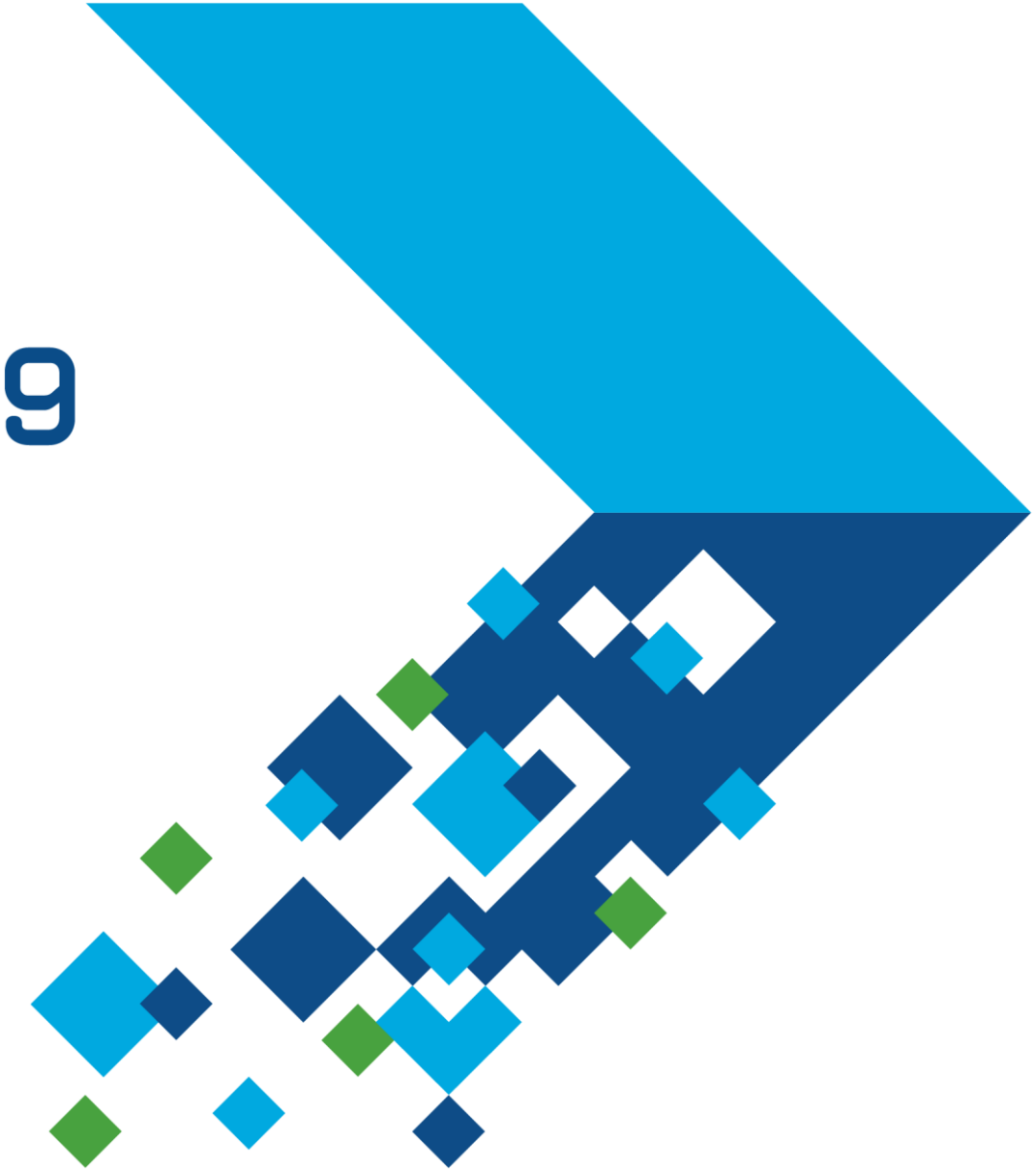
# MATLAB EXPO 2019

## 企业级代码验证流程
## 提升软件功能安全和网络安全

胡乐华



# 内容

1. 打造软件功能安全与网络安全

2. 无关键运行错误的证明能力

3. 协同合作流程提升代码质量

# 1. 打造软件功能安全与网络安全

“Program testing can be used to show the presence of bugs, but never to show their absence”

程序测试可以显示软件问题的存在，但永远无法显示其不存在。

**Edsger Dijkstra**, Computer Science Pioneer

“Given that we cannot really show there are no more errors in the program, when do we stop testing?”

既然我们无法知道程序中还有没有问题，那我们何时该停止测试？

**Brent Hailpern**, Head of Computer Science

Dijstra, “Notes on Structured Programming” (1972)
Hailern, Santhanam, “Software Debugging, Testing, and Verification”, IBM Systems Journal, (2002)

# 打造软件安全的有力武器 – 静态分析技术

- 无需运行代码即可找到软件错误
  - 识别违背MISRA/CERT规范的代码
- 证明不存在关键运行错误
  - 识别出任何运行情况下都不会出错的代码
- 动态测试的有益补充
  - 同时使用可以找到更多问题提升代码质量

main.cpp

```
20
21  static bool table_loop(void)
22  {
23      int j = 4;
24
25      // Table of basic element
26      Base* array[] = { new SAnalogic, new Sensor, new Sensor, new SAnalogic };
27
28      for (int i = 4; i >= 0; i--, j--) {
29          array[i-1]->Draw();
30
31          // Error for the 2 last elements: this cast is similar to static_cast
32          // the TypeInfo function only define in SAnalogic
33          if (i % 2)
34              ((SAnalogic*)(array[i-1]))->TypeInfo();
35          else
36              (dynamic_cast<SAnalogic*>(array[i-1]))->TypeInfo();
37      }
```

| | Event | File | Scope |
|---|---|---|---|
| 1 | Iterating on loop | main.cpp | table_loop() |
| 2 | This-pointer of TypeInfo is null | main.cpp | table_loop() |
| 3 | ● Non-terminating loop | main.cpp | table_loop() |

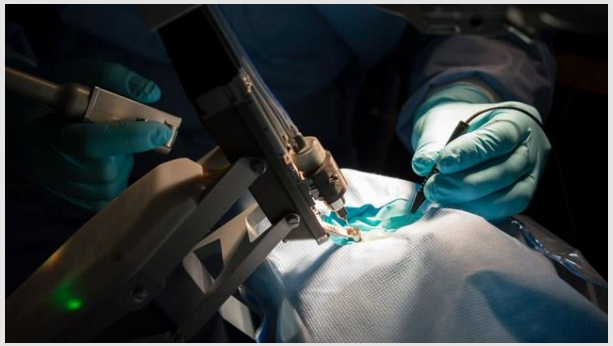
**● Non-terminating loop** (?)

The loop is infinite or contains a run-time error.

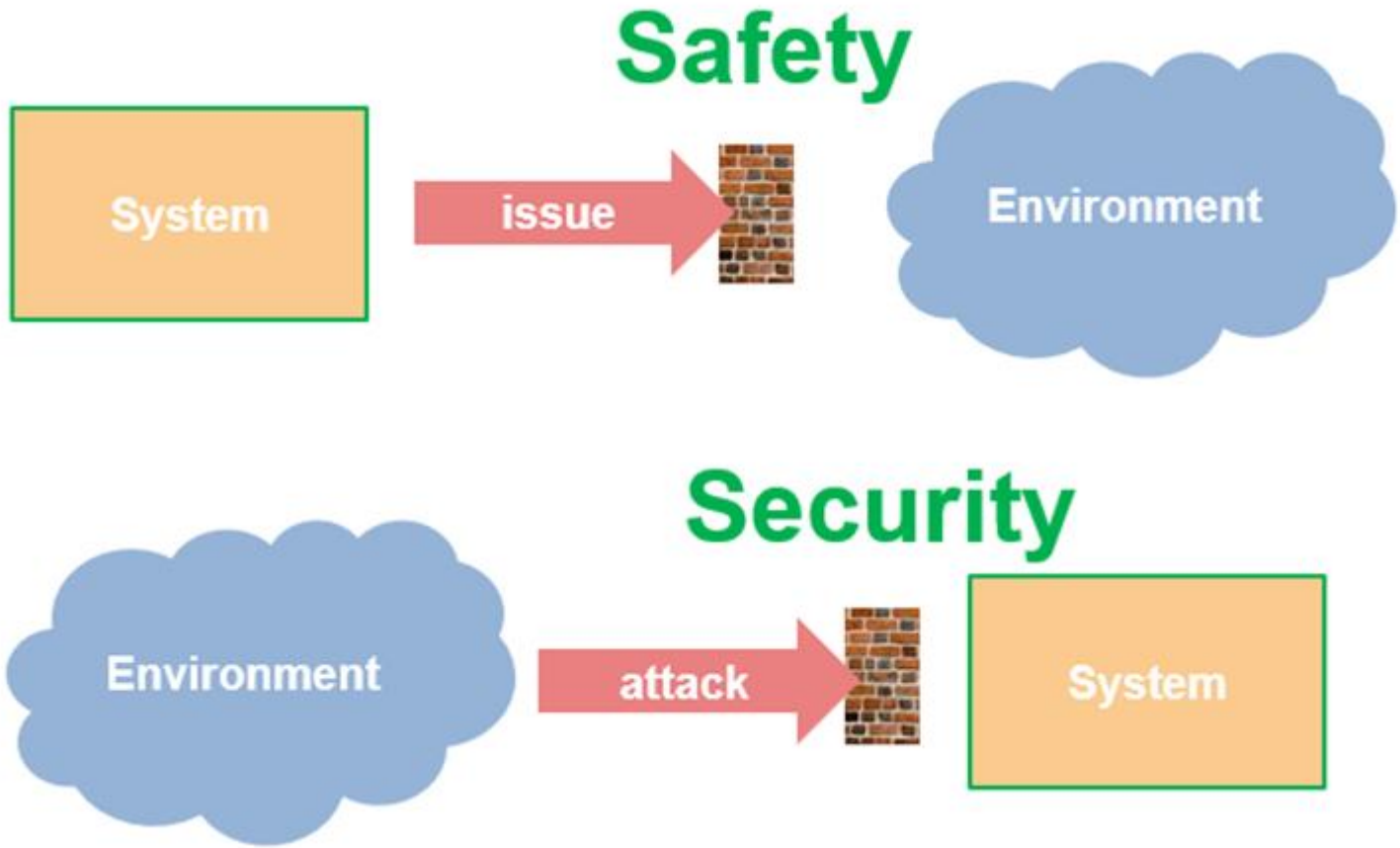
Loop fails due to a run-time error (maximum number of iterations: 3).

# 软件安全的重要性

- 高完整性系统应用行业
  - 汽车, 航空, 医疗, 工业…
- 功能安全和相关标准
  - ISO 26262, DO-178, IEC 62304, IEC 61508
  - MISRA, CERT, AUTOSAR
- 认证依据
  - 如ISO 26262和DO-178



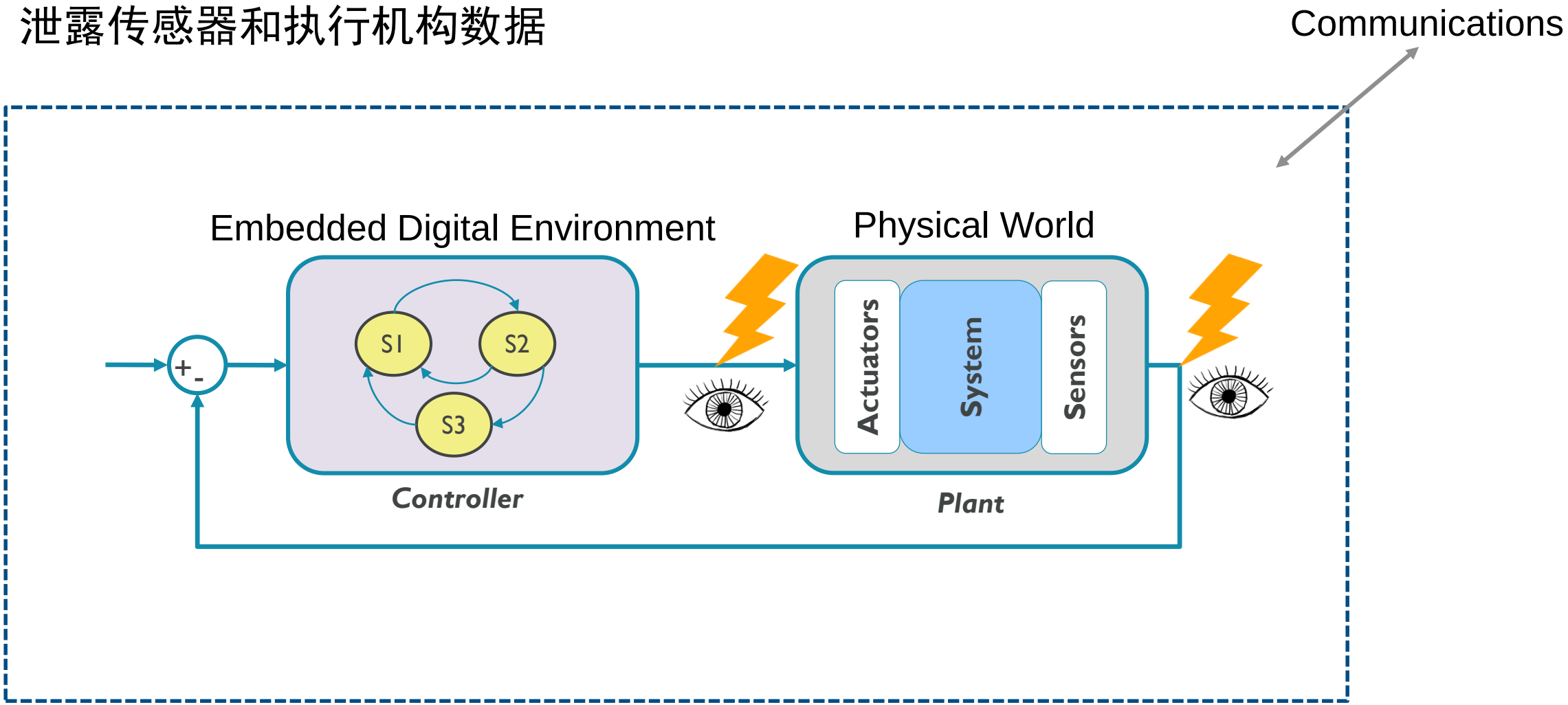
# 功能安全与网络安全




注：网络安全问题可能导致功能安全问题

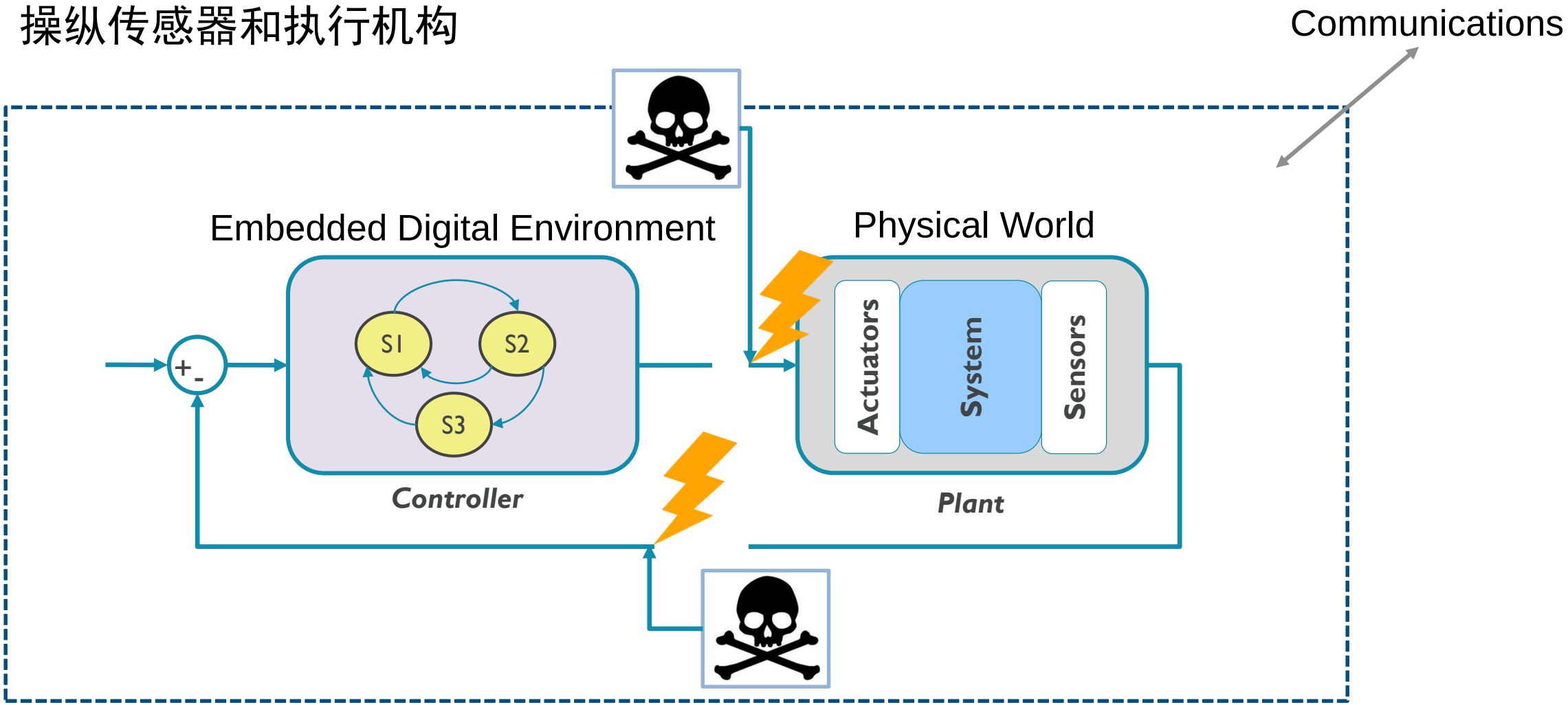
# 网络安全问题场景 1/2

泄露传感器和执行机构数据



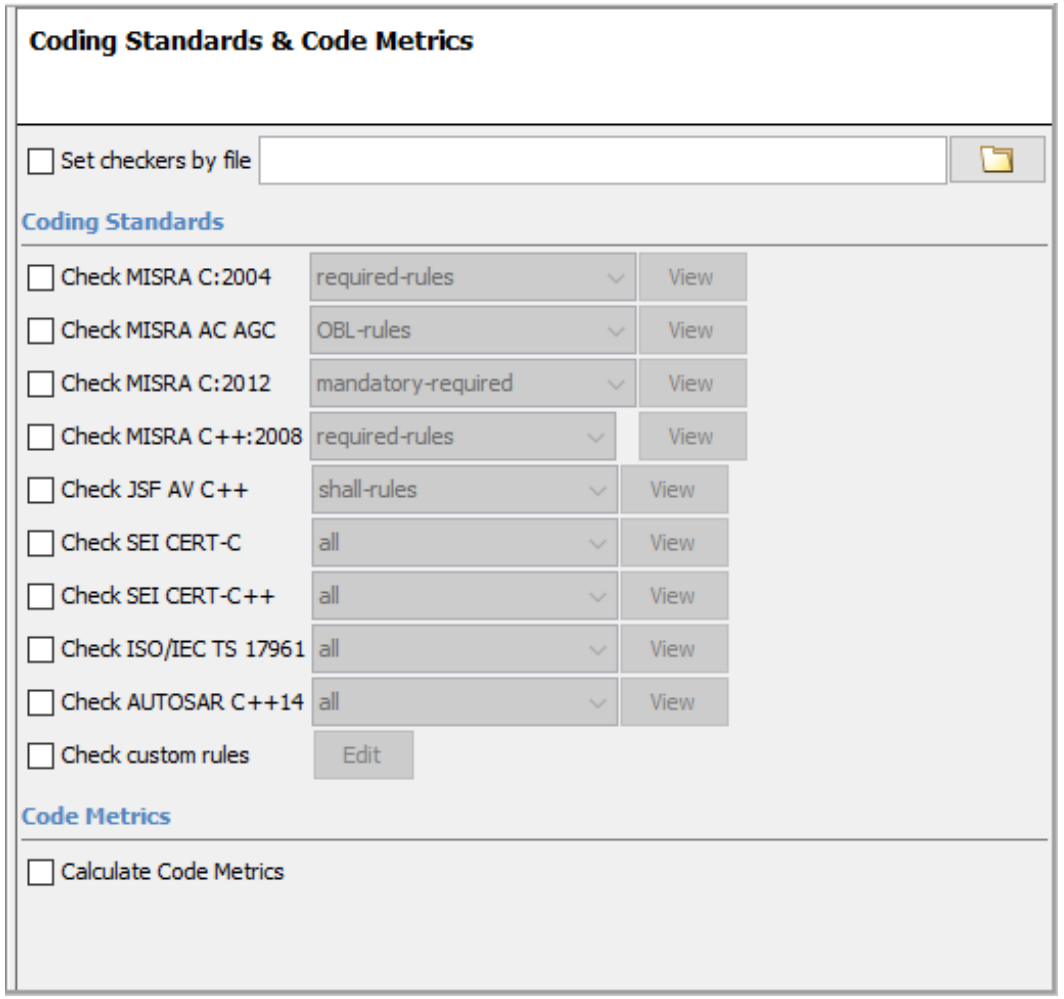
# 网络安全问题场景 2/2

## 操纵传感器和执行机构



# Polyspace对网络安全的支持

- 行业标准
  - ISO/IEC TS 17961 – C Secure Coding Rules
  - MISRA-C:2012 Amendment 1
  - CERT C
  - CWE – Common Weakness Enumeration



# 2. 无关键运行错误的证明能力

# 证明无关键运行错误

```
float x, y;

…

x = x / (x – y);
```

- 这段程序中有多少可能的运行错误?
  1. 除零
  2. 溢出
  3. 变量未初始化
- 测试是否能证明其安全?
- 如何测试所有的浮点变量组合?
- 如何证明这段程序不会出错?

# 证明无关键运行错误

Polyspace可以证明程序
永远不会发生运行错误

**✓ Division by zero** ?

Float division by zero does not occur

operator / on type float 32

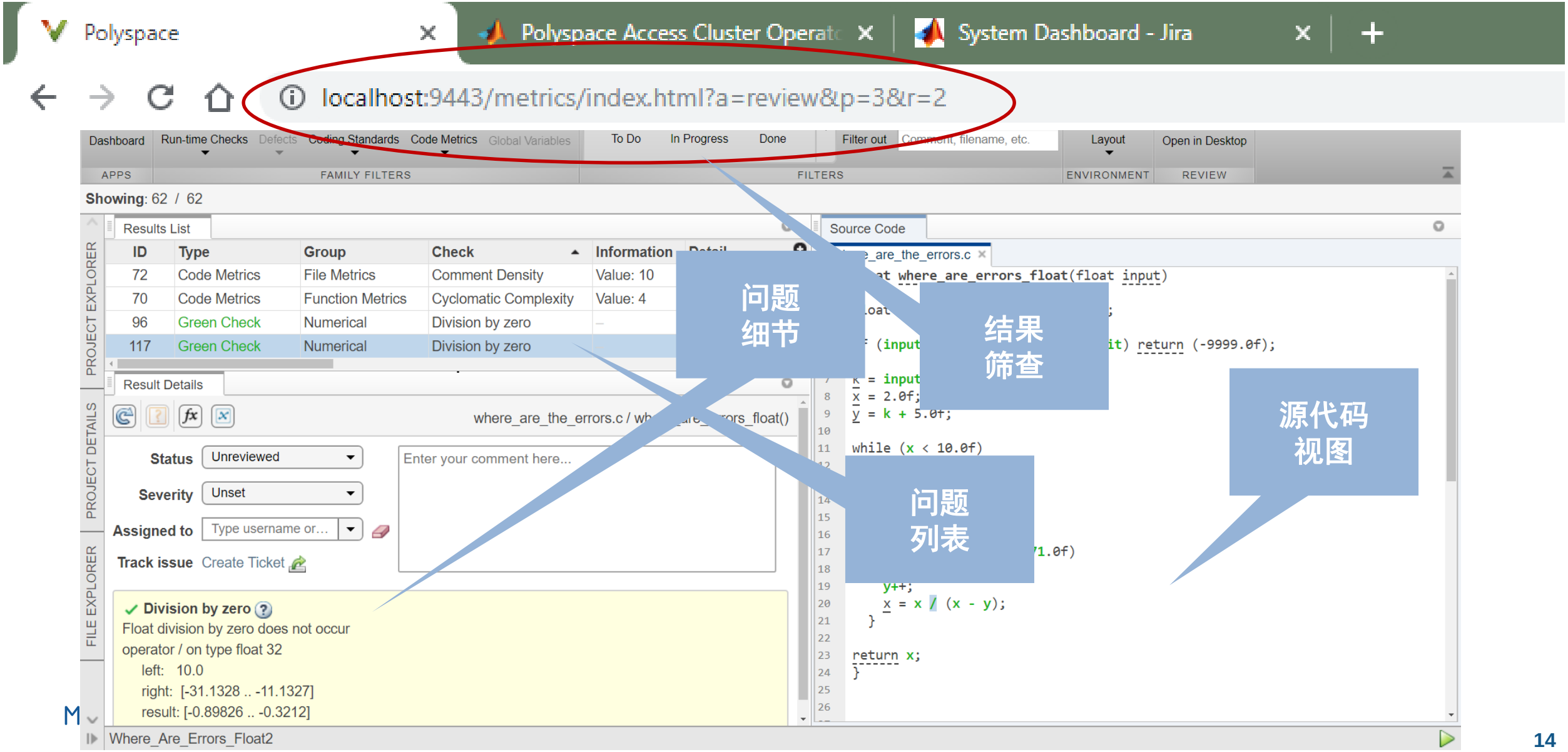
left: 10.0

right: [-31.1328 .. -11.1327]

result: [-0.89826 .. -0.3212]

```
1  float where_are_errors_float(float input)
2  {
3  float x, y, k, l, limit = 1000.0f;
4
5  if (input < -limit || input > limit) return (-9999.0f);
6
7  k = input / 100.0f;
8  x = 2.0f;
9  y = k + 5.0f;
10
11 while (x < 10.0f)
12   {
13     x++;
14     y = y + 3.141592f;
15   }
16
17 if ((3.0*k + 100.0f) > 71.0f)
18   {
19     y++;
20     x = x / (x - y);
21   }
22
23 return x;
24 }
```

# 证明无关键运行错误 – 基于网页技术的结果分享



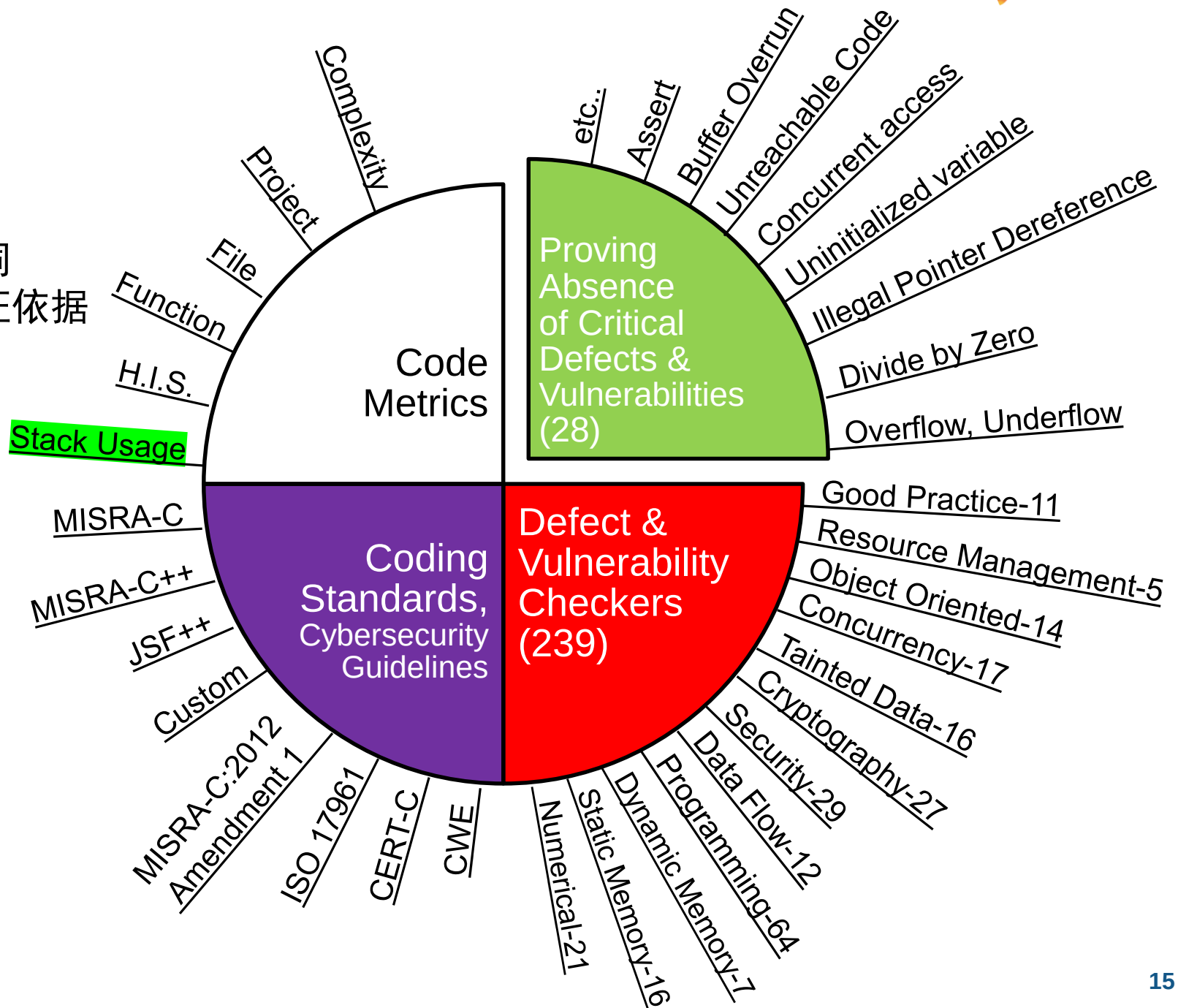
# Polyspace功能一览

## Bug Finder

- 保证可测量性和可维护性
- 排除绝大多数软件缺陷和漏洞
- 提供功能安全和网络安全认证依据

## Code Prover

- 确保可靠性和安全性
- 证明无关键运行错误和漏洞
- 提供附加认证审查证据



# Polyspace对功能安全的支持

- 以软件单元设计与验证为例

**Table 6 — Design principles for software unit design and implementation**

| Principle | | ASIL A | ASIL B | ASIL C | ASIL D |
|---|---|---|---|---|---|
| 1a | One entry and one exit point in subprograms and functions[a] | ++ | ++ | ++ | ++ |
| 1b | No dynamic objects or variables, or else online test during their creation[a] | + | ++ | ++ | ++ |
| 1c | Initialization of variables | ++ | ++ | ++ | ++ |
| 1d | No multiple use of variable names[a] | ++ | ++ | ++ | ++ |
| 1e | Avoid global variables or else justify their usage[a] | + | + | ++ | ++ |
| 1f | Restricted use of pointers[a] | . | . . | . . | . . |
| 1g | No implicit type conversions[a] | | | | |
| 1h | No hidden data flow or control flow | | | | |
| 1i | No unconditional jumps[a] | | | | |
| 1j | No recursions | | | | |

**Table 7 — Methods for software unit verification**

| Methods | | ASIL A | ASIL B | ASIL C | ASIL D |
|---|---|---|---|---|---|
| 1a | Walk-through[a] | ++ | + | o | o |
| 1b | Pair-programming[a] | + | + | + | + |
| 1c | Inspection[a] | + | ++ | ++ | ++ |
| 1d | Semi-formal verification | + | + | ++ | ++ |
| 1e | Formal verification | o | o | + | + |
| 1f | Control flow analysis[b, c] | + | + | ++ | ++ |
| 1g | Data flow analysis[b, c] | + | + | ++ | ++ |
| 1h | Static code analysis[d] | ++ | ++ | ++ | ++ |
| 1i | Static analyses based on abstract interpretation[e] | + | + | + | + |
| 1j | Requirements-based test[f] | ++ | ++ | ++ | ++ |
| 1k | Interface test[g] | ++ | ++ | ++ | ++ |

无关键运行错误证明

# 客户案例

Electronic Steering Lock

科世达亚太研发中心通过汽车软件的ISO26262 ASIL D认证

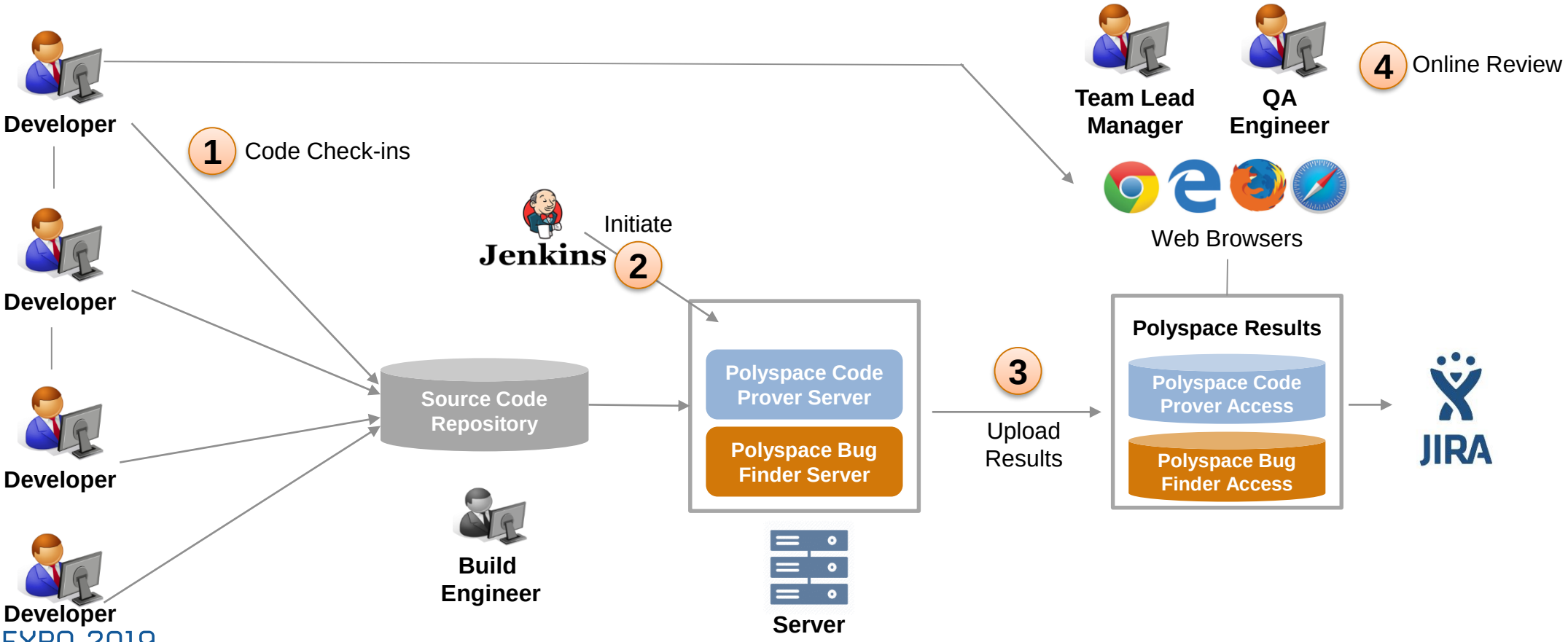
马基公司开发DO-178B Level A 认证自动驾驶软件

Miracor为Class III 医疗设备软件消除运行错误并节省测试时间

# 3. 协同合作流程提升代码质量

# R2019a Polyspace新产品系列的工作流程

1. 开发人员检入代码，构建工程师配置Jenkins流程
2. Jenkins在分析服务器端发起Polyspace分析(基于周期或项目节点)
3. Polyspace分析完成后自动上传到结果服务器
4. 项目经理、质量保证、开发主管和开发人员使用网页浏览器评审结果、发起JIRA跟踪、监控质量指标



# 企业级代码验证流程案例演示

Bob – 构建工程师
负责持续集成流程配置

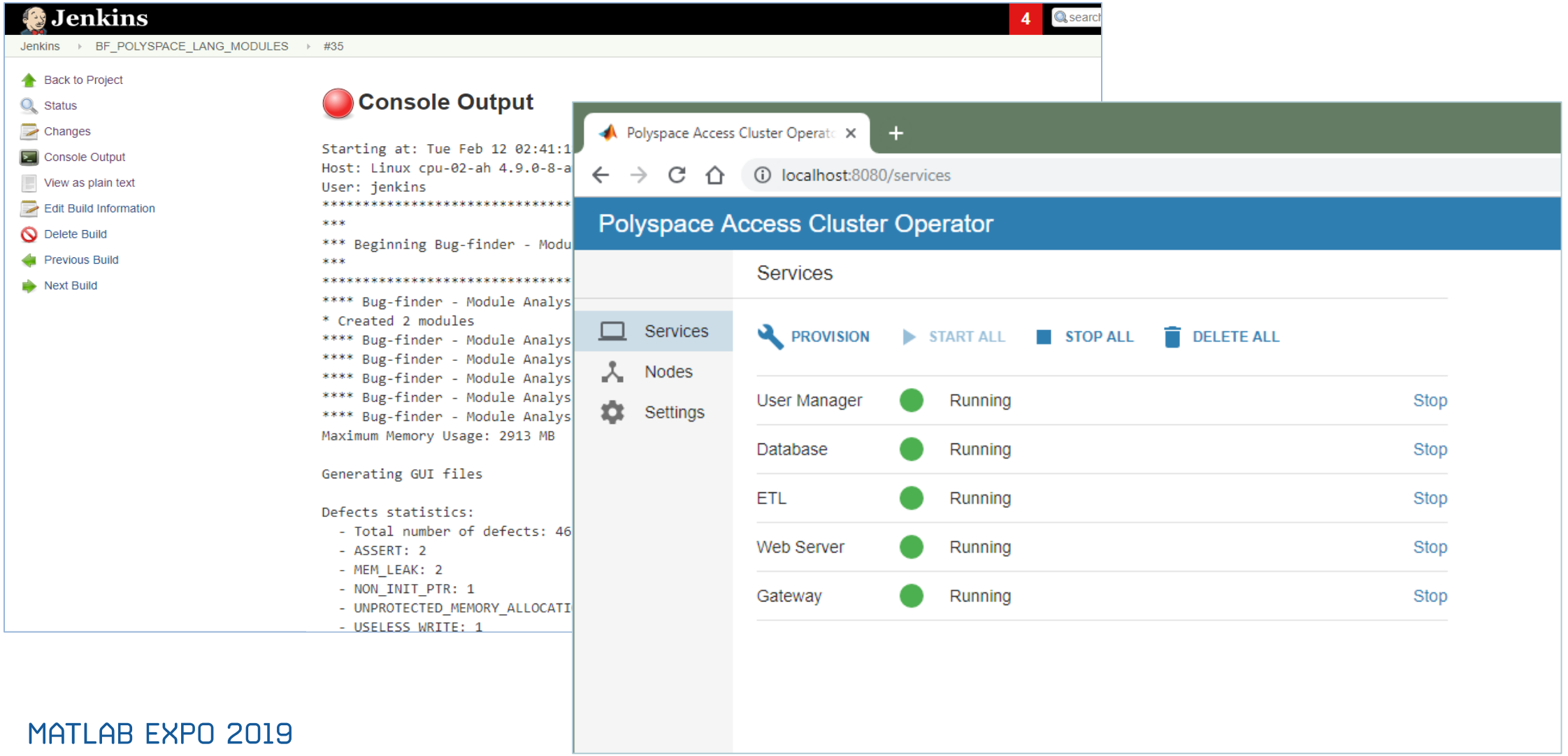
Quinn – 质量工程师
负责软件问题归类

Dara – 代码开发工程师
负责代码编写和问题修复

Martin – 模型开发工程师
负责模型开发和代码生成

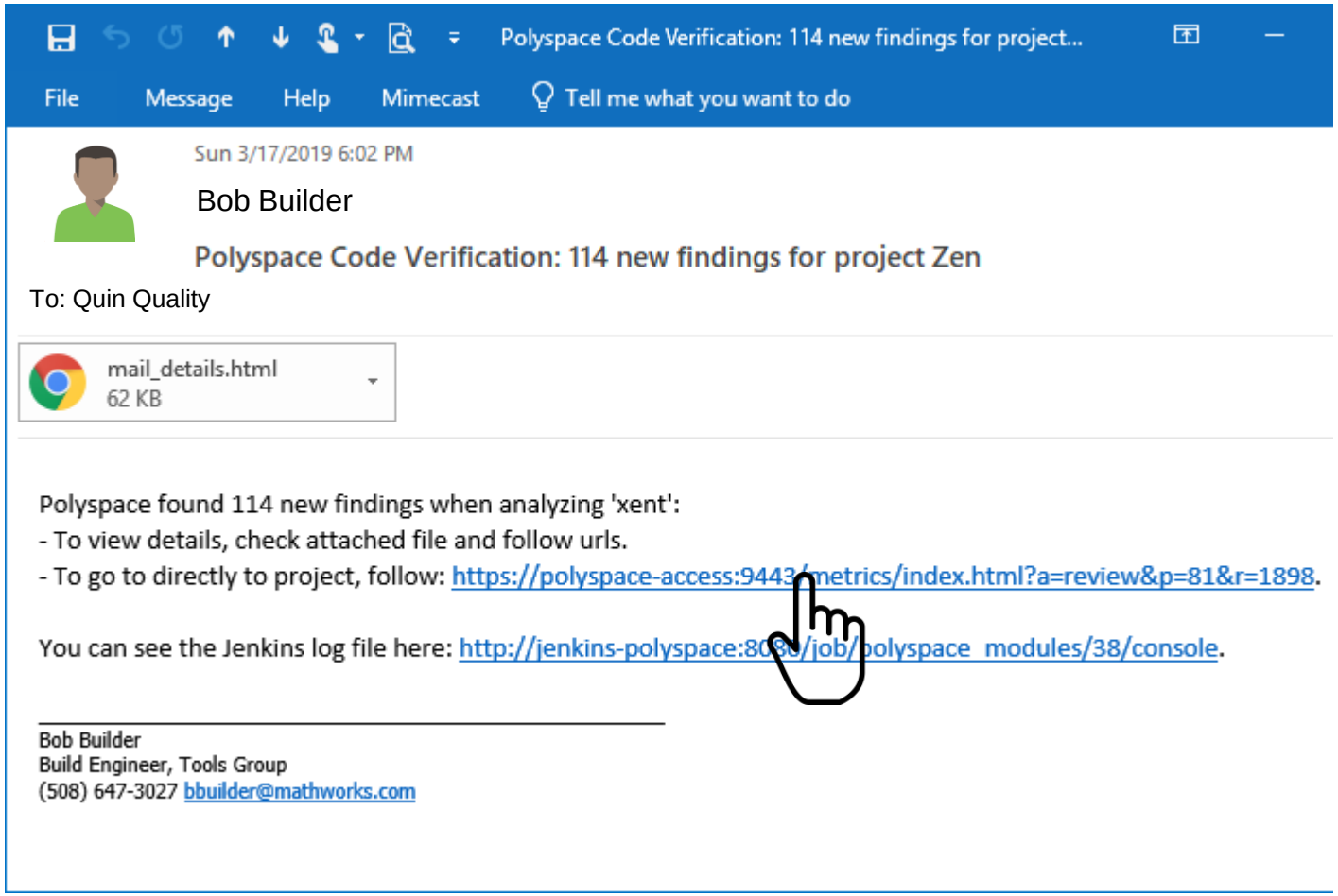
Doug – 项目经理
负责项目软件质量

# Bob – 构建工程师
# 负责持续集成流程配置



# Quinn – 质量工程师
# 负责软件问题归类

- 从昨晚Jenkin发起的Polyspace分析任务中收到邮件通知
- 邮件显示在其项目中发现了一些软件问题
- 点击邮件链接进入Polyspace Access查看具体问题



Quinn – 质量工程师
负责软件问题归类

Sign In

localhost:9443/authn/signin?&title=/static/images/polyspace_title_logo.svg&continue=%2F

POLYSPACE®

Sign in to your account

Username

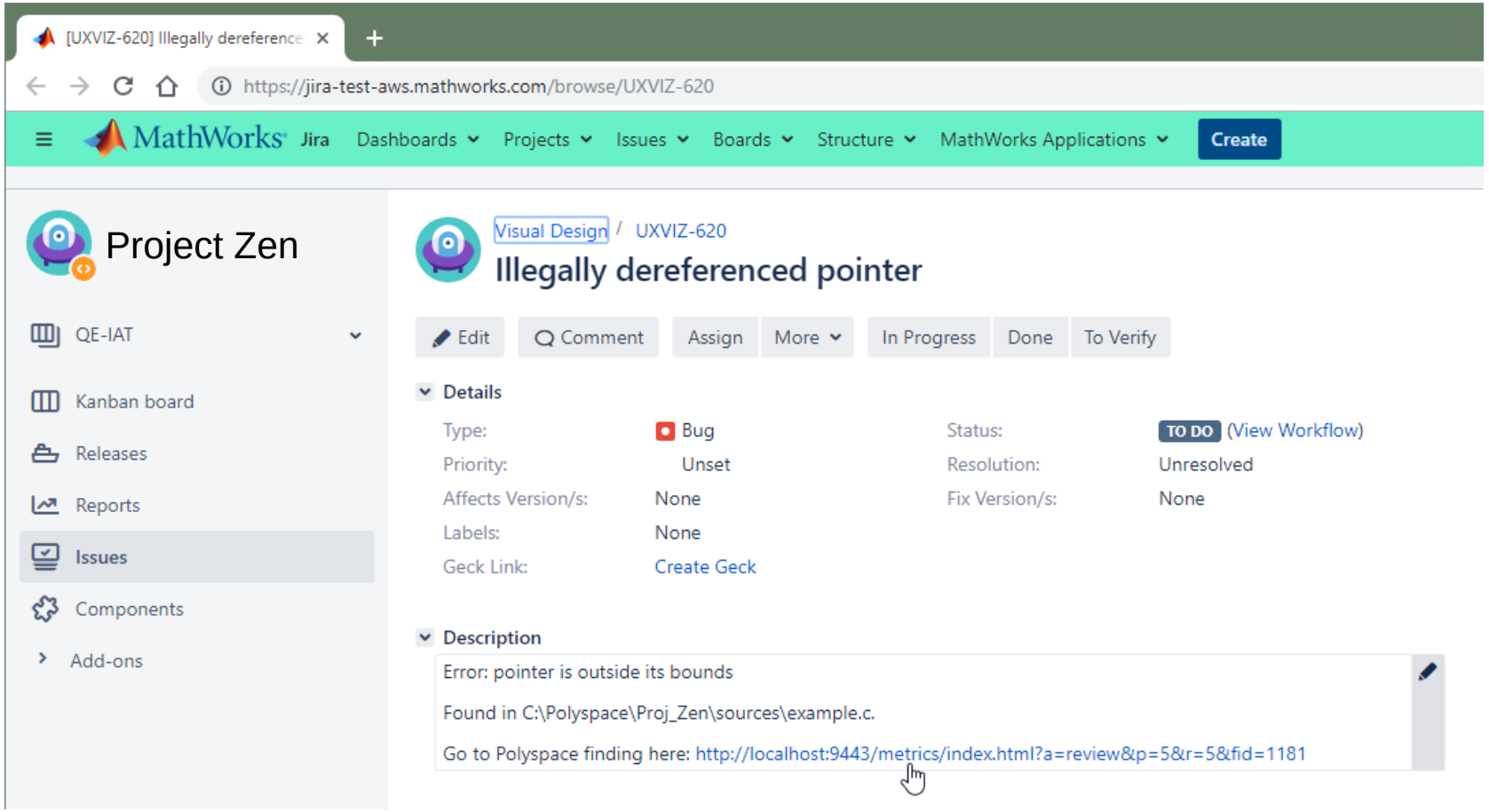
Password

Forgot password?

Sign in

# Dara – 代码开发工程师
# 负责代码编写和问题修复

- 查看到在Jira系统中被分配了两个问题标签
- 打开第一个JIRA标签并点击Polyspace Access链接



# Dara – 代码开发工程师
# 负责代码编写和问题修复

[UXVIZ-623] Illegally dereference

https://jira-test-aws.mathworks.com/browse/UXVIZ-623

MathWorks Jira Dashboards Projects Issues Boards Structure MathWorks Applications Create Search

Visual Design

QE-IAT

Kanban board

Releases

Reports

Issues

Components

Add-ons

Visual Design / UXVIZ-623

## Illegally dereferenced pointer

Edit Comment Assign More In Progress Done To Verify Export

**Details**

| | | | |
|---|---|---|---|
| Type: | Bug | Status: | TO DO (View Workflow) |
| Priority: | Unset | Resolution: | Unresolved |
| Affects Version/s: | None | Fix Version/s: | None |
| Labels: | None | | |
| Geck Link: | Create Geck | | |

**People**

Assignee: Unassigned
Assign to me
Reporter: Jay Abraham
Watchers: 1 Stop watching this issue

**Dates**

Created: 2 hours ago
Updated: 2 hours ago

**Agile**

View on Board

**Description**

Error: pointer is outside its bounds

Found in C:\Polyspace\Proj_Zen\sources\example.c.

Go to Polyspace finding here: http://localhost:9443/metrics/in Click to edit eview&p=6&r=7&fid=3949

Critical run-time error, needs investigation.

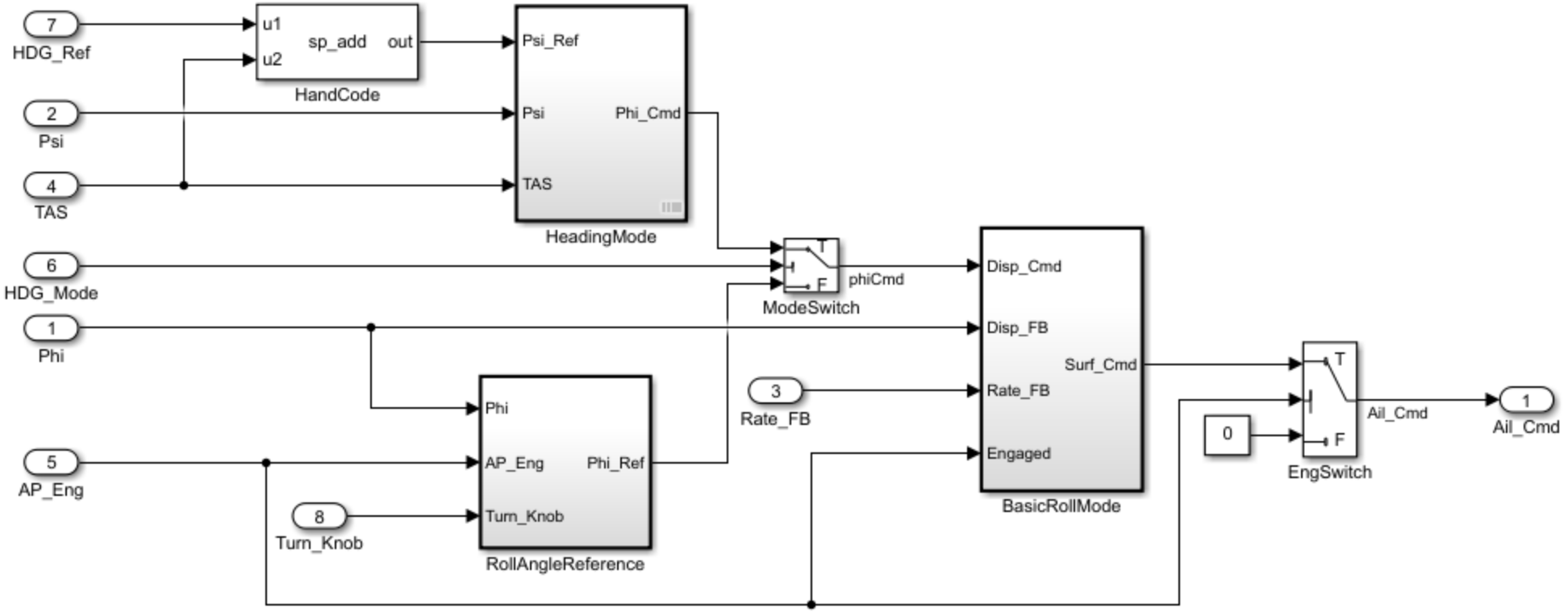
**Attachments**

Drop files to attach, or browse.

localhost:9443/metrics/index.html?a=review&p=6&r=7&fid=3949

Get Help / Give Feedback

# Martin – 模型开发工程师
# 负责模型开发和代码生成



# Martin – 模型开发工程师
# 负责模型开发和代码生成

Code Generation Report

Find: Match Case

**Contents**

- Summary
- Subsystem Report
- Code Interface Report
- Traceability Report
- Static Code Metrics Report
- Code Replacements Report
- Coder Assumptions

**Generated Code**

- [-] **Main file**
  - ert_main.c
- [-] **Model files**
  - rtwdemo_roll.c
  - rtwdemo_roll.h
- [+] **Shared files (1)**
- [+] **Other files (1)**

## Code Generation Report for 'rtwdemo_roll'

### Model Information

| | |
|---|---|
| Author | The MathWorks, Inc. |
| Last Modified By | The MathWorks, Inc. |
| Model Version | 1.162 |
| Tasking Mode | SingleTasking |

Configuration settings at time of code generation

### Code Information

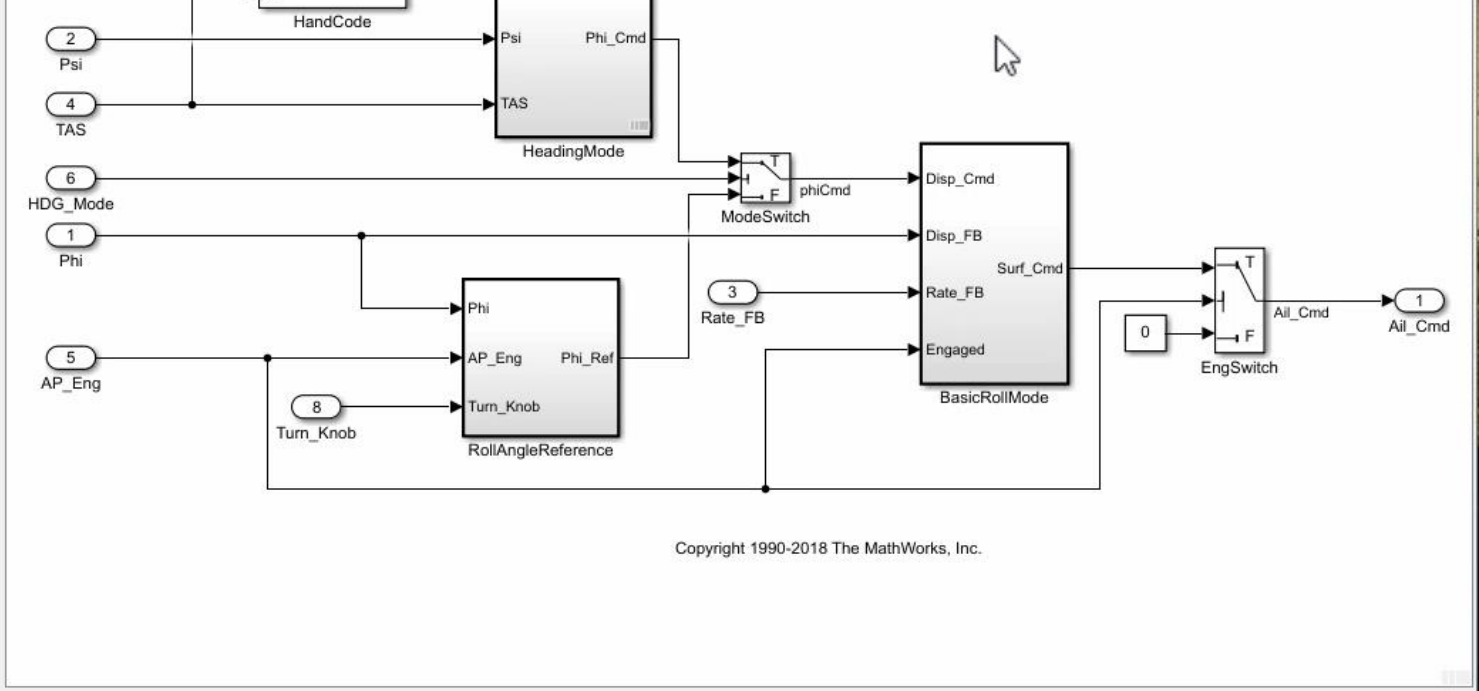
| | |
|---|---|
| System Target File | ert.tlc |
| Hardware Device Type | Intel->x86-64 (Windows64) |
| Simulink Coder Version | 9.1 (R2019a) 23-Nov-2018 |
| Timestamp of Generated Source Code | Wed Apr 10 10:36:32 2019 |
| Location of Generated Source Code | C:\Work\MATLAB\ML_Expo\rtwdemo_roll_ert_rtw\ |
| Type of Build | Model |
| Memory | Global Memory: 39(bytes) Maximum Stack: |

OK Help

rtwdemo_roll - Simulink

File Edit View Display Diagram Simulation Analysis Code Tools Help

rtwdemo_roll




Ready 115% FixedStepDiscrete

# Doug – 项目经理
# 负责项目软件质量

Polyspace

localhost:9443/metrics/index.html?a=metrics&p=1

DASHBOARD

jabraham

Project Overview | Run-time Checks | Code Metrics | MISRA C:2012 | Layout | Open in Desktop | Review

DASHBOARDS | ENVIRONMENT | REVIEW

PROJECT EXPLORER

- public
  - Proj_Zen
  - Test_Area

PROJECT DETAILS

**Project**

| | |
|---|---|
| **Name** | public |
| **Tools** | Code Prover |
| **Coding Standards** | Custom Rules, MISRA C:2012 |
| **Number of Runs** | 6 |

SUPPORT REPORT

Project Overview | Quality Objectives | Run-time Checks | Code Metrics

Summary — public

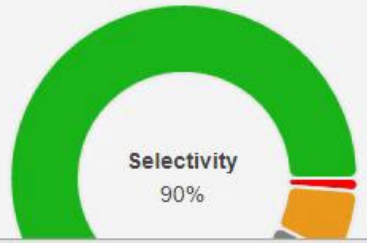
**Open Issues**

| | |
|---|---|
| Open | 104 |
| New | 0 |
| Assigned To Me | 0 |
| Unassigned | 104 |

**Code Metrics**

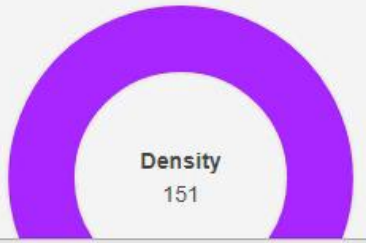
| | |
|---|---|
| Sub-project(s) | 2 |
| Number of Files | 7 |
| Number of Lines Without Comment | 450 |
| Cyclomatic Complexity | 6 |

**Run-time Checks** — Open 32




| | |
|---|---|
| Red | 4 |
| Orange | 21 |
| Gray | 8 |
| Green | 300 |

**Coding Standards** — Open 68




To Do 68

public

# 总结

- 助力打造高质量软件
  - 功能安全
  - 网络安全
- 无缝支持软件开发流程
  - 自动化构建
  - 问题追踪
- 协同合作共享结果
  - 基于网页的轻量级技术
  - 灵活切换不同颗粒度