7th UCAAT User Conference on Advanced Automated Testing

ETSI

Bordeaux, 22-24 October 2019

HOW TO ENSURE TESTİNG ROBUSTNESS İN MİCROSERVİCE ARCHİTECTURES AND COPE WİTH TEST SMELLS

Presented by [Mesut Durukal & Buse Ozarslan]

SIEMENS
Ingenuity for life

About US

- Mesut Durukal, Test Manager, Siemens AG

  in linkedin.com/in/mesutdurukal/

  @DurukalMesut

- Buse Ozarslan, Test Leader, Siemens AG

  in linkedin.com/in/buseozarslan/

AGENDA

- Introduction
  - SUT
  - Our Test Automation Environment
- Test Smells
  - Definition
  - Types
- Solutions
  - Helper Classes
  - Polling Mechanisms
  - Historical Test Executions
  - Clean Up
  - Annotations
  - Coverage Improvements
  - Static Code Analysis

INTRODUCTION

System Under Test

MindApps

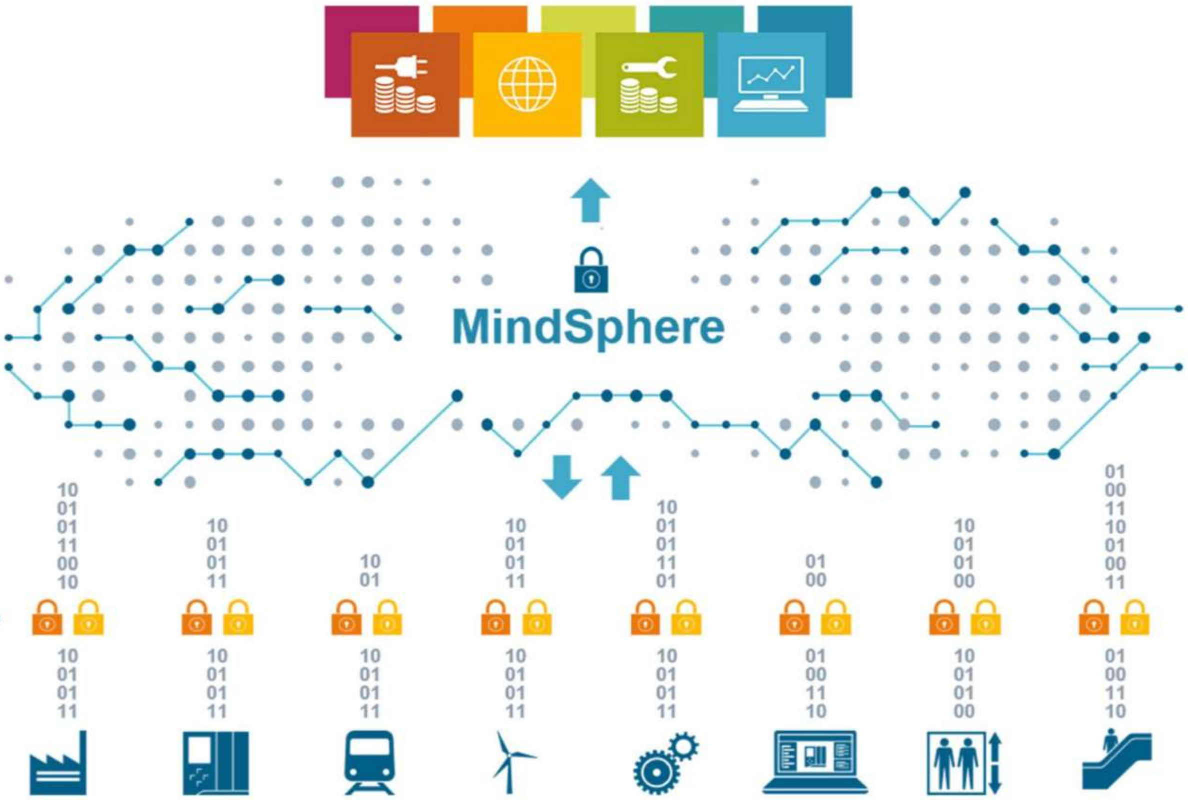
Powerful industry applications and digital services for asset transparency and analytical insights

MindSphere

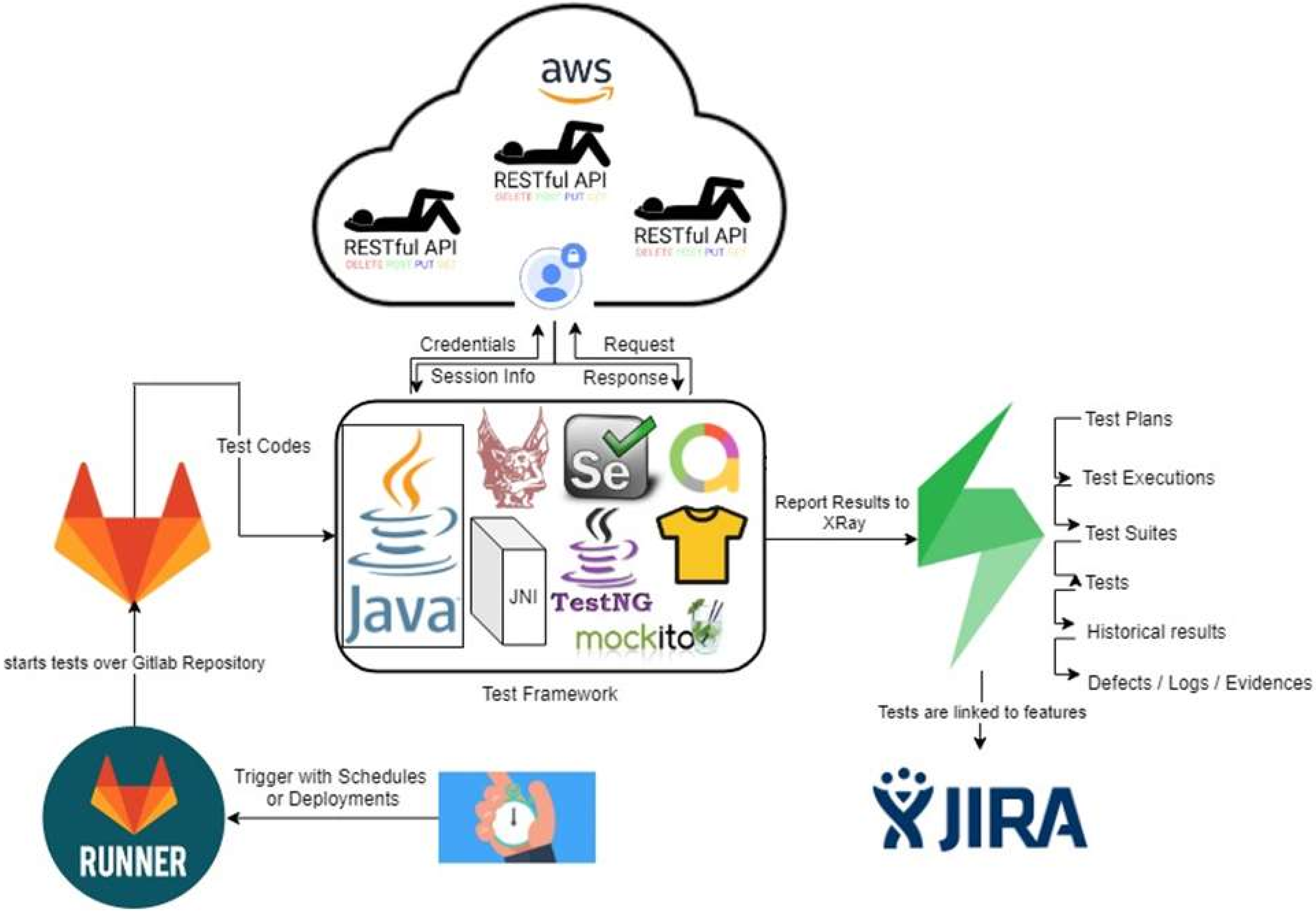
Open Platform as a Service (PaaS) for scalable, global IoT connectivity and application development

MindConnect

Secured plug-and-play connection of Siemens and third-party products, plants, systems and machines

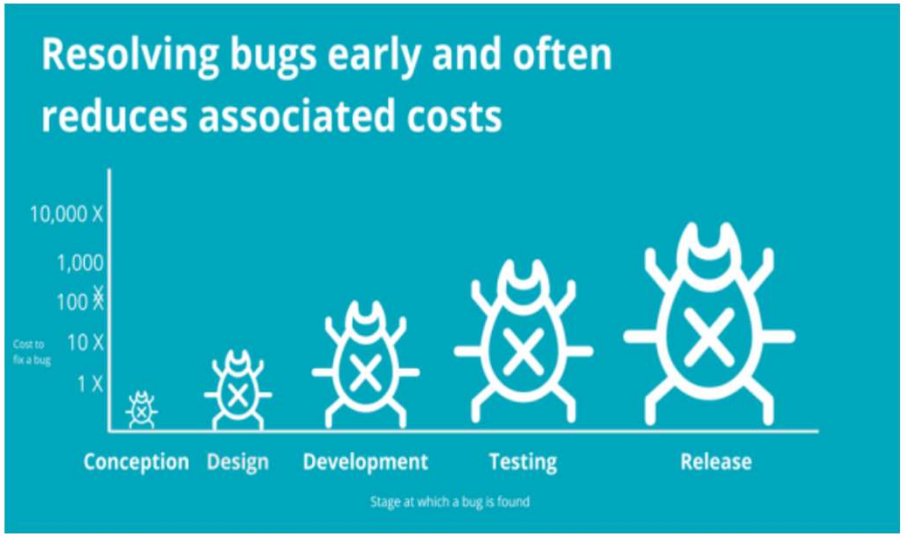


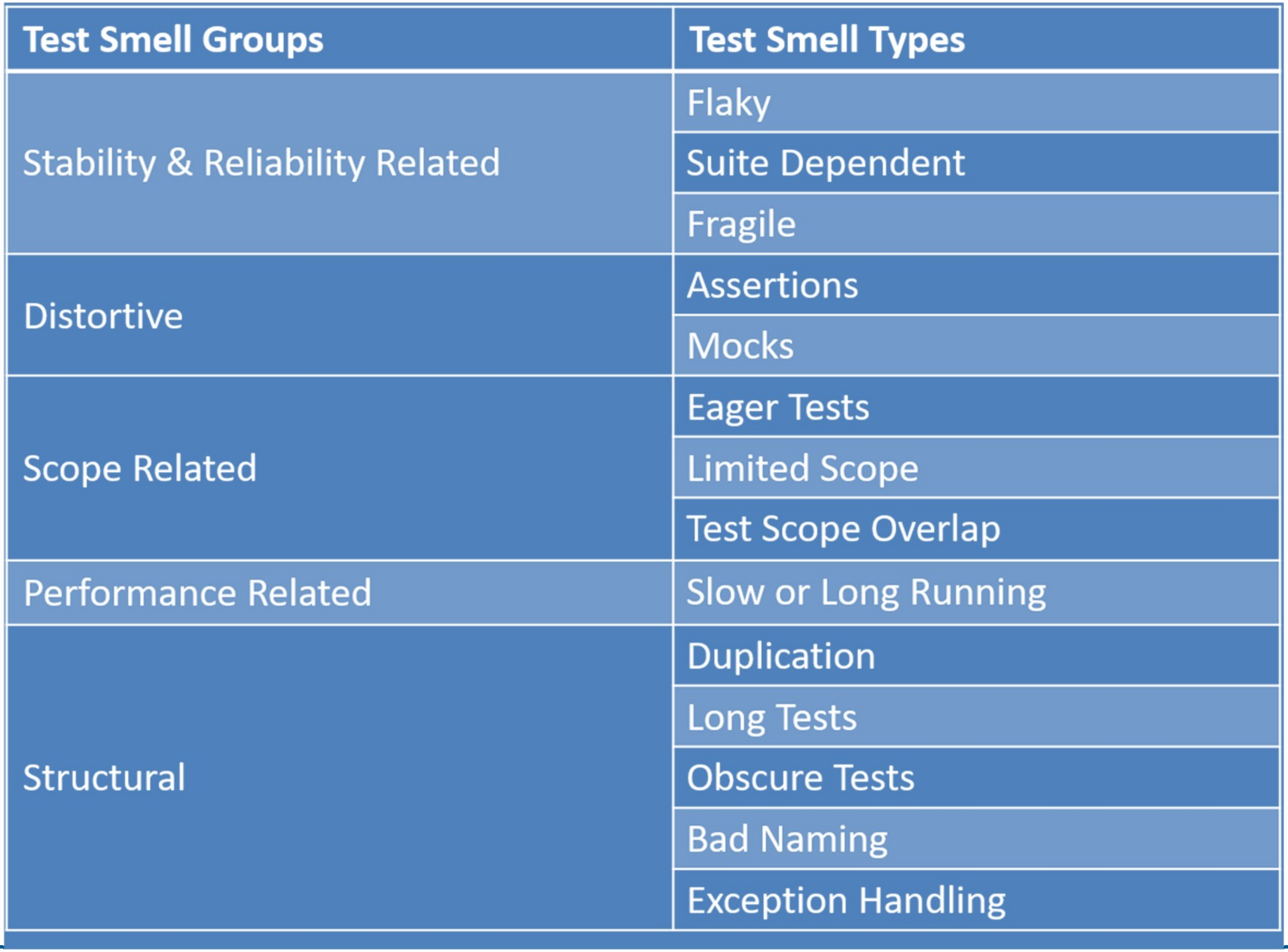
Microservices


aws
RESTful API
DELETE POST PUT GET
RESTful API
DELETE POST PUT GET
RESTful API
DELETE POST PUT GET
Credentials
Request
Session Info
Response
Test Codes
Java
Se
JNI
TestNG
mockito
Test Framework
Report Results to XRay
Test Plans
Test Executions
Test Suites
Tests
Historical results
Defects / Logs / Evidences
Tests are linked to features
JIRA
starts tests over Gitlab Repository
RUNNER
Trigger with Schedules or Deployments

TEST SMELLS

		Correct Results	
		Pass	*Fail*
Execution Results	*Pass*	*No Problem*	Silent Horror
	Fail	False Alarm	Real Bugs



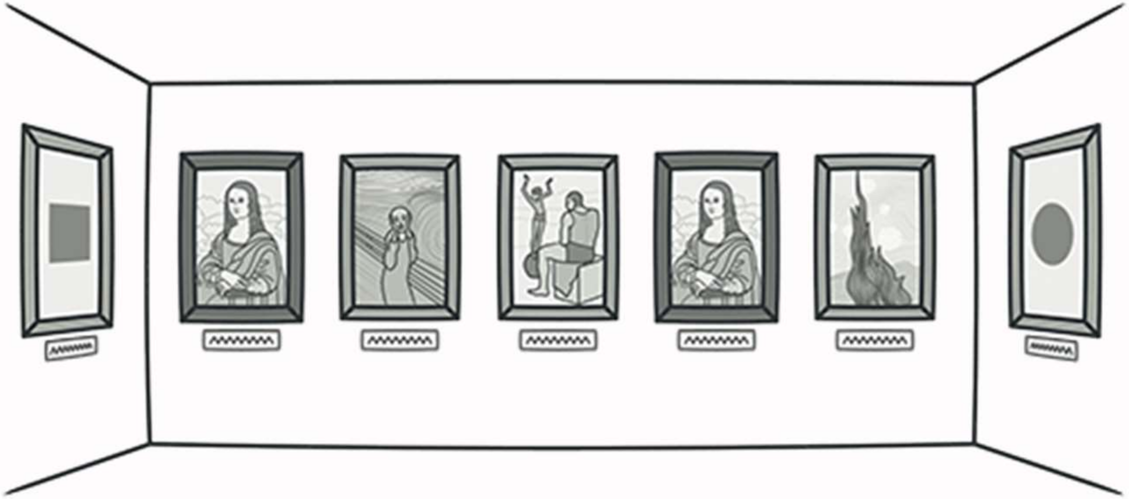


Test Smell Groups	Test Smell Types
Stability & Reliability Related	Flaky
	Suite Dependent
	Fragile
Distortive	Assertions
	Mocks
Scope Related	Eager Tests
	Limited Scope
	Test Scope Overlap
Performance Related	Slow or Long Running
Structural	Duplication
	Long Tests
	Obscure Tests
	Bad Naming
	Exception Handling

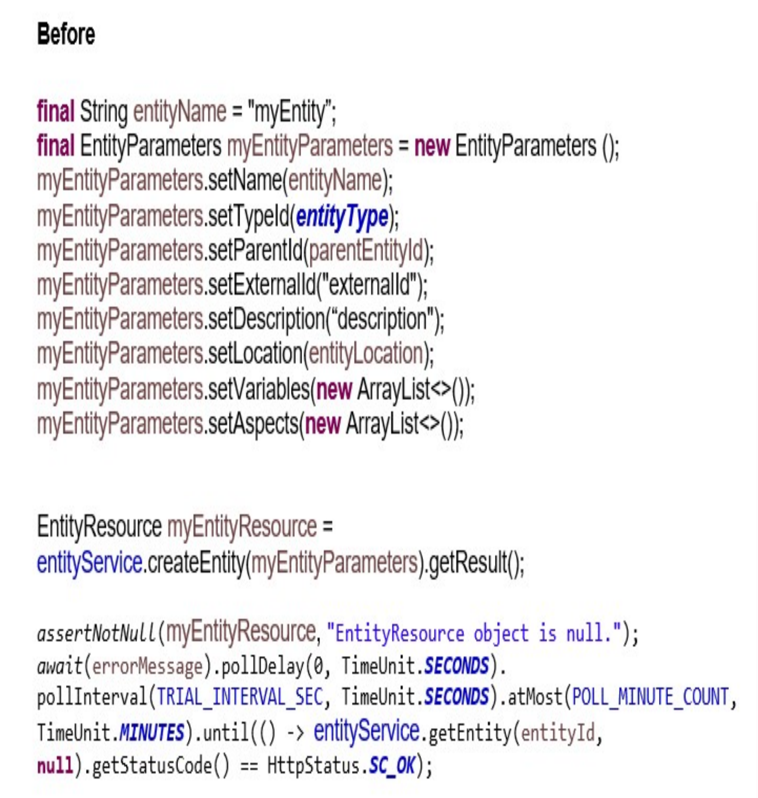
SOLUTIONS

Helper Classes

Duplication

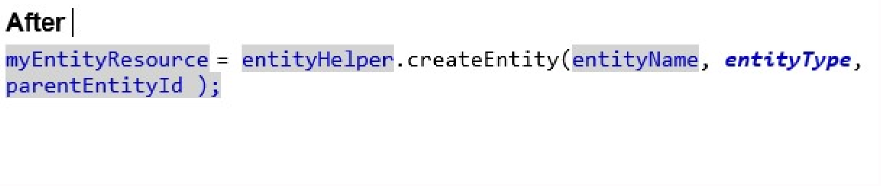


Helper Classes



Before

```
final String entityName = "myEntity";
final EntityParameters myEntityParameters = new EntityParameters ();
myEntityParameters.setName(entityName);
myEntityParameters.setTypeId(entityType);
myEntityParameters.setParentId(parentEntityId);
myEntityParameters.setExternalId("externalId");
myEntityParameters.setDescription(“description");
myEntityParameters.setLocation(entityLocation);
myEntityParameters.setVariables(new ArrayList<>());
myEntityParameters.setAspects(new ArrayList<>());


EntityResource myEntityResource =
entityService.createEntity(myEntityParameters).getResult();

assertNotNull(myEntityResource, "EntityResource object is null.");
await(errorMessage).pollDelay(0, TimeUnit.SECONDS).
pollInterval(TRIAL_INTERVAL_SEC, TimeUnit.SECONDS).atMost(POLL_MINUTE_COUNT,
TimeUnit.MINUTES).until(() -> entityService.getEntity(entityId,
null).getStatusCode() == HttpStatus.SC_OK);
```

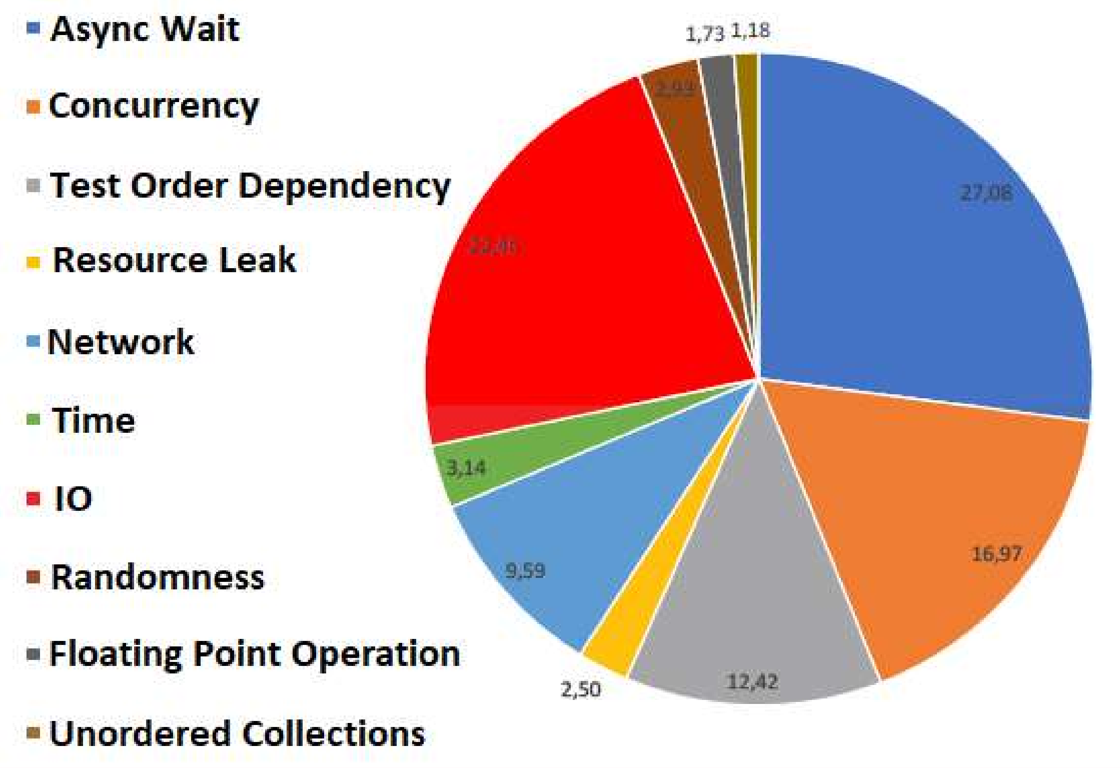


After

```
myEntityResource = entityHelper.createEntity(entityName, entityType,
parentEntityId );
```

Retry Algorithms

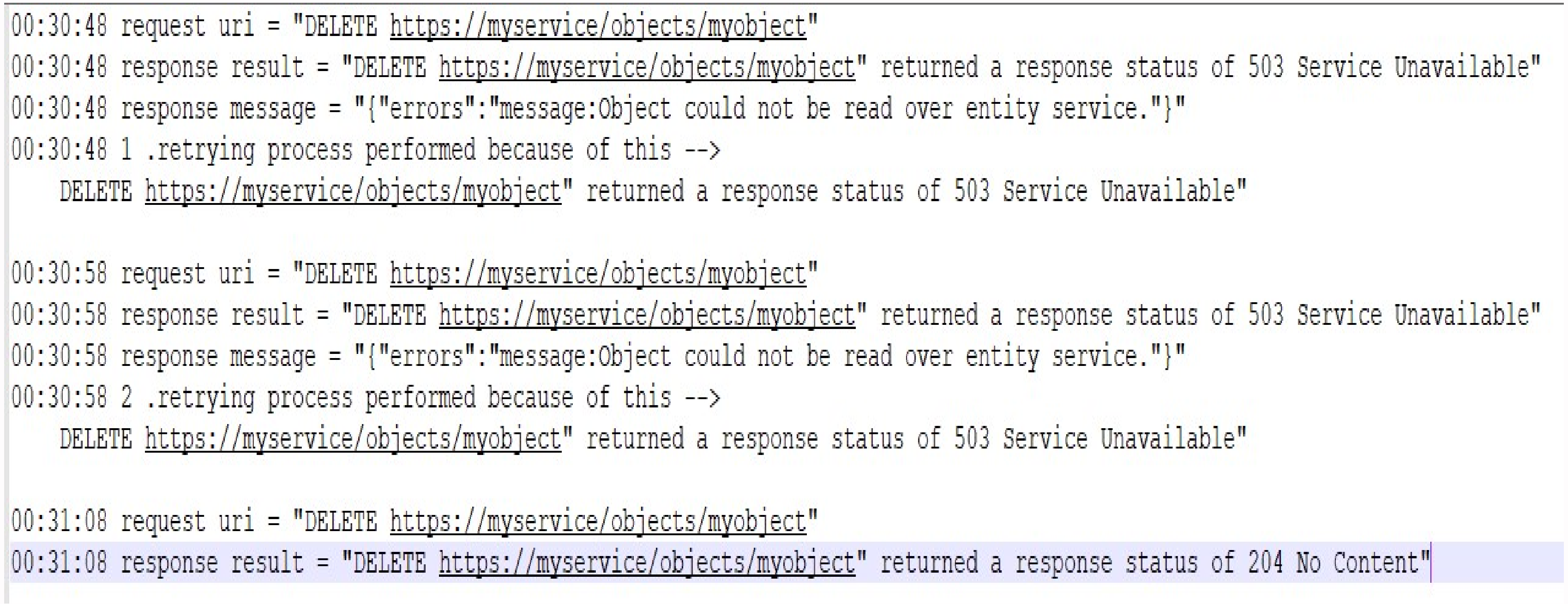
Flaky Tests



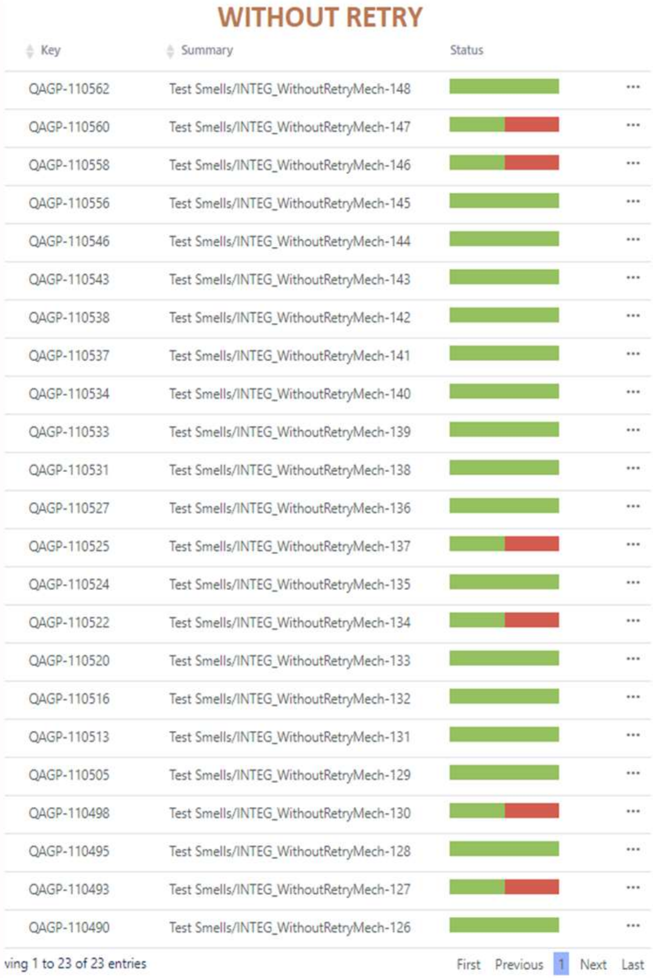
F. Palomba & A. Zaidman,

Does Refactoring of Test Smells Induce Fixing Flaky Tests?,

2017 ICSME



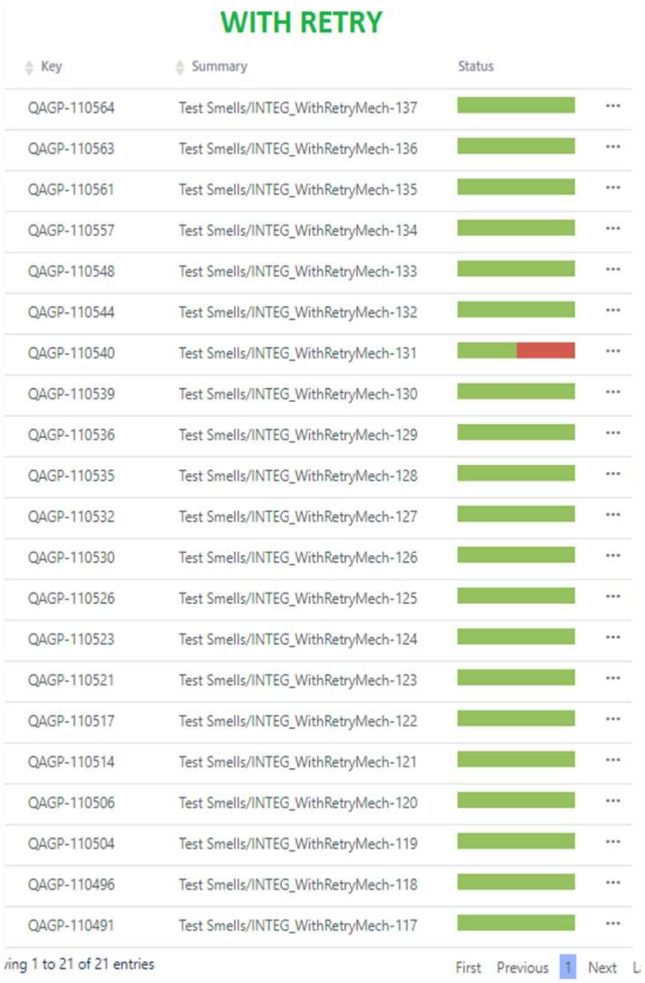
```
00:30:48 request uri = "DELETE https://myservice/objects/myobject"
00:30:48 response result = "DELETE https://myservice/objects/myobject" returned a response status of 503 Service Unavailable"
00:30:48 response message = "{"errors":"message:Object could not be read over entity service."}"
00:30:48 1 .retrying process performed because of this -->
    DELETE https://myservice/objects/myobject" returned a response status of 503 Service Unavailable"

00:30:58 request uri = "DELETE https://myservice/objects/myobject"
00:30:58 response result = "DELETE https://myservice/objects/myobject" returned a response status of 503 Service Unavailable"
00:30:58 response message = "{"errors":"message:Object could not be read over entity service."}"
00:30:58 2 .retrying process performed because of this -->
    DELETE https://myservice/objects/myobject" returned a response status of 503 Service Unavailable"

00:31:08 request uri = "DELETE https://myservice/objects/myobject"
00:31:08 response result = "DELETE https://myservice/objects/myobject" returned a response status of 204 No Content"
```



WITHOUT RETRY

Key	Summary	Status	
QAGP-110562	Test Smells/INTEG_WithoutRetryMech-148	🟩	···
QAGP-110560	Test Smells/INTEG_WithoutRetryMech-147	🟩🟥	···
QAGP-110558	Test Smells/INTEG_WithoutRetryMech-146	🟩🟥	···
QAGP-110556	Test Smells/INTEG_WithoutRetryMech-145	🟩	···
QAGP-110546	Test Smells/INTEG_WithoutRetryMech-144	🟩	···
QAGP-110543	Test Smells/INTEG_WithoutRetryMech-143	🟩	···
QAGP-110538	Test Smells/INTEG_WithoutRetryMech-142	🟩	···
QAGP-110537	Test Smells/INTEG_WithoutRetryMech-141	🟩	···
QAGP-110534	Test Smells/INTEG_WithoutRetryMech-140	🟩	···
QAGP-110533	Test Smells/INTEG_WithoutRetryMech-139	🟩	···
QAGP-110531	Test Smells/INTEG_WithoutRetryMech-138	🟩	···
QAGP-110527	Test Smells/INTEG_WithoutRetryMech-136	🟩	···
QAGP-110525	Test Smells/INTEG_WithoutRetryMech-137	🟩🟥	···
QAGP-110524	Test Smells/INTEG_WithoutRetryMech-135	🟩	···
QAGP-110522	Test Smells/INTEG_WithoutRetryMech-134	🟩🟥	···
QAGP-110520	Test Smells/INTEG_WithoutRetryMech-133	🟩	···
QAGP-110516	Test Smells/INTEG_WithoutRetryMech-132	🟩	···
QAGP-110513	Test Smells/INTEG_WithoutRetryMech-131	🟩	···
QAGP-110505	Test Smells/INTEG_WithoutRetryMech-129	🟩	···
QAGP-110498	Test Smells/INTEG_WithoutRetryMech-130	🟩🟥	···
QAGP-110495	Test Smells/INTEG_WithoutRetryMech-128	🟩	···
QAGP-110493	Test Smells/INTEG_WithoutRetryMech-127	🟩🟥	···
QAGP-110490	Test Smells/INTEG_WithoutRetryMech-126	🟩	···

ving 1 to 23 of 23 entries



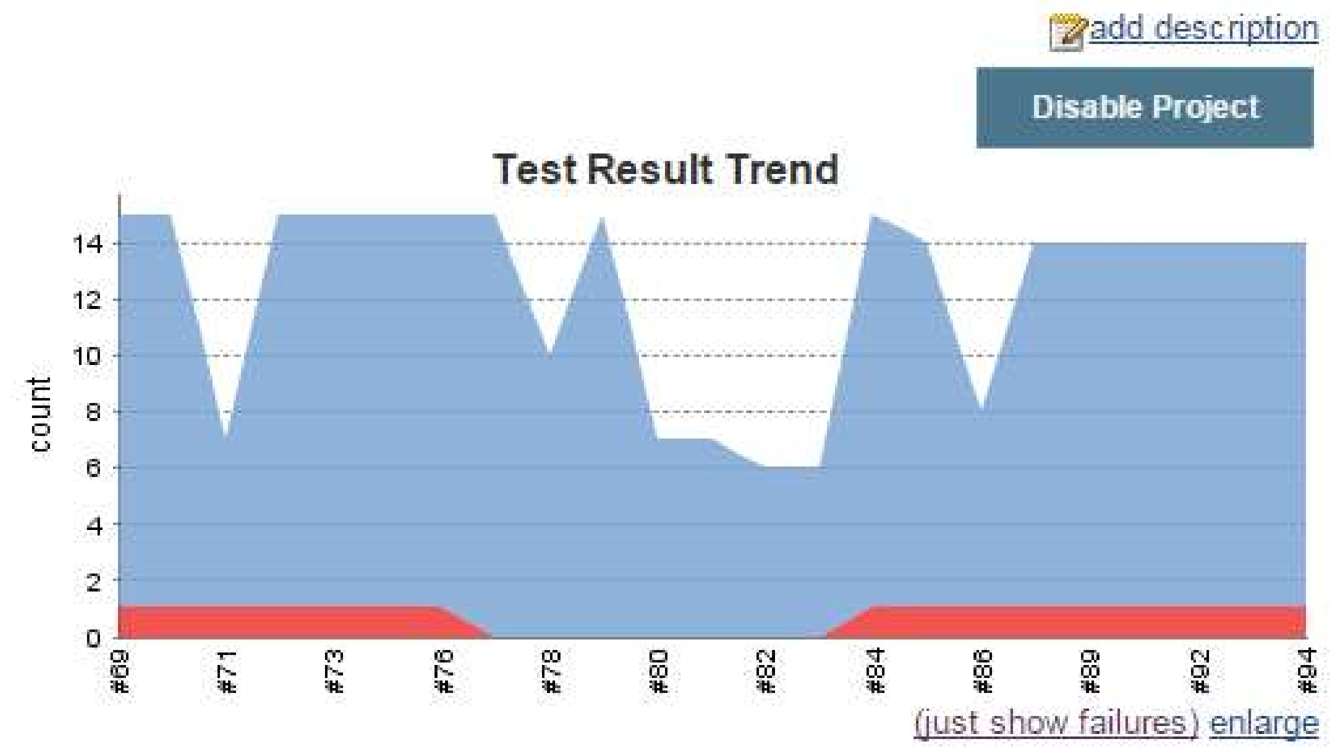
WITH RETRY

Key	Summary	Status	
QAGP-110564	Test Smells/INTEG_WithRetryMech-137	🟩	···
QAGP-110563	Test Smells/INTEG_WithRetryMech-136	🟩	···
QAGP-110561	Test Smells/INTEG_WithRetryMech-135	🟩	···
QAGP-110557	Test Smells/INTEG_WithRetryMech-134	🟩	···
QAGP-110548	Test Smells/INTEG_WithRetryMech-133	🟩	···
QAGP-110544	Test Smells/INTEG_WithRetryMech-132	🟩	···
QAGP-110540	Test Smells/INTEG_WithRetryMech-131	🟩🟥	···
QAGP-110539	Test Smells/INTEG_WithRetryMech-130	🟩	···
QAGP-110536	Test Smells/INTEG_WithRetryMech-129	🟩	···
QAGP-110535	Test Smells/INTEG_WithRetryMech-128	🟩	···
QAGP-110532	Test Smells/INTEG_WithRetryMech-127	🟩	···
QAGP-110530	Test Smells/INTEG_WithRetryMech-126	🟩	···
QAGP-110526	Test Smells/INTEG_WithRetryMech-125	🟩	···
QAGP-110523	Test Smells/INTEG_WithRetryMech-124	🟩	···
QAGP-110521	Test Smells/INTEG_WithRetryMech-123	🟩	···
QAGP-110517	Test Smells/INTEG_WithRetryMech-122	🟩	···
QAGP-110514	Test Smells/INTEG_WithRetryMech-121	🟩	···
QAGP-110506	Test Smells/INTEG_WithRetryMech-120	🟩	···
QAGP-110504	Test Smells/INTEG_WithRetryMech-119	🟩	···
QAGP-110496	Test Smells/INTEG_WithRetryMech-118	🟩	···
QAGP-110491	Test Smells/INTEG_WithRetryMech-117	🟩	···

ving 1 to 21 of 21 entries

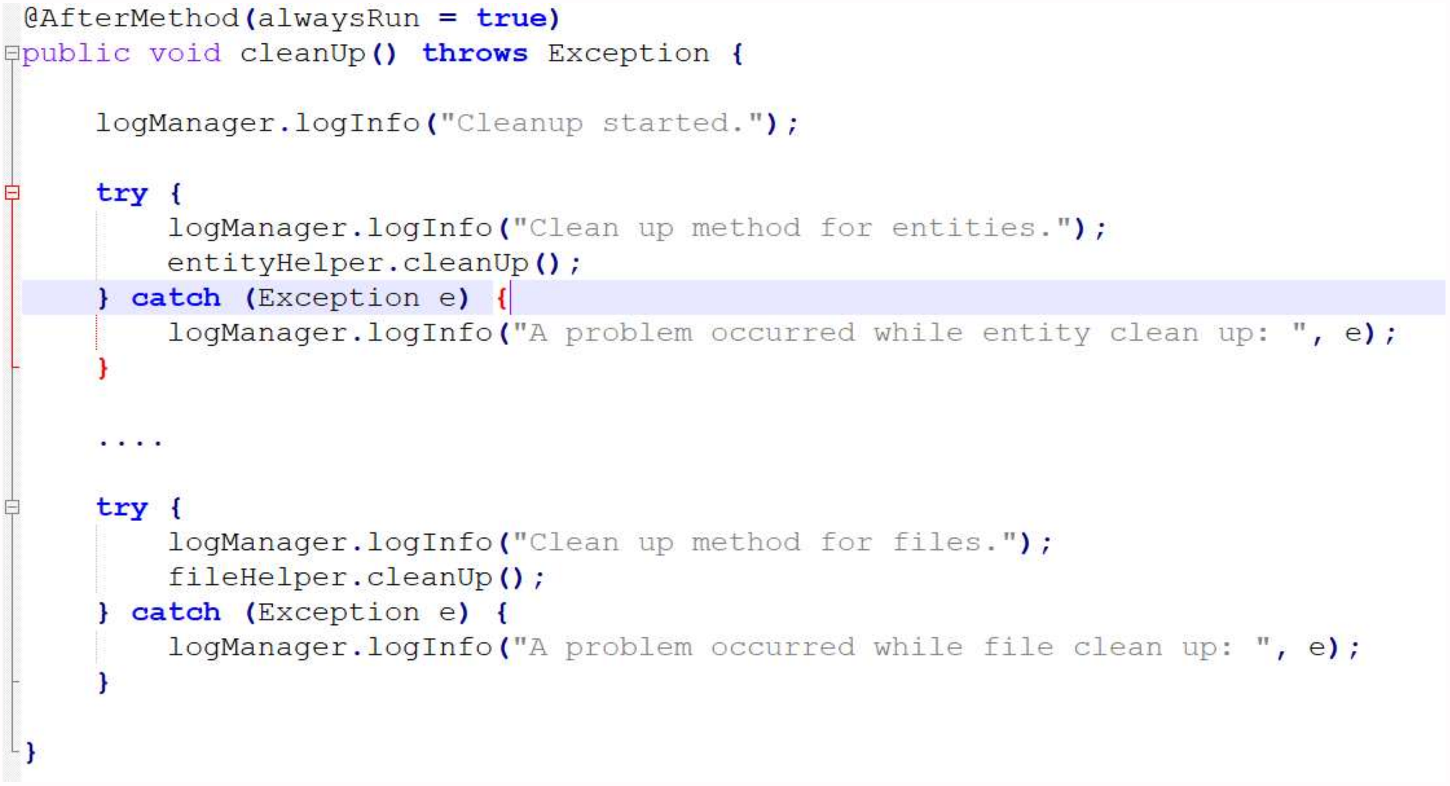
Test History

Flaky Tests



Flaky Tests

Clean Up



```
@AfterMethod(alwaysRun = true)
public void cleanUp() throws Exception {

    logManager.logInfo("Cleanup started.");

    try {
        logManager.logInfo("Clean up method for entities.");
        entityHelper.cleanUp();
    } catch (Exception e) {
        logManager.logInfo("A problem occurred while entity clean up: ", e);
    }

    ....

    try {
        logManager.logInfo("Clean up method for files.");
        fileHelper.cleanUp();
    } catch (Exception e) {
        logManager.logInfo("A problem occurred while file clean up: ", e);
    }

}
```

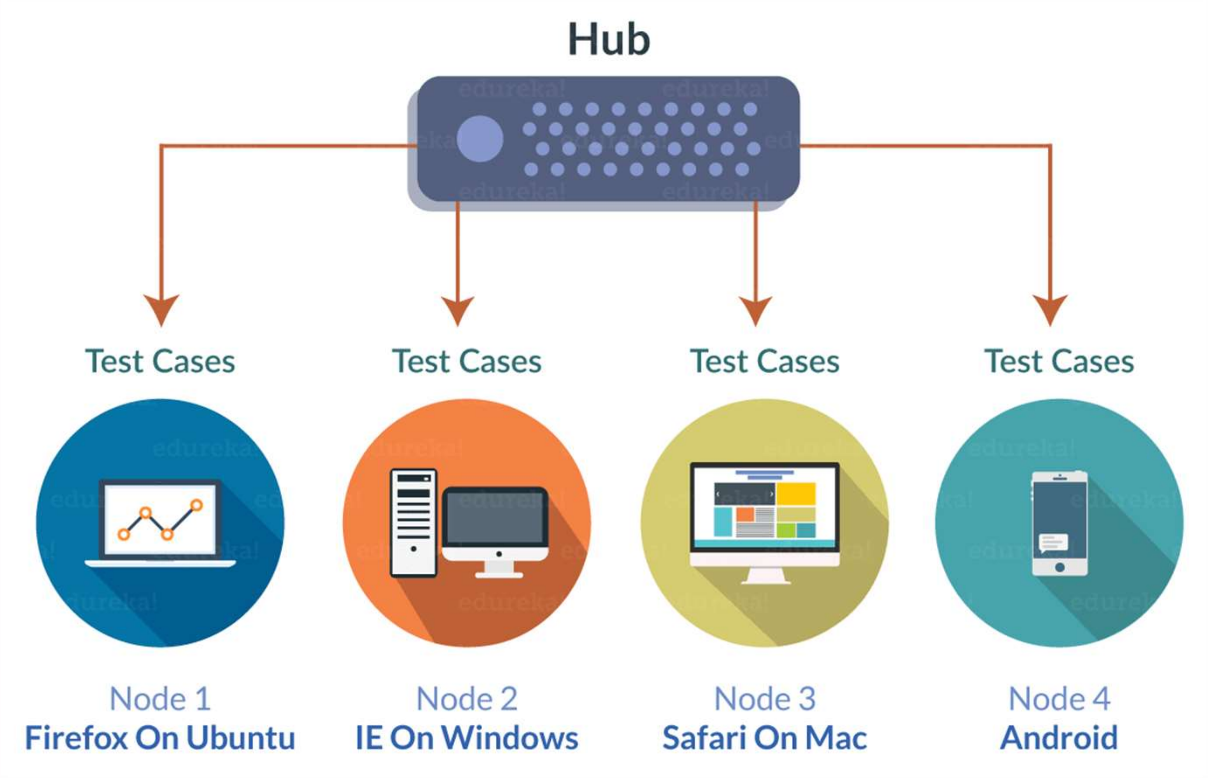
Annotations

Suite Dependency

```
@Test(groups = { TestGroups.SANITY,
TestGroups.ISTANBUL, TestGroups.E2E,
TestGroups.REGRESSION, TestGroups.UI }, enabled =
true)
```

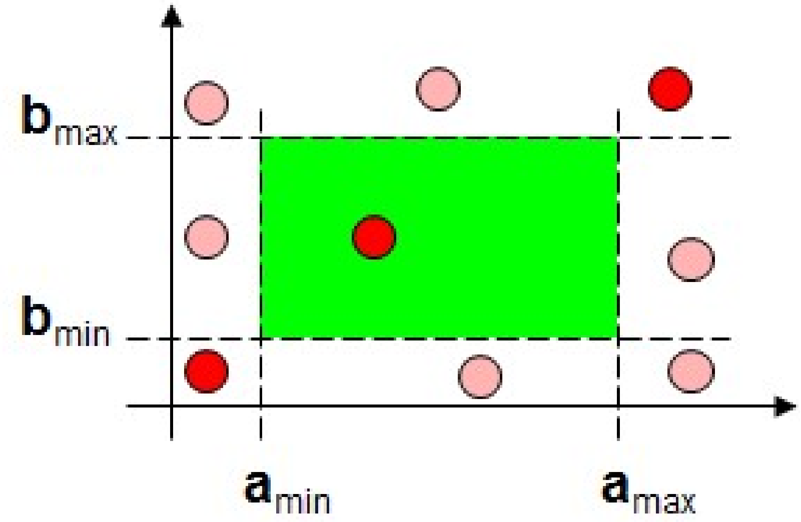
Test Coverage

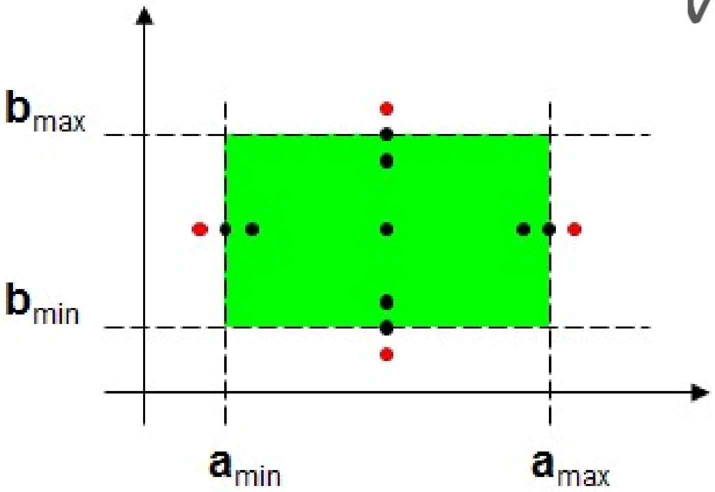
Scope



Test Input

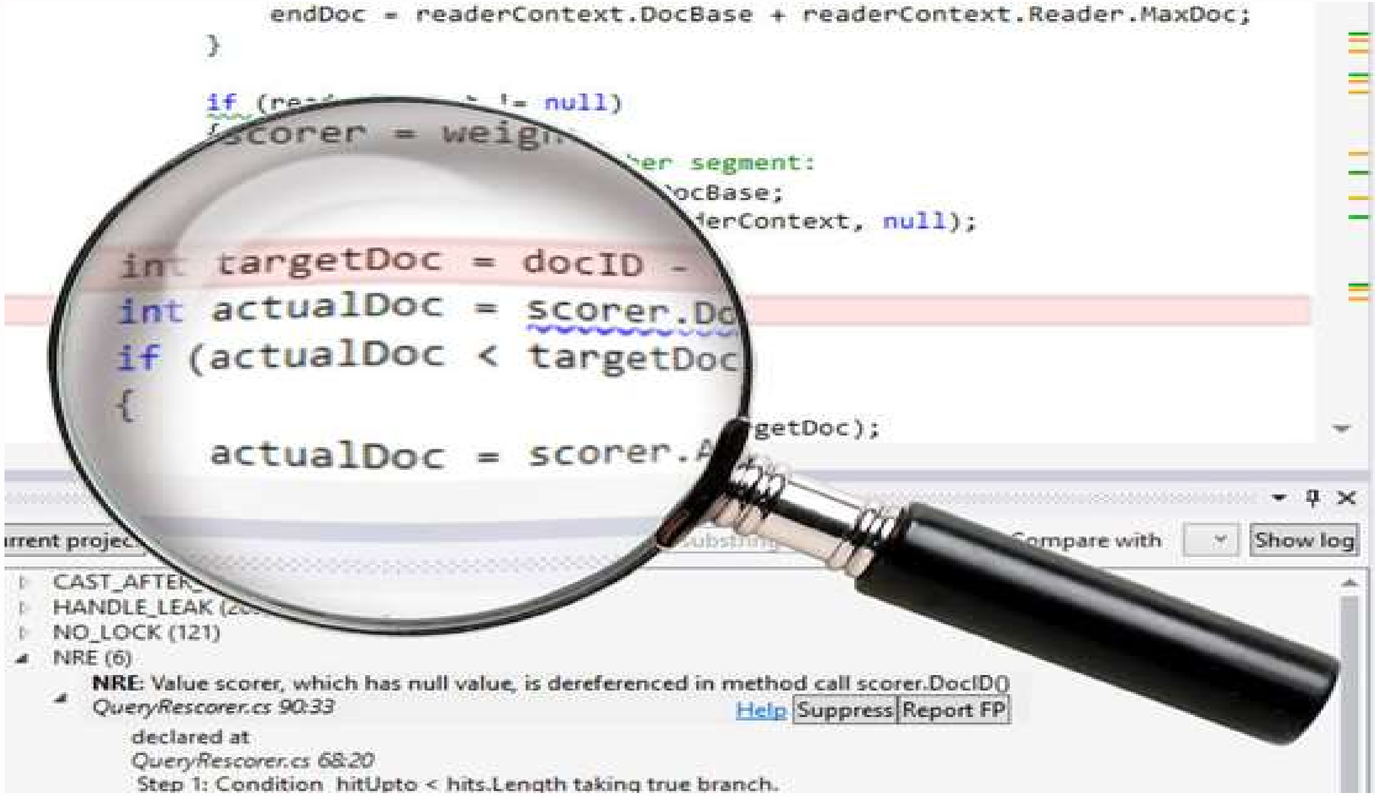
Scope





Code Quality

Fragile Tests, Distortive Tests, Structural Smells

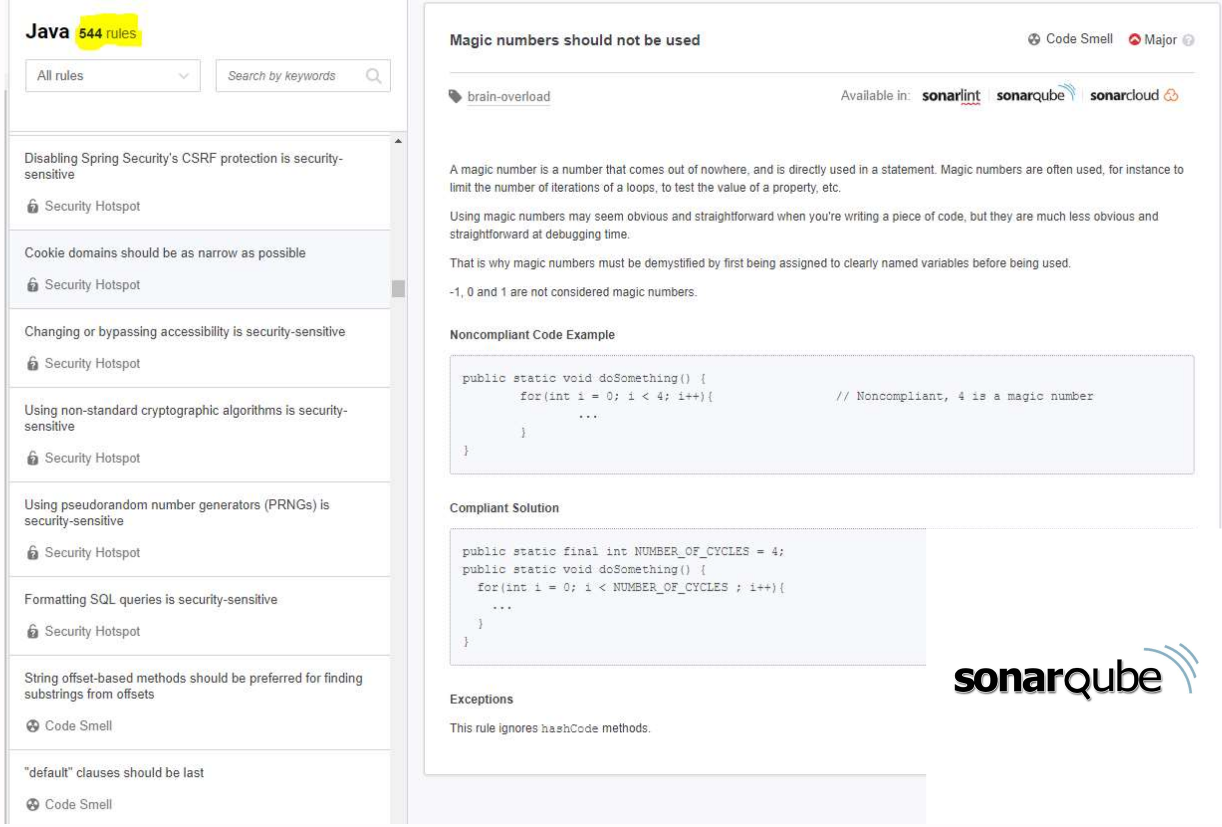


```
endDoc = readerContext.DocBase + readerContext.Reader.MaxDoc;
}

if (re       != null)
{scorer = weig
                   er segment:
                   ocBase;
                   derContext, null);
int targetDoc = docID -
int actualDoc = scorer.Do
if (actualDoc < targetDoc
{
    actualDoc = scorer.A        getDoc);
```

```
CAST_AFTER
HANDLE_LEAK (2
NO_LOCK (121)
NRE (6)
  NRE: Value scorer, which has null value, is dereferenced in method call scorer.DocID()
  QueryRescorer.cs 90:33
    declared at
    QueryRescorer.cs 68:20
    Step 1: Condition  hitUpto < hits.Length taking true branch.
```

Make Use of Tools

Fragile Tests, Distortive Tests, Structural Smells



Java 544 rules

Disabling Spring Security's CSRF protection is security-sensitive
Security Hotspot

Cookie domains should be as narrow as possible
Security Hotspot

Changing or bypassing accessibility is security-sensitive
Security Hotspot

Using non-standard cryptographic algorithms is security-sensitive
Security Hotspot

Using pseudorandom number generators (PRNGs) is security-sensitive
Security Hotspot

Formatting SQL queries is security-sensitive
Security Hotspot

String offset-based methods should be preferred for finding substrings from offsets
Code Smell

"default" clauses should be last
Code Smell

Magic numbers should not be used

Code Smell Major

brain-overload

Available in: sonarlint sonarqube sonarcloud

A magic number is a number that comes out of nowhere, and is directly used in a statement. Magic numbers are often used, for instance to limit the number of iterations of a loops, to test the value of a property, etc.

Using magic numbers may seem obvious and straightforward when you're writing a piece of code, but they are much less obvious and straightforward at debugging time.

That is why magic numbers must be demystified by first being assigned to clearly named variables before being used.

-1, 0 and 1 are not considered magic numbers.

Noncompliant Code Example

```
public static void doSomething() {
        for(int i = 0; i < 4; i++){                 // Noncompliant, 4 is a magic number
                ...
        }
}
```

Compliant Solution

```
public static final int NUMBER_OF_CYCLES = 4;
public static void doSomething() {
  for(int i = 0; i < NUMBER_OF_CYCLES ; i++){
    ...
  }
}
```

Exceptions

This rule ignores `hashCode` methods.

SUMMARY

Action	Resolves
Helper Classes	Duplication
Polling Mechanisms	Flaky Tests
Test History	Flaky Tests
Clean Up	Flaky Tests, Suite Dependency
Annotations	Suite Dependency
Improved Coverage	Scope Smells
Static Code Analysis	Fragile Tests, Distortive Smells, Structural Smells

рахмат
ngiyabonga
danke
謝謝
spas
Баярлалаа
mersi
kia ora
barka
welalin
tack
teşekkür ederim
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
спасибі
blagodaram
dank je
misaotra
matondo
paldies
grazzi
kiitos
dankie
nanni
nandri
dhanyavad
mauruuru
köszönöm
thank you
gracias
хвала
asante
mana
bedankt
bayarlalaa
gràcie
hvala
dankon
aciū
akun
obrigada
murakoze
nkosi
djiere dieuf
tau
mochchakkeram
дякую
mamnun
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
go raibh maith agat
chokrane
sobodi
děkuji
sukriya
kop khun krap
trugarez
obrigado
mèsi
didi madloba
sagolun
ありがとう
tanemirt
grazie
arigatô
takk
dakujem
najis tuke
terima kasih
rahmet
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercé
мерсі
kam sah hamnida
rahmat
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci