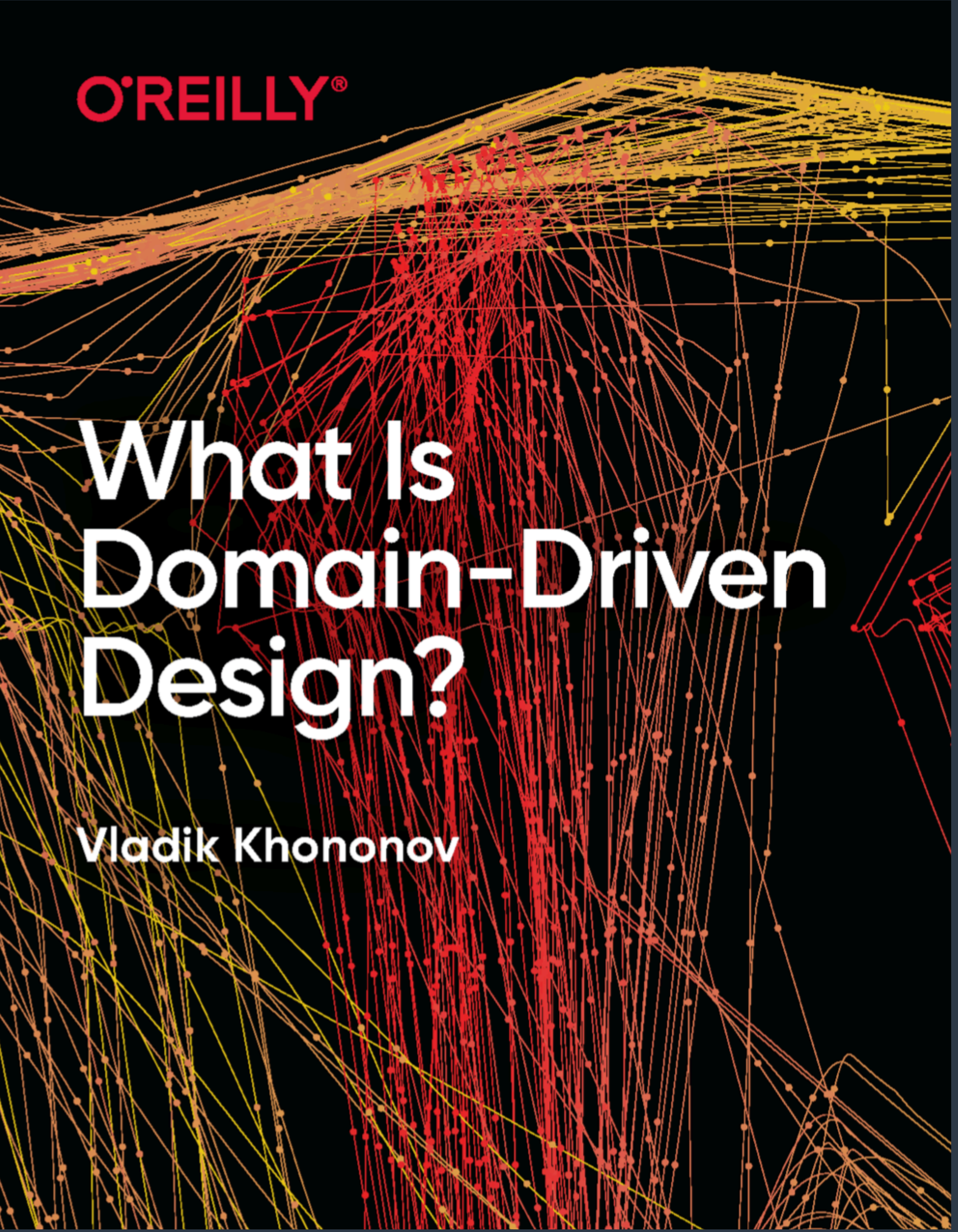


O'REILLY

Building Evolutionary Architectures

SUPPORT CONSTANT CHANGE

Neal Ford, Rebecca Parsons & Patrick Kua



O'REILLY®

What Is Domain-Driven Design?

Vladik Khononov

https://bit.ly/2ptJ5fS

THANK YOU!

🐦 @vladikk

⌨ vladikk.com

DOMAIN DRIVEN DESIGN

The First 15 Years

Essays from the DDD Community

Sponsored by
Domain-Driven Design Europe

https://bit.ly/31nSD9c

http://careers.doit-intl.com