

How to leverage static code analysis in your CI/CD pipelines for continuous code quality

Dana Epp

Microsoft Regional Director

<https://danaepp.com>







When the code is incorrect, you can't really talk about security. When the code is faulty, it cannot be safe.



- Gene 'Spaf' Spafford



**Quality is not an act,
it is a habit.**

- Aristotle

“

If you can't champion code
quality with your team, how
can you ever champion secure
code?

”

- Dana Epp

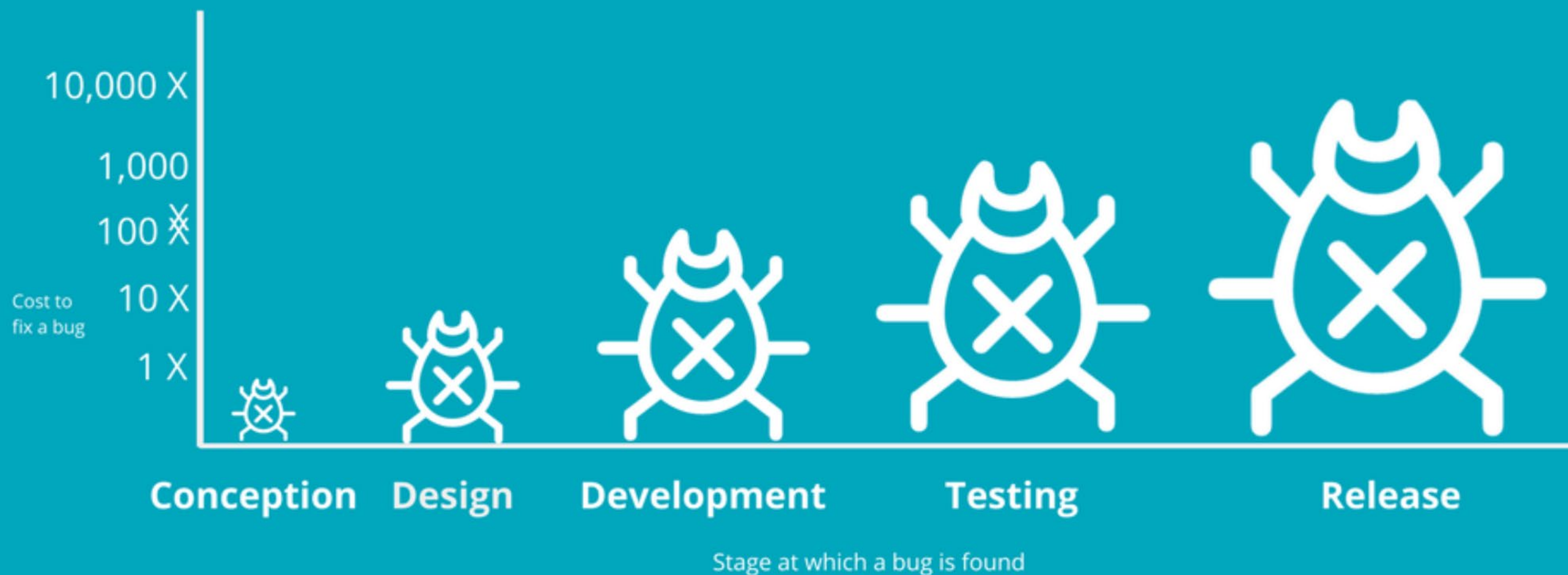


Passing static code analysis
doesn't prove your code is
safe... but failing it pretty much
signals it isn't.



- Dana Epp

Resolving bugs early and often reduces associated costs



WHY IS THAT?



Most studies show that inspection is cheaper than testing. [We] found that code reading detected 80% more faults per hour than testing.



- Basili and Selby 1987

Comparing defect detection approaches

Table 20-2 Defect-Detection Rates

Removal Step	Lowest Rate	Modal Rate	Highest Rate
Informal design reviews	25%	35%	40%
Formal design inspections	45%	55%	65%
Informal code reviews	20%	25%	35%
Formal code inspections	45%	60%	70%
Modeling or prototyping	35%	65%	80%
Personal desk-checking of code	20%	40%	60%
Unit test	15%	30%	50%
New function (component) test	20%	30%	35%
Integration test	25%	35%	40%
Regression test	15%	25%	30%
System test	25%	40%	55%
Low-volume beta test (<10 sites)	25%	35%	40%
High-volume beta test (>1,000 sites)	60%	75%	85%

Source: Adapted from *Programming Productivity* (Jones 1986a), "Software Defect-Removal Efficiency" (Jones 1996), and "What We Have Learned About Fighting Defects" (Shull et al. 2002).

**What can
static code analysis
do for me??**

Know the quality of your code at all times


```
246 if (Provider.class == roleTypeClass) {
247     Type providedType = ReflectionUtils.getLastTypeGenericArgument(dependen
248     2 Class providedClass = 1 ReflectionUtils.getTypeClass(providedType);
249
250     if (this.componentManager.hasComponent(providedType, dependencyDescript
251         || 3 providedClass.isAssignableFrom(List.class) || providedClass.
```

A "NullPointerException" could be thrown; "providedClass" is nullable here.

 Bug  Major 

```
252         continue;
253     }
```

Reliability

 Bugs 2  **1** 

Security

 Security Vulnerabilities 0  **0** 

 Security Hotspots 39 - **0** -

Maintainability

 Technical Debt 6 days  **0** 

 Code Smells 319 - **0** -

New code Since last release

1 

0 

0 -

0 

0 -

Detect bugs

A "NullPointerException" could be thrown; "getFilter()" can return null.

🐛 Bug ⬆️ Major

+4

1 Implies 'filter' can be null.

2 'Exception' is caught.

3 'getFilter()' can return null.

4 Result of 'getFilter()' is dereferenced.

activemq-broker/.../broker/interceptor/pack...

Insert a <!DOCTYPE> declaration to before this <html> tag.

🐛 Bug ⬆️ Major

Add a 'favicon' declaration in this 'header' tag.

🐛 Bug ⬆️ Major

105

106

```
public void injectMessage(ProducerBrokerExchange producerExchange,
4 getFilter().injectMessage(producerExchange, messageSend);
```

A "NullPointerException" could be thrown; "getFilter()" can return null. ...

🐛 Bug ⬆️ Major 🔓 Open ▾ Not assigned ▾ 10min effort Comment

107

```
}
```

108

109

110

```
private MessageInterceptorFilter getFilter() {
```

111

```
if (1 filter == null) {
```

112

```
try {
```

113

```
MutableBrokerFilter mutableBrokerFilter = (MutableBroke
```

114

```
Broker next = mutableBrokerFilter.getNext();
```

115

```
filter = new MessageInterceptorFilter(next);
```

116

```
mutableBrokerFilter.setNext(filter);
```

117

```
} catch (2 Exception e) {
```

118

```
LOG.error("Failed to create MessageInterceptorFilter",
```

119

```
}
```

Detect 'code smells'

```
namespace AdWorks.MVC.Controllers
{
    public class HomeController : Controller
    {
        public IActionResult Index()
        {
            dynamic obj = "hello";
        }
    }
}
```

Remove this useless assignment to local variable 'obj'. ...

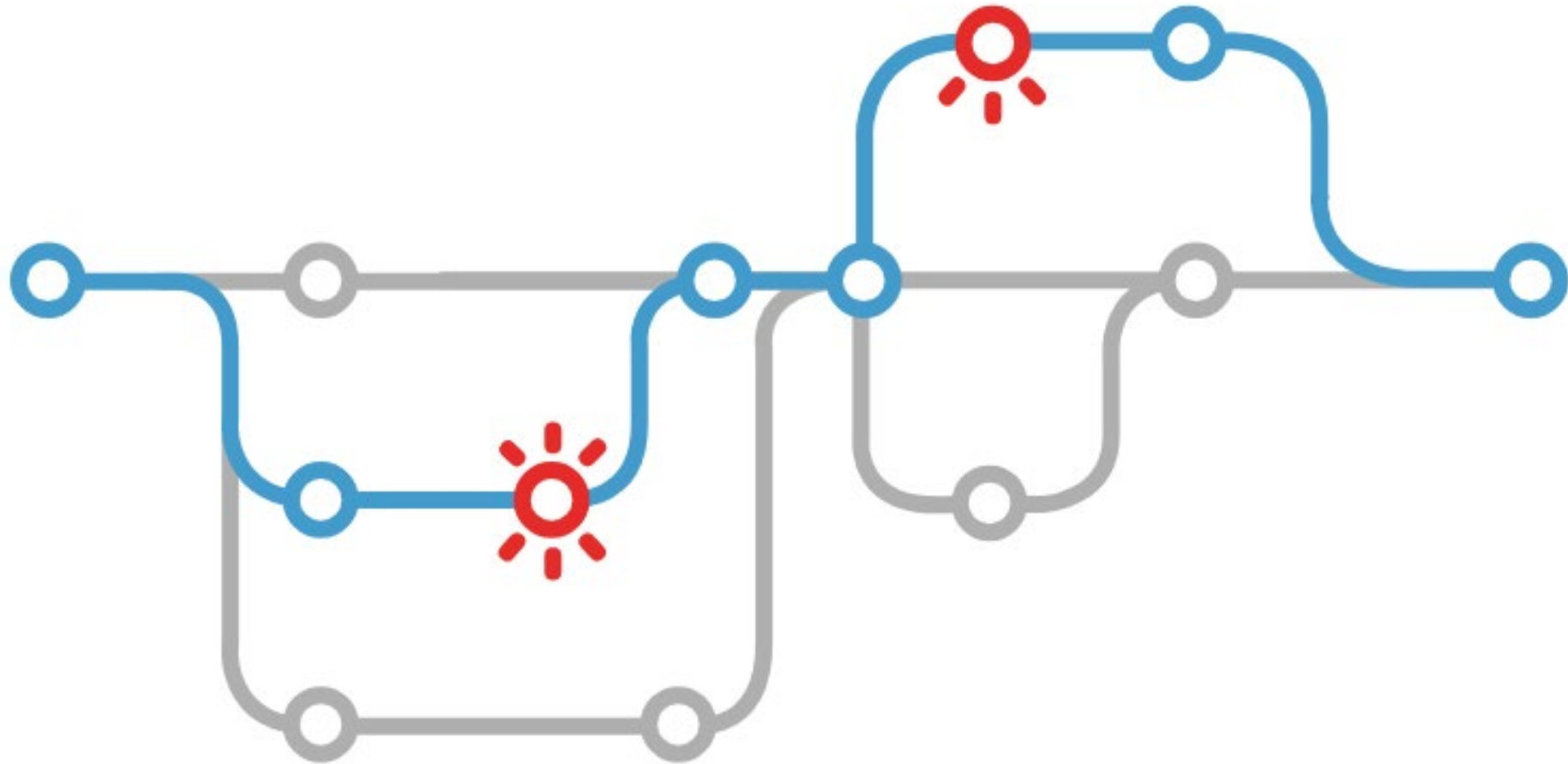
2 months ago ▾ L15 

 Code Smell  Major  Open ▾ Not assigned ▾ 15min effort [Comment](#)

 cert, cwe, unused ▾

```
obj = new { name = "fred" };
obj = 10;
```

Explore more execution paths



Discover cognitive complexity issues

Newtonsoft.Json/Bson/BsonReader.cs

Refactor this method to reduce its Cognitive Complexity from 19 to the 15 allowed.

Code Smell +10

- +1
- +2 (incl 1 for nesting)
- +1
- +2 (incl 1 for nesting)
- +1
- +1
- +3 (incl 2 for nesting)
- +3 (incl 2 for nesting)
- +1
- +4 (incl 3 for nesting)

alt + ↑ ↓ to navigate issue locations

1 of 1 shown

```
589 ... // used in case of left over multibyte characters in the buffer
590 ... int offset = 0;
591 ... 1 while (true)
592 ... {
593 ...     int count = offset;
594 ...     byte b;
595 ...     2 while (count < MaxCharBytesSize 3 && (b = _reader.ReadByte()) > 0)
596 ...     {
597 ...         _byteBuffer[count++] = b;
598 ...     }
599 ...     int byteCount = count - offset;
600 ...     totalBytesRead += byteCount;
601 ...
602 ... 4 if (count < MaxCharBytesSize 5 && builder == null)
603 ...     {
604 ...         // pref optimization to avoid reading into a string builder
```

```
609 ...     return new string(_charBuffer, 0, length);
610 ... }
611 ... 6 else
612 ... {
613 ...     // calculate the index of the end of the last full character in the buffer
614 ...     int lastFullCharStop = GetLastFullCharStop(count - 1);
615 ...
616 ...     int charCount = Encoding.UTF8.GetChars(_byteBuffer, 0, lastFullCharStop + 1, _charBuffer, 0);
617 ...
618 ...     7 if (builder == null)
619 ...     {
620 ...         builder = new StringBuilder(MaxCharBytesSize * 2);
621 ...     }
622 ...
623 ...     builder.Append(_charBuffer, 0, charCount);
624 ...
625 ...     8 if (lastFullCharStop < byteCount - 1)
```


Find security vulnerabilities

```
// dumpObj:  
dumpObj: function( spec ) {  
  var val = "<undefined>";  
  try {  
    val = eval( "this."+spec ).toString();  
  }  
}
```

Review the arguments of this "eval" call to make sure they are validated. 

7 months ago ▼ L989 

 Vulnerability  Critical  Open ▼ Not assigned ▼ 30min effort [Comment](#)

 [cwe, owasp-a3](#) ▼



```
  } catch( exception ) {  
  }  
  this.dump( spec + "=" + val + "\n" );  
},
```


Review security 'hotspots'

```
95     /**
96     * Sets the maximum age of the cookie in seconds.
97     */
98     public CookieBuilder setExpiry(int expiry) {
99         this.expiry = expiry;
100        return this;
101    }
102
103     public Cookie build() {
104        Cookie cookie = new Cookie(requireNonNull(name), value);
```

Make sure that this cookie is used safely. [See Rule](#)

10 months ago ▾ L104 🔗

 Security Hotspot To Review  Julien Lancelot

 cert, cwe, owasp-a3

```
105        cookie.setPath(getContextPath(request));
106        cookie.setSecure(isHttps(request));
107        cookie.setHttpOnly(httpOnly);
108        cookie.setMaxAge(expiry);
109        return cookie;
110    }
111
```

Enforce security best practices

```
434
435 public static KeyPair generateRsaOrDsa(boolean rsa) throws Exception {
436     if (rsa) {
437         KeyPairGenerator keyPairGen =
438             KeyPairGenerator.getInstance("RSA");
439         keyPairGen.initialize(1024);
```

Use a key length of at least 2048 bits. ...

10 years ago ▾ L439 🔗

 Vulnerability ▾  Blocker ▾  Open ▾ Not assigned ▾ 2min effort [Comment](#)

 cwe, owasp-a3 ▾

```
440
441     RSAKeyGenParameterSpec keySpec = new RSAKeyGenParameterSpec(1024,
442         RSAKeyGenParameterSpec.F0);
443     keyPairGen.initialize(keySpec);
444
445     KeyPair rsaKeyPair = keyPairGen.generateKeyPair();
446
447     return rsaKeyPair;
448 } else {
```

Untrusted input analysis (taint analysis)

```
12 ...
13 [HttpGet("{id}", Name = "Get")]
14 public string 1 GetThing( 2 string id)
15 {
16     var connection = new SqlConnection();
17     try
18     {
19         connection.ConnectionString = "db info";
20         connection.Open();
21         var 4 selectSql = 3 string.Format("select from MyStuff where id='{0}';", id);
22         var selectCommand = new 5 SqlCommand(selectSql, connection);
```

Refactor this code to not construct SQL queries directly from tainted user-controlled data. [See Rule](#)

last month ▾ L22 🔗

🔒 Vulnerability 🚫 Blocker 🔵 Open ▾ 👤 Jeff Zapotoczny ▾ 30min effort Comment

🔍 cert, cwe, owasp-a1, sans-top25-inse... ▾

```
23
24     var dataReader = selectCommand.ExecuteReader();
25     return dataReader.GetString(0);
26 }
27 catch (Exception ex)
28 {
29 }
30 finally
31 {
```

OWASP / SANS security reports

Security Reports
Track the Vulnerabilities and Security Hotspots in your Portfolio.

Overall security metrics

- Security Vulnerabilities **A**
- Security Review **A**

SonarSource | **OWASP Top 10** | SANS Top 25

Vulnerabilities and Security Hotspots conforming to the OWASP Top 10 standard

Categories	Security Vulnerabilities	Security Hotspots	
		To Review	In Review
A1 - Injection	2 D	302	0
A2 - Broken Authentication	0 A	0	0
A3 - Sensitive Data Exposure	1 C	133	0
A4 - XML External Entities (XXE)	0 A	0	0
A5 - Broken Access Control	0 A	0	0
A6 - Security Misconfiguration	3 C	31	0
A7 - Cross-Site Scripting (XSS)	0 A	3	0
A8 - Insecure Deserialization	0 A	2	0
A9 - Using Components with Known Vulnerabilities	0 A	0	0
A10 - Insufficient Logging & Monitoring	0 A	17	0

- Requires SonarQube Enterprise

OWASP / SANS security reports

▼ Security Category	
▼ SonarSource	
Others	2.4k
Insecure Configuration	20
Weak Cryptography	1
▼ OWASP Top 10	
A3 - Sensitive Data Exposure	1
A6 - Security Misconfiguration	1
▼ SANS Top 25	
Porous Defenses	1

▼ CWE	
<input type="text" value="Search for CWEs..."/>	
No CWE associated	2.2k
CWE-546 - Suspicious Comment	81
CWE-563 - Assignment to Variable ...	60
CWE-397 - Declaration of Throws f...	26
CWE-489 - Leftover Debug Code	20
CWE-570 - Expression is Always F...	20
CWE-571 - Expression is Always True	20
CWE-476 - NULL Pointer Dereference	19
CWE-493 - Critical Public Variable ...	19
CWE-327 - Use of a Broken or Risky...	1
CWE-328 - Reversible One-Way Hash	1
CWE-759 - Use of a One-Way Hash ...	1
CWE-760 - Use of a One-Way Hash ...	1
CWE-916 - Use of Password Hash ...	1

- Available in SonarCloud



AUDITWOLF

Our DevOps toolchain



Azure Boards



Azure Repos



Azure Pipelines



Azure Artifacts

sonarlint

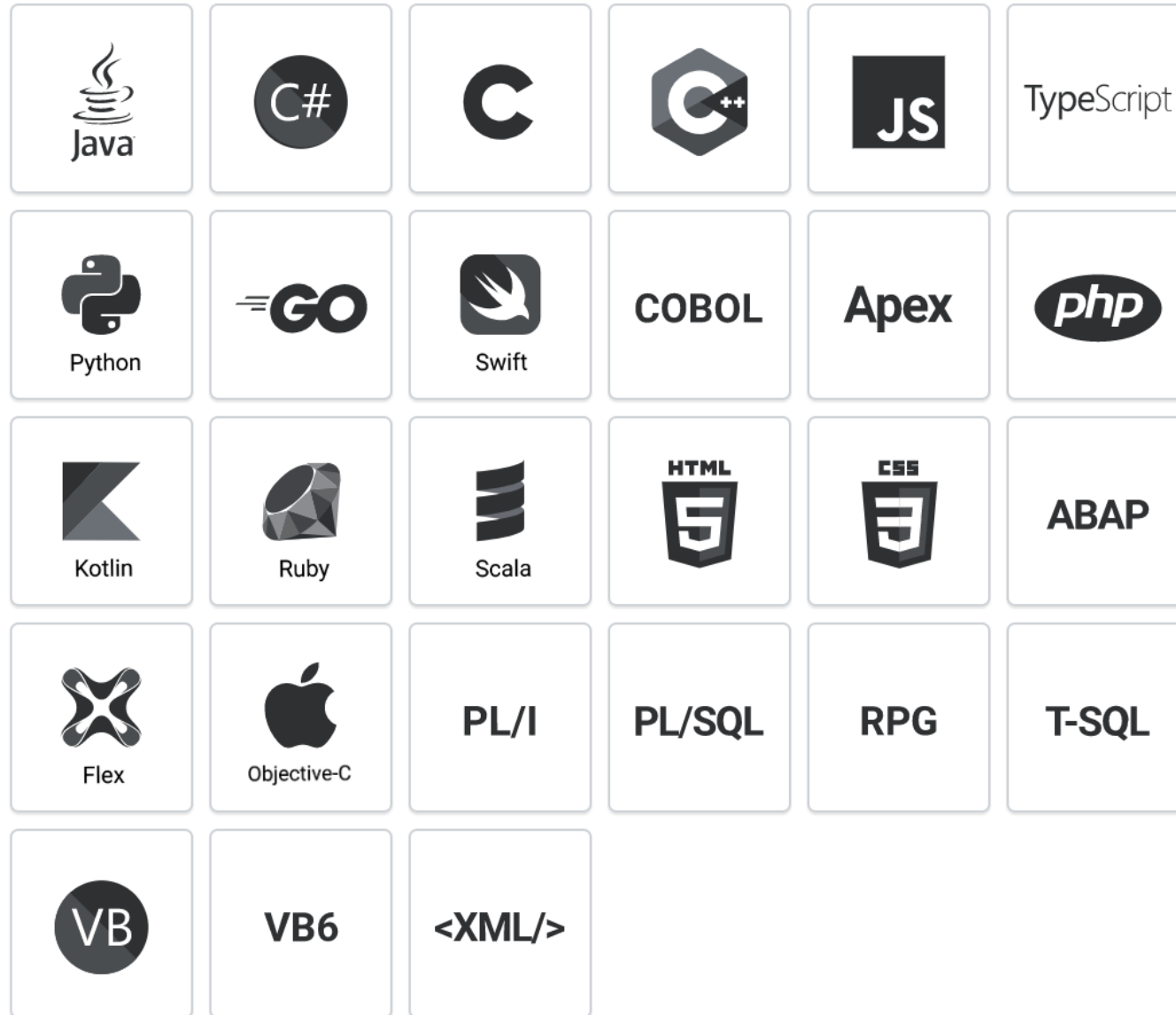

sonarcloud 

Our Stack

- Typescript code targeting NodeJS deployed to Web App for Containers
- C++ code targeting Linux shell deployed to Azure Container Instances
- C# code targeting .NET Core 3.1 deployed to Azure Container Instances
- C# code targeting .NET Core 2.1 deployed to Azure Functions
- Typescript code targeting Angular 8 deployed to Azure CDN / Frontdoor



Languages SonarQube supports



We start with SonarLint – Democratize quality

JS memory-stats.js ●

```
35 |     msGraph.appendChild( bar );
36 |
37 | }
38 |
39 | var updateGraph = function ( dom, height, color ) {
40 |
41 |     var child = dom.appendChild( dom.firstChild );
42 |     child.style.height = height + 'px';
43 |     if( color ) child.style.backgroundColor = color;
44 |
45 | }
46 |
47 | var perf = window.performance || {};
48 | // polyfill usedJSHeapSize
49 | if (!perf.memory){
50 |     perf.memory = { usedJSHeapSize : 0 };
51 | }
52 |
53 | // support of the API?
54 | if( perf.memory.totalJSHeapSize === 0 ){
55 |     console.warn('totalJSHeapSize === 0... performance.memory is only available in Chrome .')
56 | }
57 |
58 | // TODO, add a sanity check to see if values are bucketed.
59 | // If so, remind user to adopt the --enable-precise-memory-info flag.
```



We enforce peer code review before merge

Azure DevOps

wildrook / AuditWolf / Repos / Branches / Scanner

AuditWolf +

- Overview
- Boards
- Repos**
- Files
- Commits
- Pushes
- Branches**
- Tags
- Pull requests
- Pipelines
- Test Plans
- Artifacts

Branch policies for Develop

Save changes Discard changes

Protect this branch

- Setting a Required policy will enforce the use of pull requests when updating the branch
- Setting a Required policy will prevent branch deletion
- Manage permissions for this branch on the [Security page](#)

Require a minimum number of reviewers
Require approval from a specified number of reviewers on pull requests.

Minimum number of reviewers

Requestors can approve their own changes

Allow completion even if some reviewers vote to wait or reject

Reset code reviewer votes when there are new changes

Check for linked work items
Encourage traceability by checking for linked work items on pull requests.

Policy requirement

Required
Block pull requests from being completed unless they have at least one linked work item.

Optional
Warn if there are no linked work items, but allow pull requests to be completed.

Check for comment resolution
Check to see that all comments have been resolved on pull requests.

Require at least one other code reviewer

Don't allow requestor to approve their own work

Require all code to be linked to work on the board

Merge triggers build pipeline

Inject static code analysis agent into build environment, configured to your project in SonarCloud

Execute static code analysis

Report results to SonarCloud

The screenshot displays a build pipeline configuration in Azure DevOps. The pipeline is titled "Pipeline" and "Build pipeline". It is triggered by "Webhooks" and runs on the "develop" branch. The pipeline consists of a single phase, "Phase 1", which runs on an agent. The steps in the pipeline are:

- Prepare analysis on SonarCloud** (Prepare Analysis Configuration) - This step is highlighted in blue and has a checkmark icon, indicating it is the current step or a completed step. It is associated with the SonarCloud icon.
- Use NuGet 4.4.1** (NuGet tool installer)
- NuGet restore** (NuGet)
- Build solution** (Visual Studio build)
- Test Assemblies** (Visual Studio Test)
- Run Code Analysis** (Run Code Analysis) - This step is associated with the SonarCloud icon.
- Publish Quality Gate Result** (Publish Quality Gate Result) - This step is associated with the SonarCloud icon.
- Publish symbols path** (Index sources and publish symbols)
- Publish Artifact** (Publish build artifacts)

Blue arrows from the text on the left point to the corresponding steps in the pipeline: "Inject static code analysis agent into build environment, configured to your project in SonarCloud" points to "Prepare analysis on SonarCloud"; "Execute static code analysis" points to "Run Code Analysis"; and "Report results to SonarCloud" points to "Publish Quality Gate Result".

Build success triggers release pipeline

Enable Deployment Gates



Quality Gate enforcement



Pre-deployment conditions
GraphCollector to QA

Triggers ▼
Define the trigger that will start deployment to this stage

Pre-deployment approvals ⏻ Disabled
Select the users who can approve or reject deployments to this stage

Gates ^ ⓘ Enabled
Define gates to evaluate before the deployment. [Learn more](#)

The delay before evaluation ⓘ

▼

Deployment gates ⓘ + Add ▼

Check SonarCloud Quality Gate status ⓘ Enabled

SonarCloud Quality Gate status check (Preview) ⓘ

Task version ▼

Display name *

Output Variables ▼

More information / links

Tools

- Azure DevOps : <https://dev.azure.com>
- SonarLint : <https://www.sonarlint.org/>
- SonarQube : <https://www.sonarqube.org/>
- SonarCloud: <https://www.sonarcloud.io>

Follow

- Dana Epp: <https://danaepp.com>
- AuditWolf: <https://www.auditwolf.com>

Questions??

Dana Epp

Microsoft Regional Director

<https://danaepp.com>

