MathWorks

MathWorks
Accelerating the pace of engineering and science

5月 - 北京、 上海、 深圳

2014 MATLAB 巡回研讨会

迎接复杂性设计的挑战

使用Polyspace进行软件代码运行错误检查和验证

李春彦
应用工程师
MathWorks China

内容

- Polyspace背景介绍
- Polyspace产品功能及使用技巧
- 在基于模型设计开发流程中使用Polyspace
- Polyspace及相关工具对认证的支持

Hello, Polyspace!

1996

A 研
实
成
客

199

成

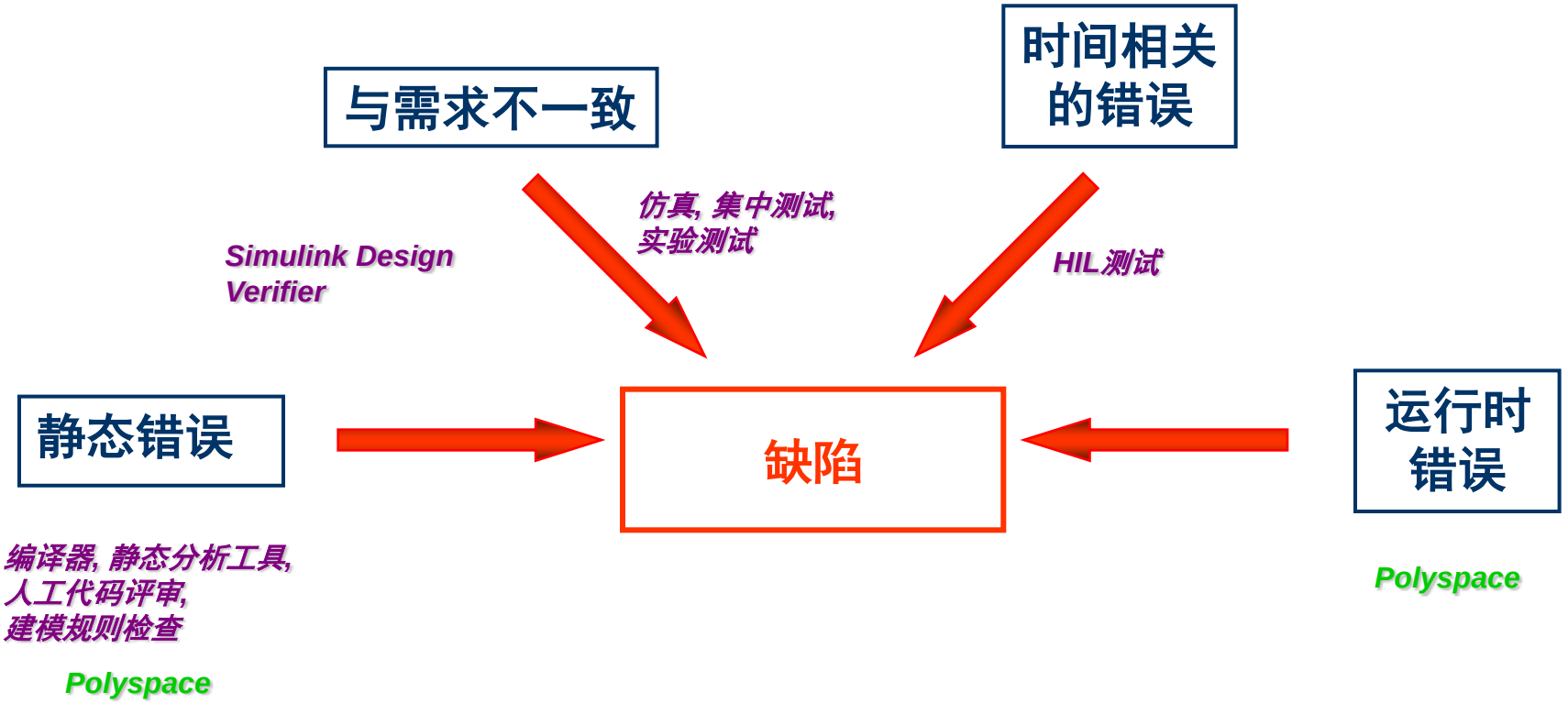
200 lain Deutsch

MathWorks® 收购Polyspace

Polyspace：静态代码分析工具

- 作用：解决软件的鲁棒性问题，确保软件的安全性和可靠性；
- 能够做什么？
  - 发现编码错误
  - 检查编码规则一致性（MISRA/JSF）
  - 提供代码静态度量信息（圈复杂度等）
  - 证明代码中确定存在或者确定不存在的错误
  - 表示代码已经达到了预期的质量水平
  - 认证帮助：DO-178 C, ISO 26262, …

嵌入式系统的缺陷



什么样的工具和流程能够发现这些缺陷?

你能证明系统不存在特定缺陷吗?

常见的软件代码错误

- 运行时错误
  - 40%左右的软件错误为运行时错误；
  - 运行时错误是软件代码的潜在缺陷，并不频繁出现；
  - 运行期错误会导致系统崩溃和意外的行为。
- 并发问题
- 编程错误
- 不可达代码
- 静态或者动态内存错误

Polyspace C / C ++产品系列

- Polyspace Bug Finder
  - 快速找到嵌入式软件中的错误
  - 检查代码符合MISRA和JSF编码规则
  - 供软件工程师日常使用

- Polyspace Code Prover
  - 证明代码是安全和可靠的(是否包含运行时错误)
  - 软件组件的深层验证
  - 执行QA对生产就绪代码的签收

还支持Ada语言代码证明

使用Polyspace迎接复杂系统设计的挑战

现代汽车动力系统
500-1,000(KLOC)

波音787飞控系统
6,500 (KLOC)

航天器*
3-1,700 (KLOC)

- 减少软件故障风险
  - 应用于质量关键、业务关键或者高完整性系统中；
  - 减少产品代码中可能导致系统崩溃的潜在错误。
- 缩短软件测试和验证周期
  - 减少软件的鲁棒性测试；
  - 显著缩短系统开发周期。

*Automated Software Verification & Validation: An emerging approach for ground operations
Bell and Brat, NASA

内容

- Polyspace背景介绍
- Polyspace产品功能及使用技巧
- 在基于模型设计开发流程中使用Polyspace
- Polyspace及相关工具对认证的支持

如何找到代码的运行时错误？

```
0 k=ioread_i32();
1 I=2;
2 J=K+5;
3 while (I<10) {
4     I=I+1;
5     J=J+3;
6 }
7
8 … /(I-J);
```

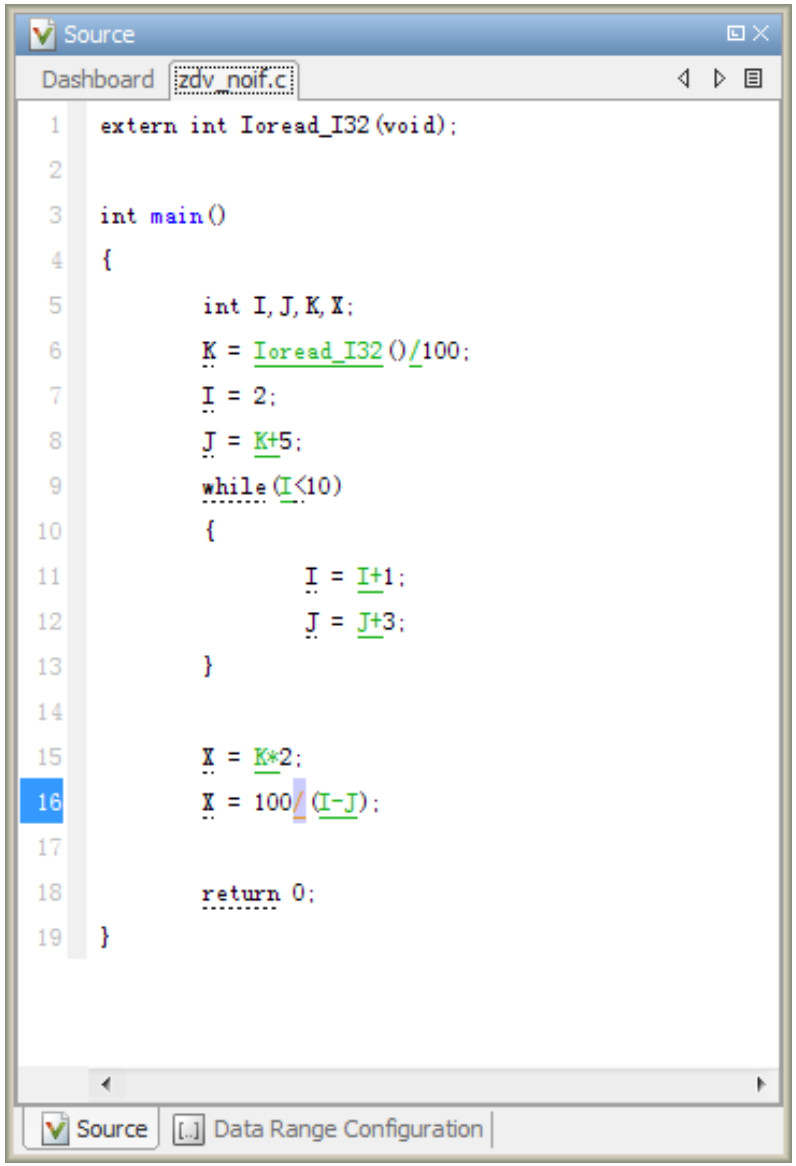
当k=-19时，第8行会出现除零错误

第8行有没有潜在的“除零”风险？

Polyspace是如何显示的？

```
0       k=ioread_i32();
1       I=2;
2       J=K+5;
3       while (I<10) {
4           I=I+1;
5           J=J+3;
6       }
7
8       … /(I-J);
```

Polyspace



Polyspace的结果

A colorful world

Proven

Green: reliable
safe pointer access

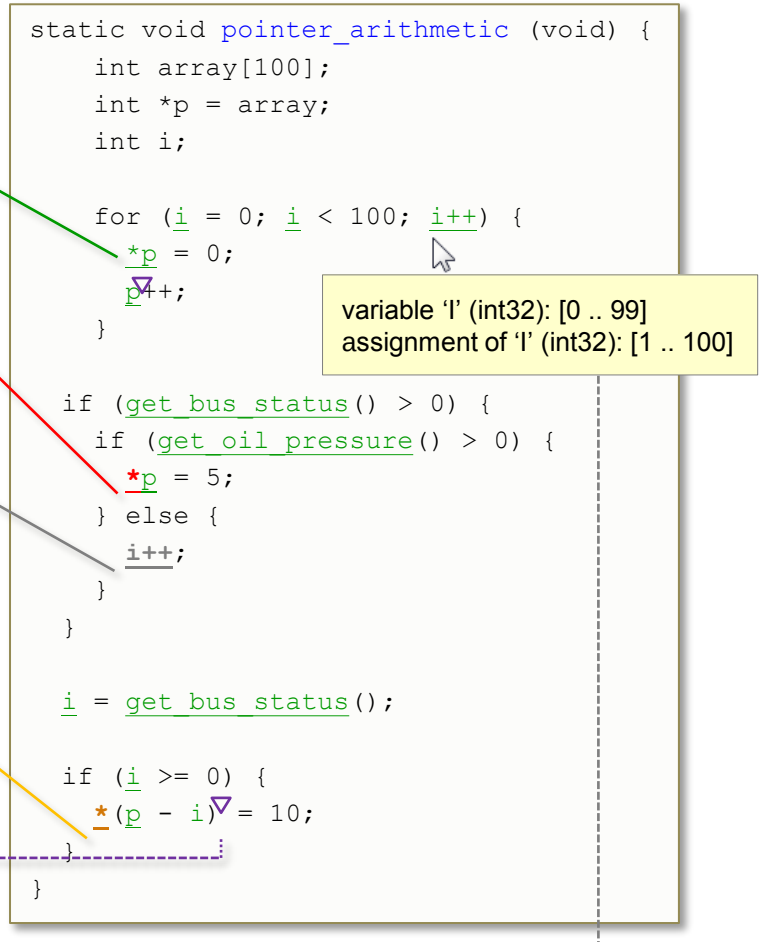
Red: faulty
out of bounds error

Gray: dead
unreachable code

Orange: unproven
may be unsafe for some conditions

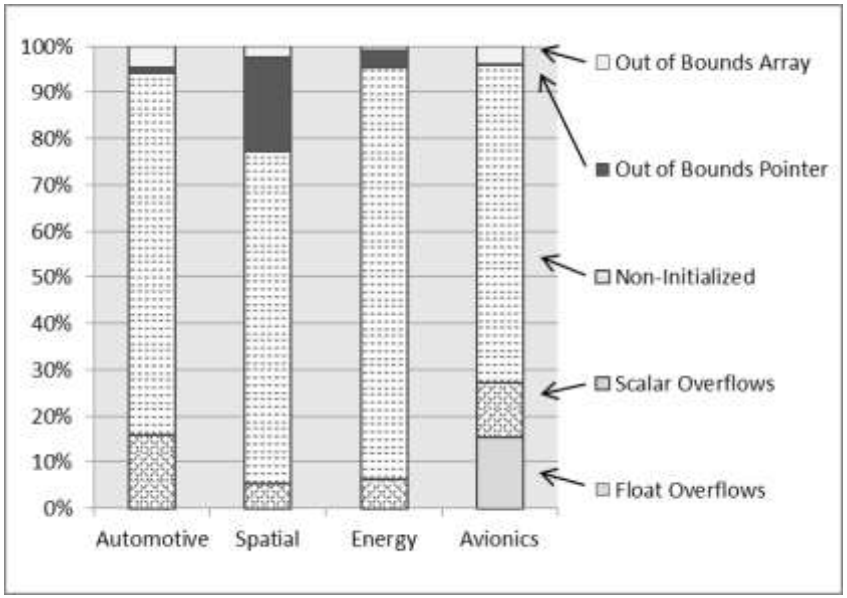
Purple: violation
MISRA-C/C++ or JSF++ code rules

Range data
tool tip

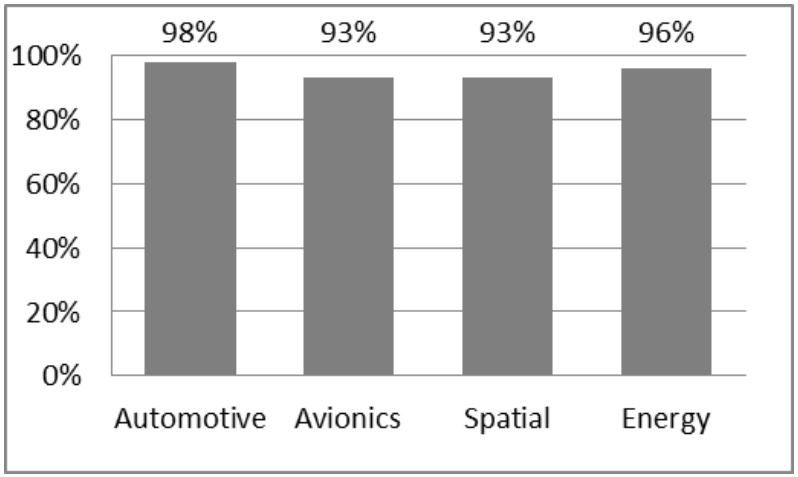


Polyspace行业应用

- Polyspace对于任何类型的代码都是高效的



不同应用的代码缺陷分布



不同应用的代码证明比例

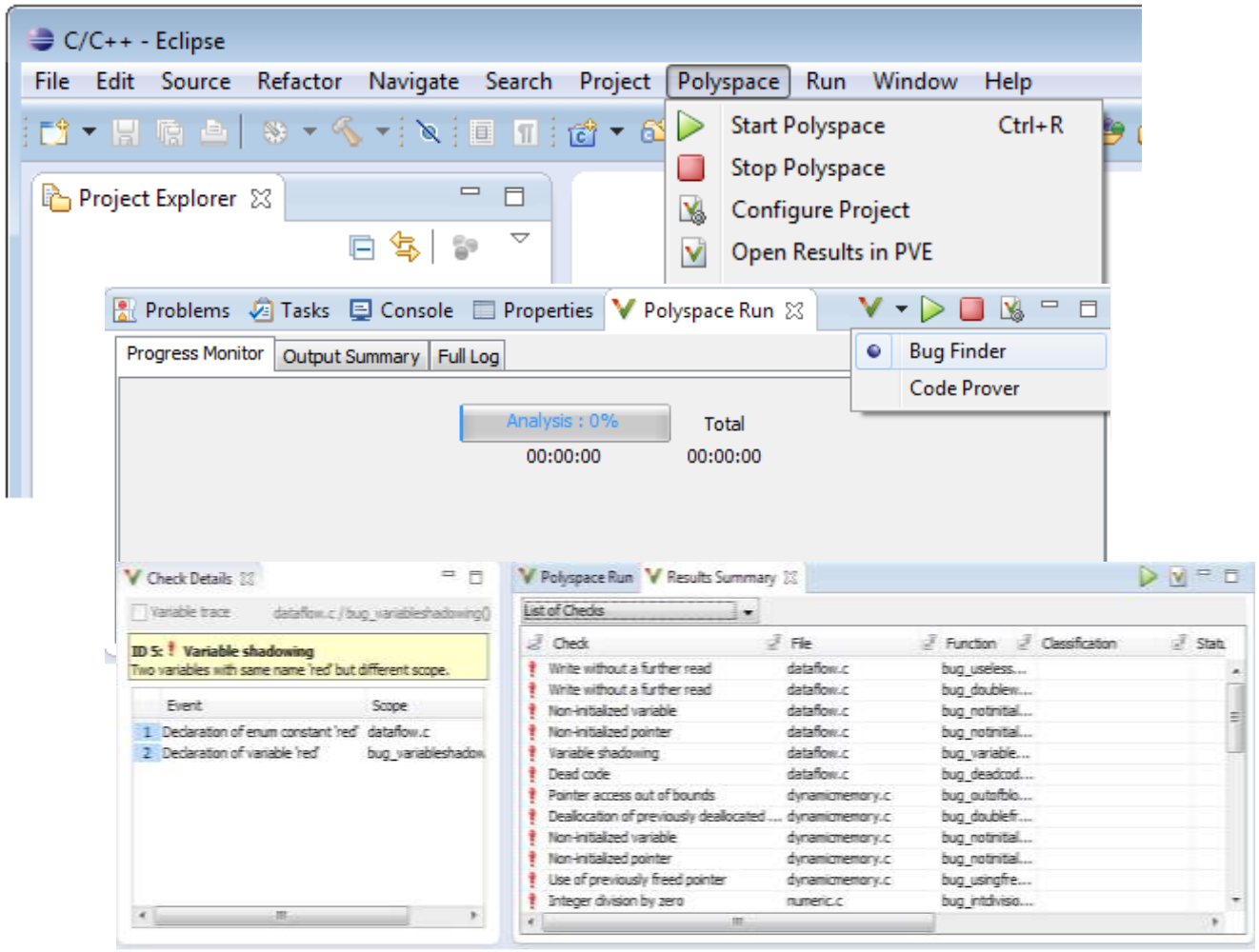
98%意味着只有2%的检查是橙色的

Polyspace采用的形式化方法是通用的

Polyspace产品的功能

- 第三方IDE环境的集成
  - Visual Studio，Eclipse
- 能够与用户的构建环境进行集成并自动生成Polyspace配置文件
  - polyspace-configure
- 自动代码生成工具的集成
  - Embedded Coder, Target Link, IBM Rational Rhapsody
- 支持在集群上进行部署安装
  - 利用集群优势加快Polyspace Code Prover的验证速度

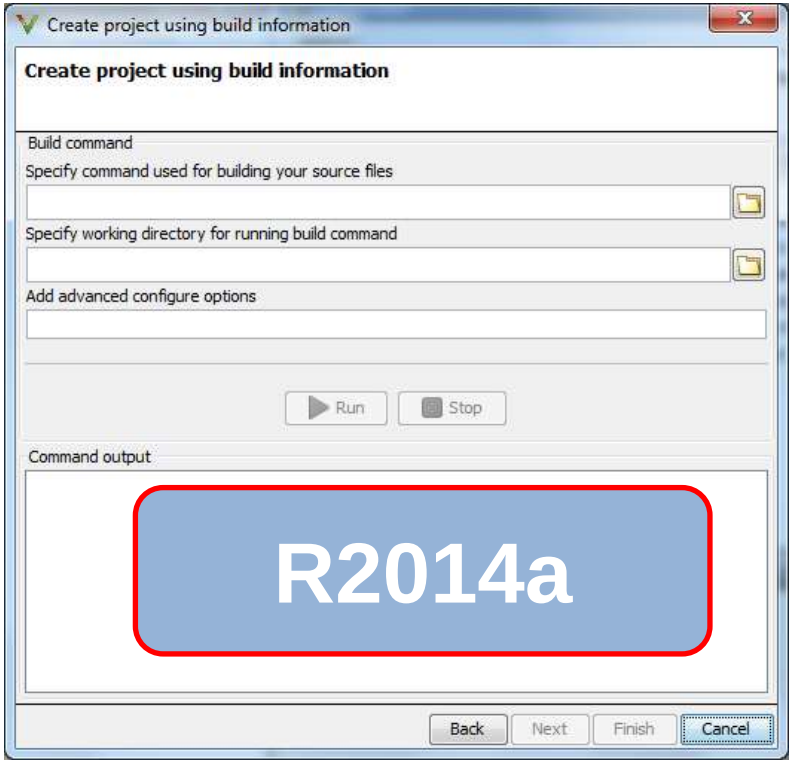
Polyspace支持与第三方IDE环境集成



- 支持的环境有Eclipse，Visual Studio

Polyspace配置工具：polyspace-configure

- 与用户的构建环境进行集成，自动生成Polyspace工程或配置文件
- 命令：polyspace-configure
- 使用方法：

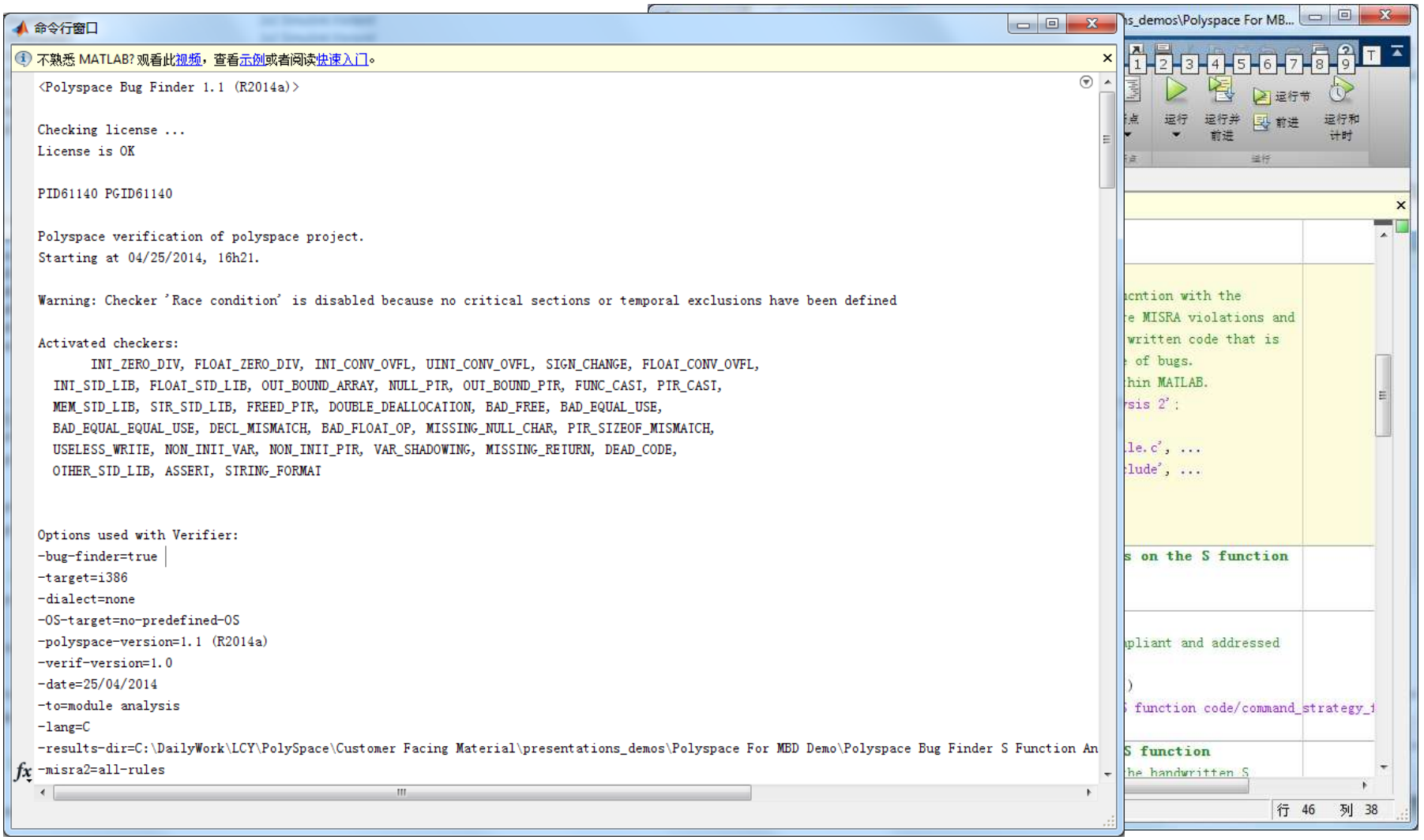



```
polyspace-configure  -prog myProject make -B targetName buildOptions
```

Video

Polyspace支持与自动代码生成工具进行集成

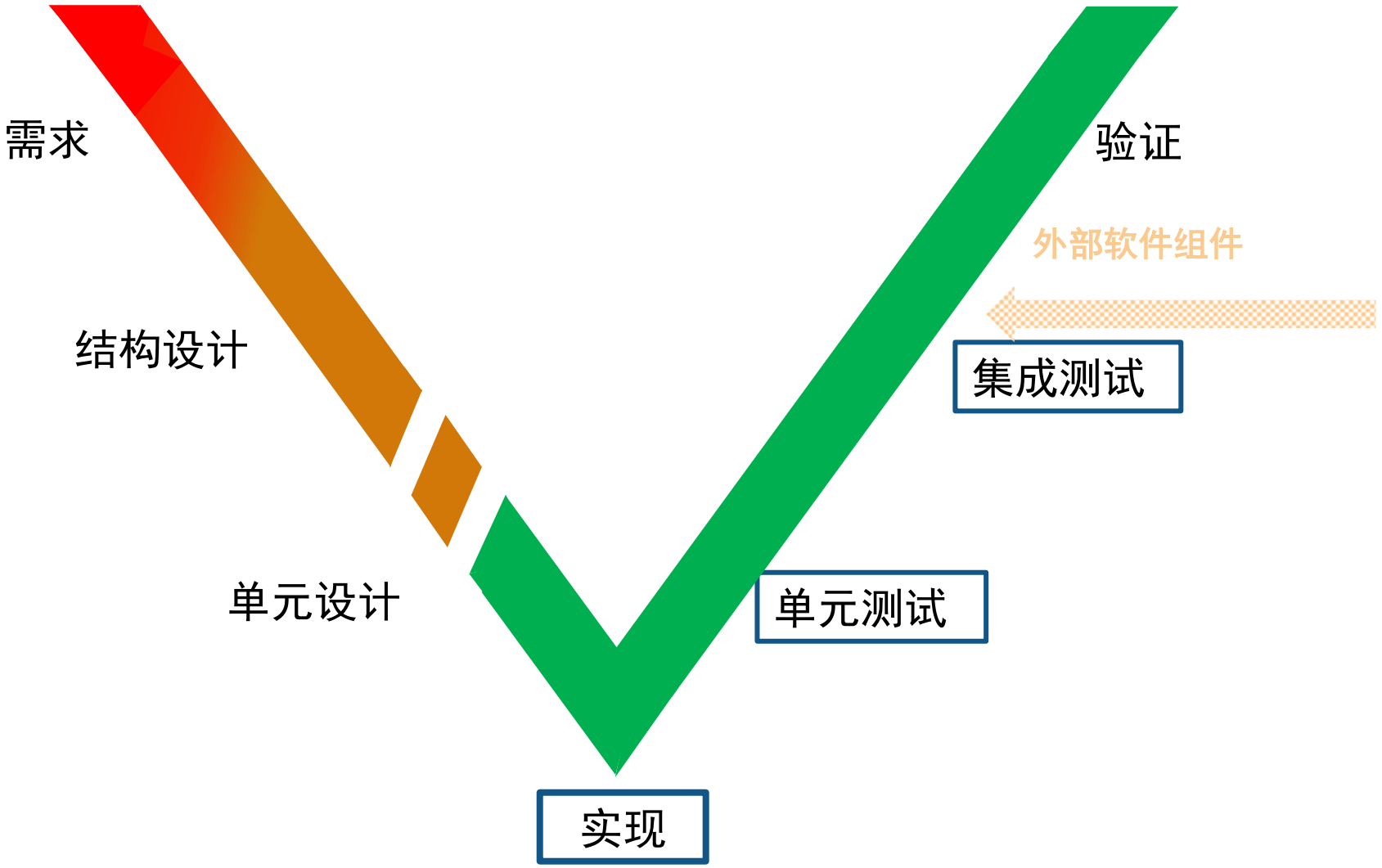
- Embedded Coder, TargetLink, IBM Rational Rhapsody



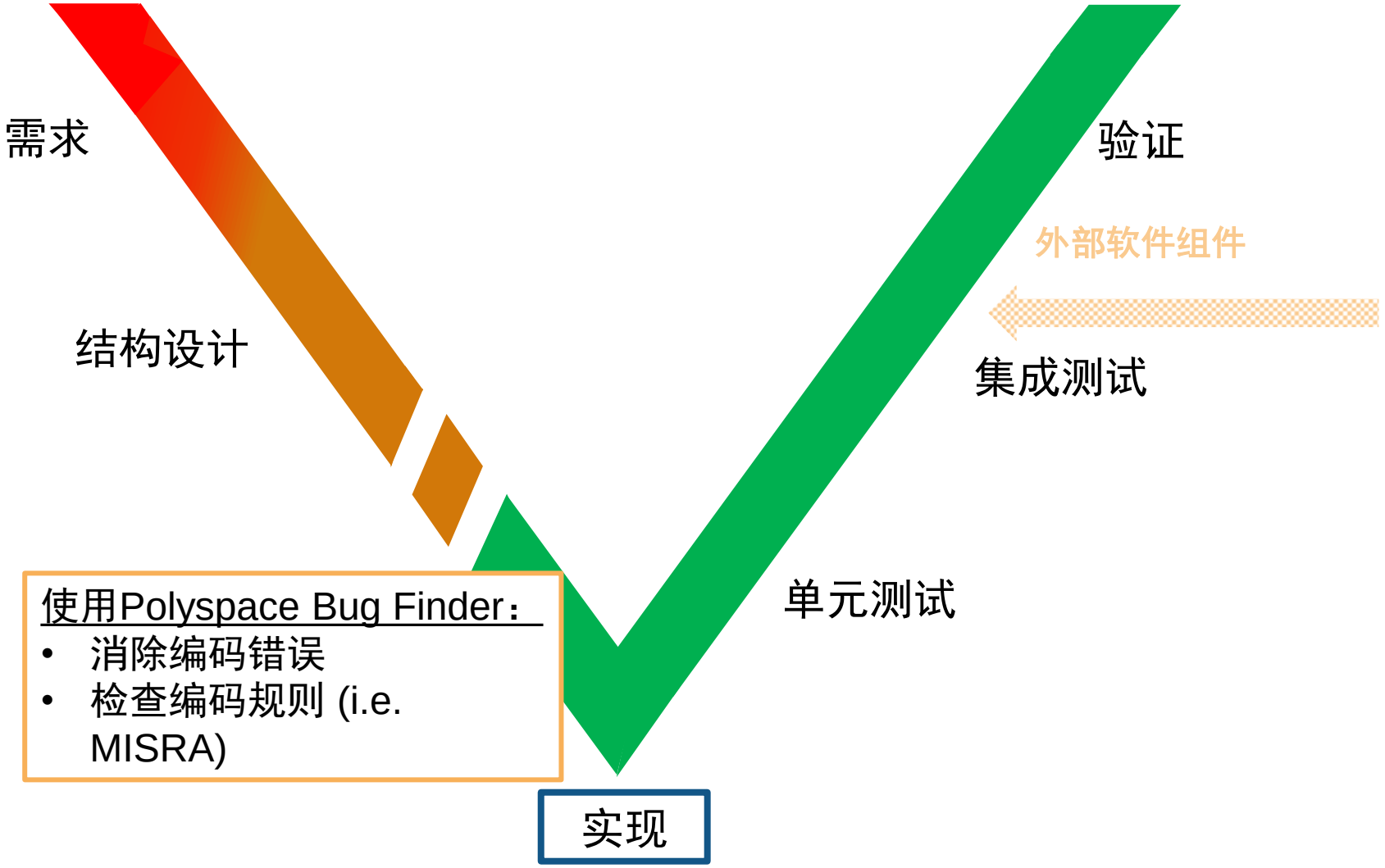
内容

- Polyspace背景介绍
- Polyspace产品功能及使用技巧
- 在基于模型设计开发流程中使用Polyspace
- Polyspace及相关工具对认证的支持

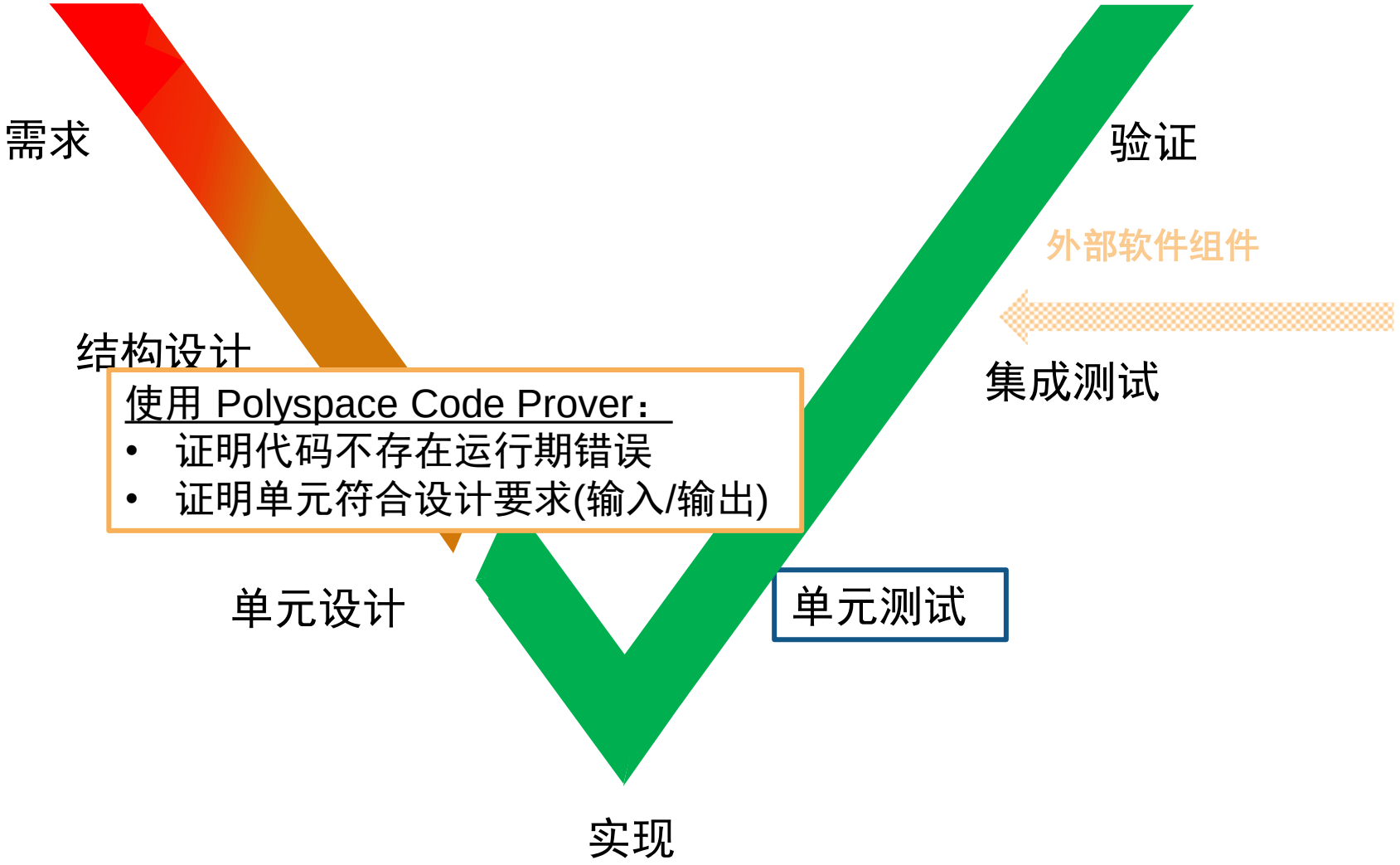
在开发流程中引入Polyspace



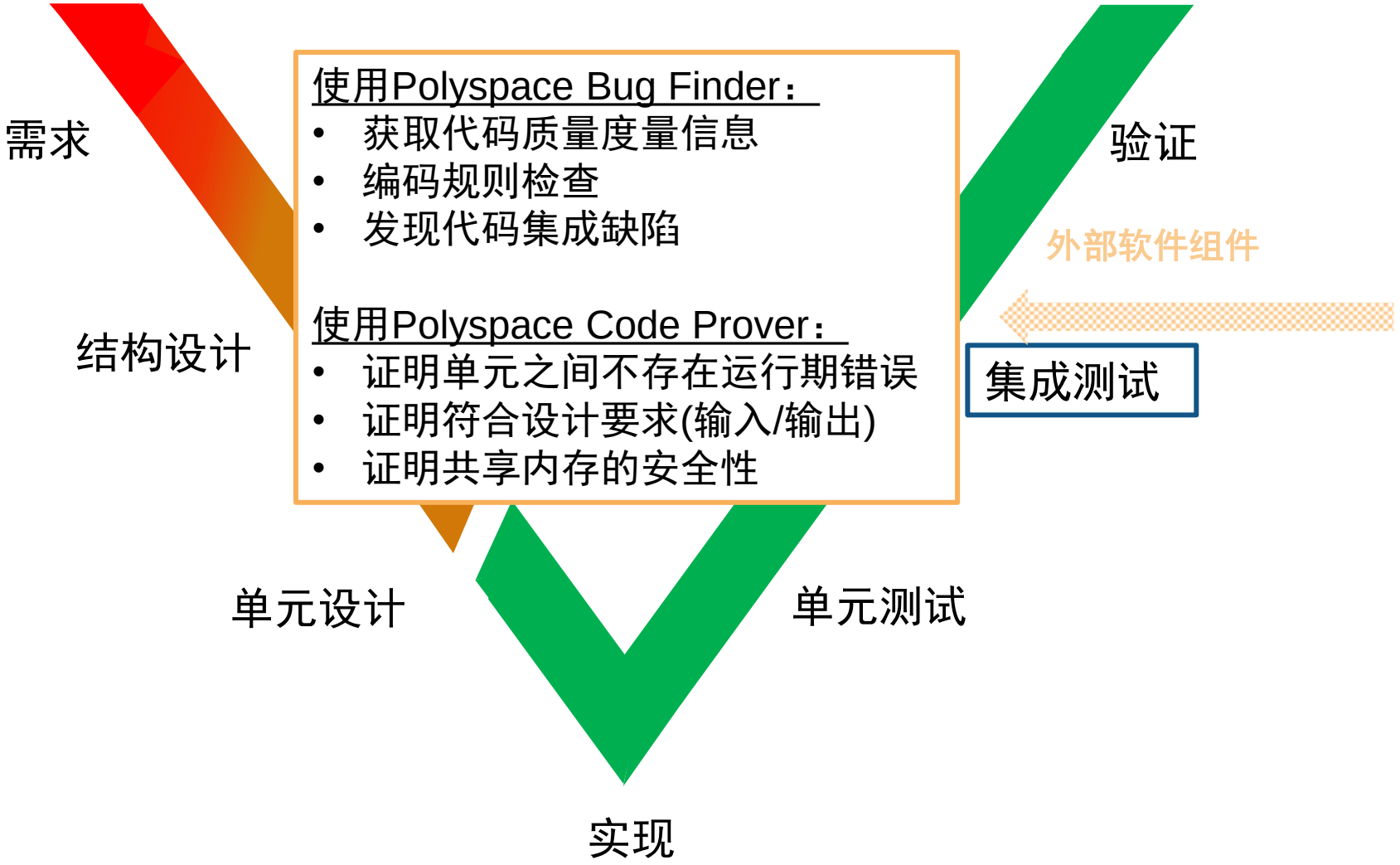
在开发流程中引入Polyspace



在开发流程中引入Polyspace



在开发流程中引入Polyspace



在开发流程中引入Polyspace

需求

单元设计

Polyspace Code Verification

SQO Report for: Demo_C_Workflow

Report Author: cpreve

发布Polyspace 结果

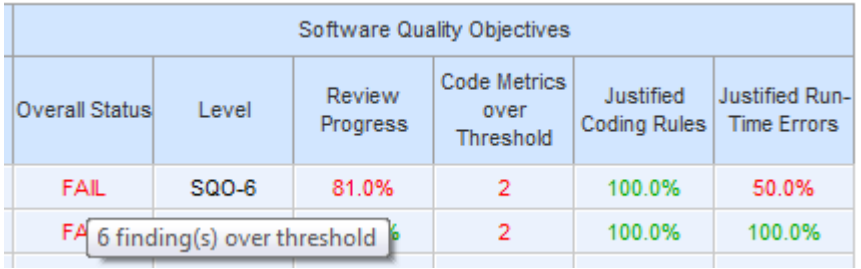
- Polyspace Web Metrics
- 生成报告

内容

- Polyspace背景介绍
- Polyspace产品功能及使用技巧
- 在基于模型设计开发流程中使用Polyspace
- Polyspace及相关工具对认证的支持

软件质量目标(Software Quality Objectives)

- 在Polyspace中设定软件质量目标
  - 确定文件、模块或者组件是否达到了预定质量目标
- 定义软件质量目标门槛
  - 软件代码度量信息
  - 违反编码规则数量
  - 红色、灰色、橙色的检查数量
- 将SQO作为流程指南
  - 根据质量目标来指定工具使用和流程改进的计划



使用Polyspace 进行DO-178 认证

MathWorks can provide a DO-178 Qualification Kit to help qualify Polyspace for your DO-178 certification projects

- 使用DO Qualification Kit, 您可以
  - 证明Polyspace 产品可以用于满足DO-178C 和相关标准如DO-330产品开发活动
  - 帮助获得认证
- 套件包括：
  - 文档，测试用例和流程指导书
  - 工具资质证明计划，工具的使用要求和其它材料
  - 允许针对特定项目对工具认证套件进行修改

Polyspace Bug Finder is a Criteria 3 Tool for DO-178C
Polyspace Code Prover is a Criteria 2 Tool for DO-178C

Polyspace支持DO-178B/C认证

- 代码与软件设计结构一致(architecture)
  - 数据字典，函数调用树
- 代码是可以被验证的(Verifiable)
  - 发现不可达代码
- 代码要符合编码规则
  - MISRA C, MISRA C++, 或 JSF++ 规则检查
  - 静态度量信息，例如圈复杂度(编码标准的一部分)
- 代码是准确的(Accurate)和一致的(Consistent)
  - 发现代码中的运行期错误，例如除零、溢出、变量未初始化等

使用Polyspace for IEC 61508, ISO 26262, EN 50128, IEC 62304认证

MathWorks can provide a Certification Kit for IEC 61508 related standards

- 使用认证套件, 您可以
  - 在推荐流程下使用Polyspace来验证您的代码
  - 帮助获得认证
- 套件包括
  - 工具资质证书，审核材料和测试套件
  - 可以使用Polyspace的工具证明材料
  - 推荐的工作流程


CERTIFICATE

Polyspace可以提供的帮助