

# 2022 MathWorks 中国汽车年会

基于模型的持续集成加速车载软件开发和交付

胡乐华, MathWorks中国



# 内容提要

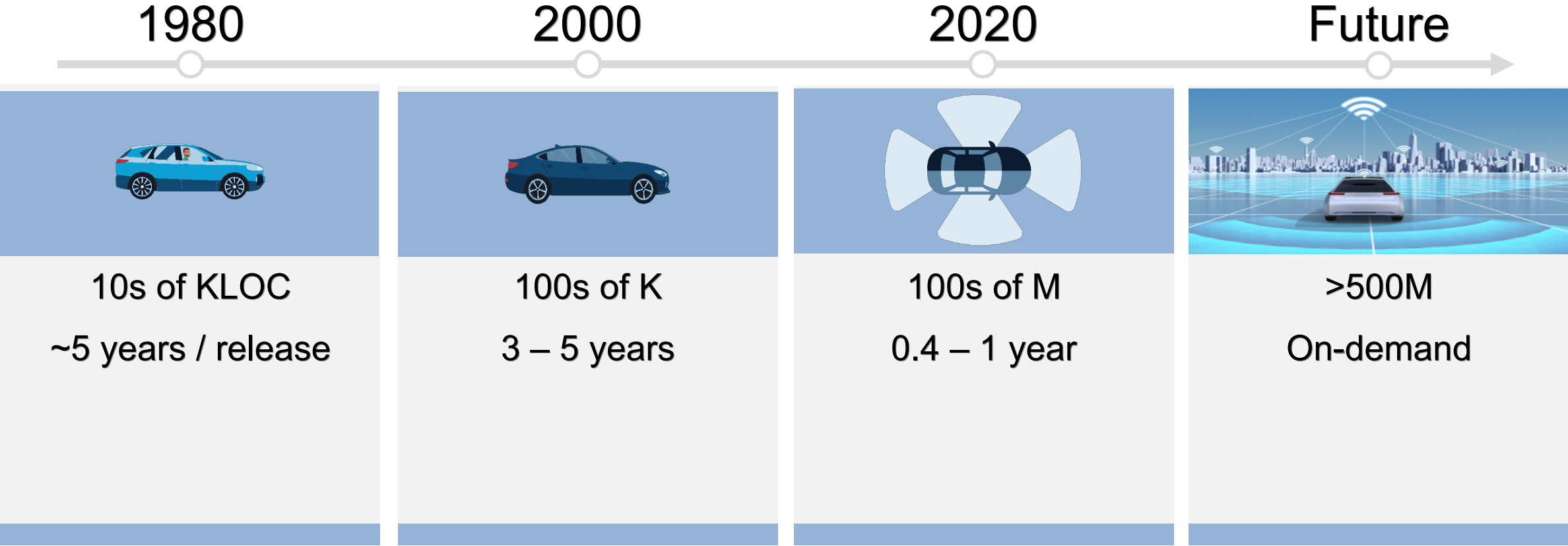
车载软件系统开发新挑战

基于模型设计用于CI的流程

云端部署CI系统用于大规模系统

# 车载软件系统开发新挑战

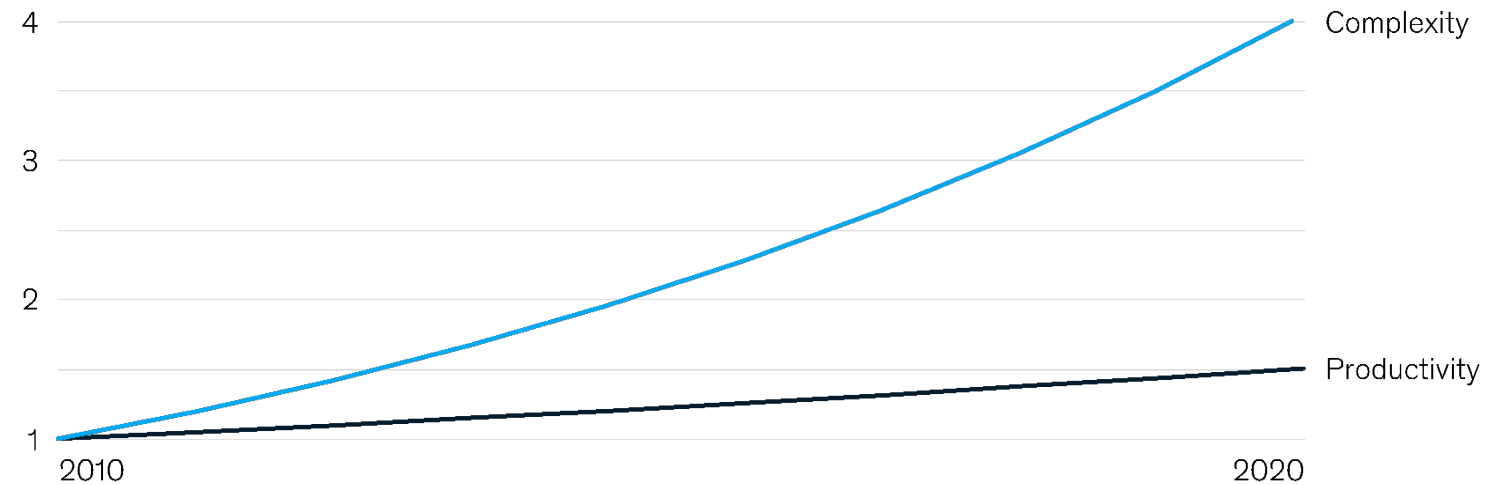
# 汽车代码量增长迅猛



# 数字化转型

## 软件复杂度的增长远超开发效率

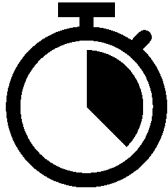
Relative growth of software complexity and productivity over time, indexed for automotive features



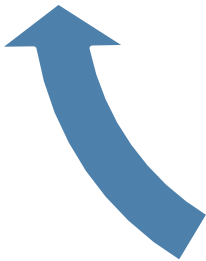
Source: McKinsey's SoftCoster embedded software project database

# 车载嵌入式系统开发面临更多挑战

Complexity and Size  
Low Productivity  
C++, AUTOSAR  
Staffing SW developers

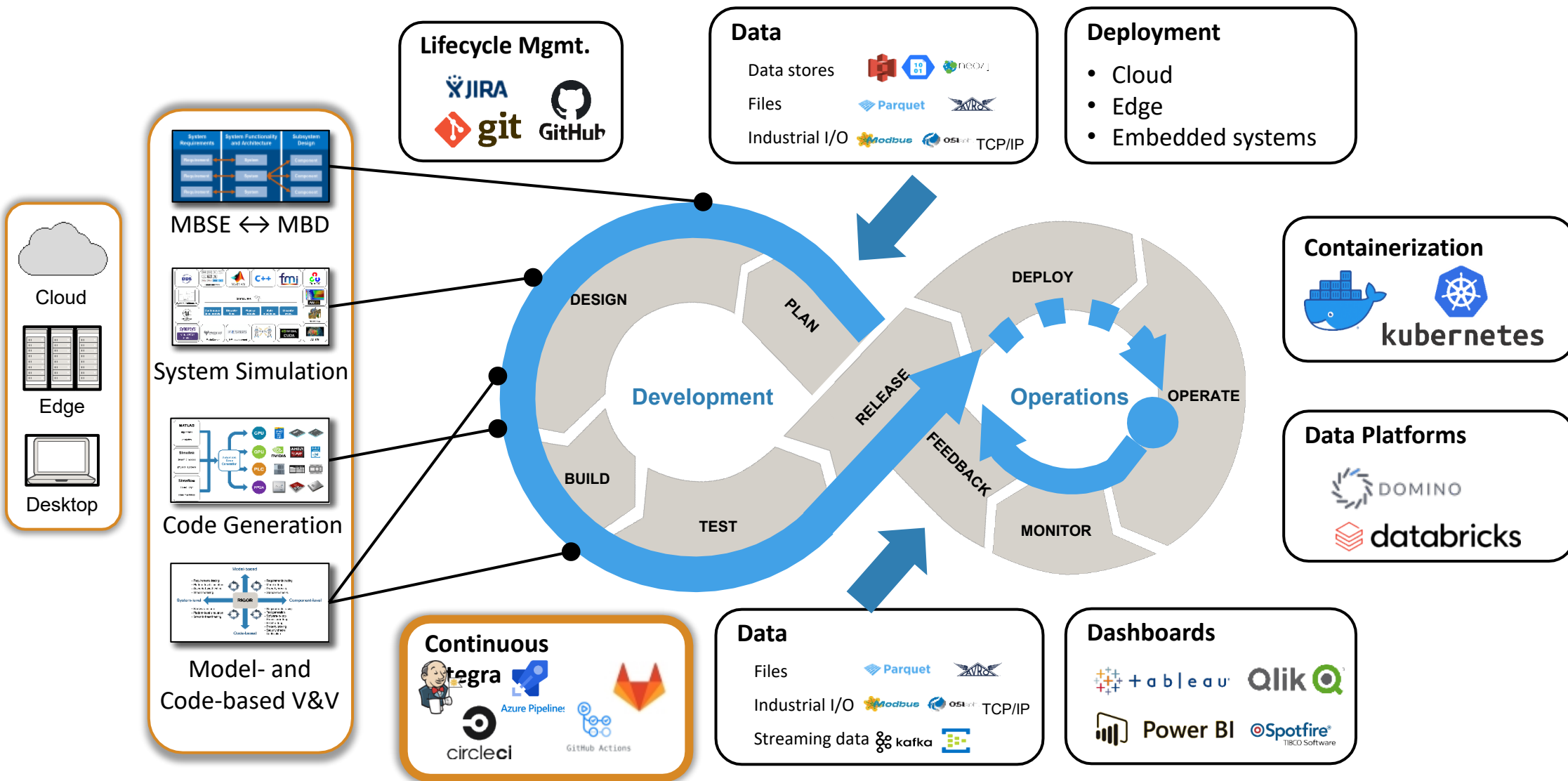


Faster delivery



Safety and Cybersecurity  
ISO 26262, ISO/SAE 21434

# 持续集成是 DevOps 的重要环节



# 基于模型设计促进高效的 DevOps

- DevOps 目标: 缩短提交代码到产品更新的时间, 并保证高质量、高安全和安全标准的合规性

ISO 26262

Functional Safety

ISO/SAE 21434

Cybersecurity for Road Vehicles

ISO 21448

Safety of the intended functionality (SOTIF)

	High Performers	Low Performers
Lead Time	<1hour	>6months
Change Failure Rate	0-5%	15-30%

Source: [state-of-devops-2021.pdf \(google.com\)](https://www.google.com/search?q=state-of-devops-2021.pdf)

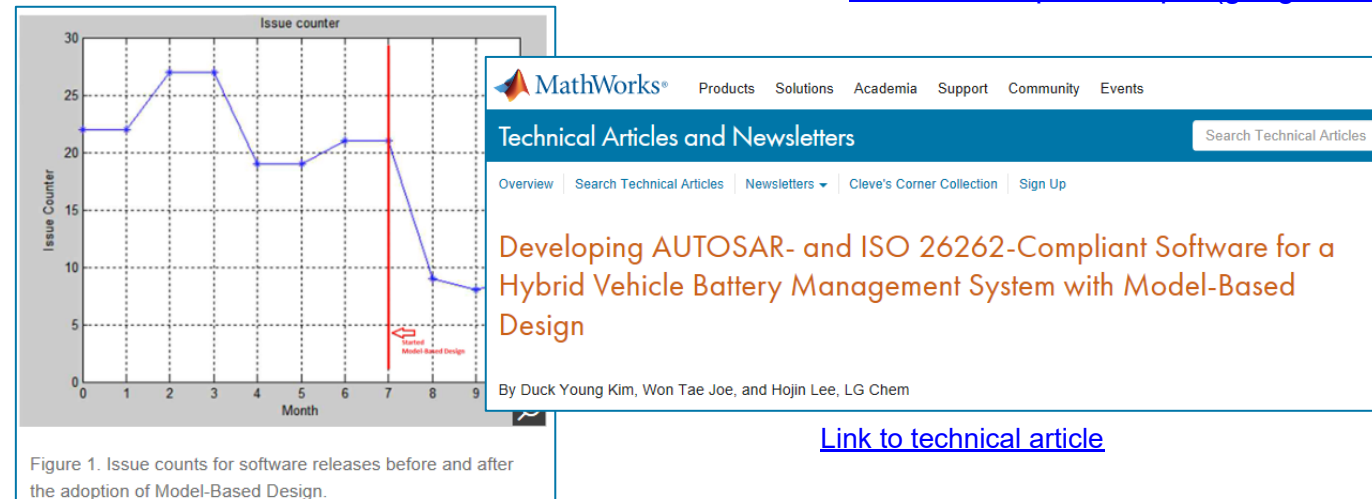


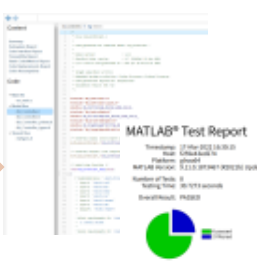
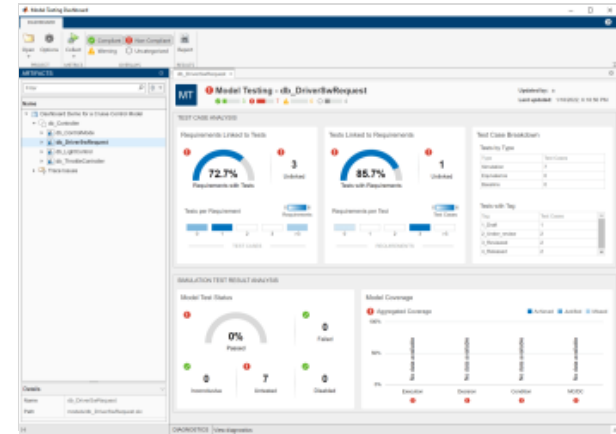
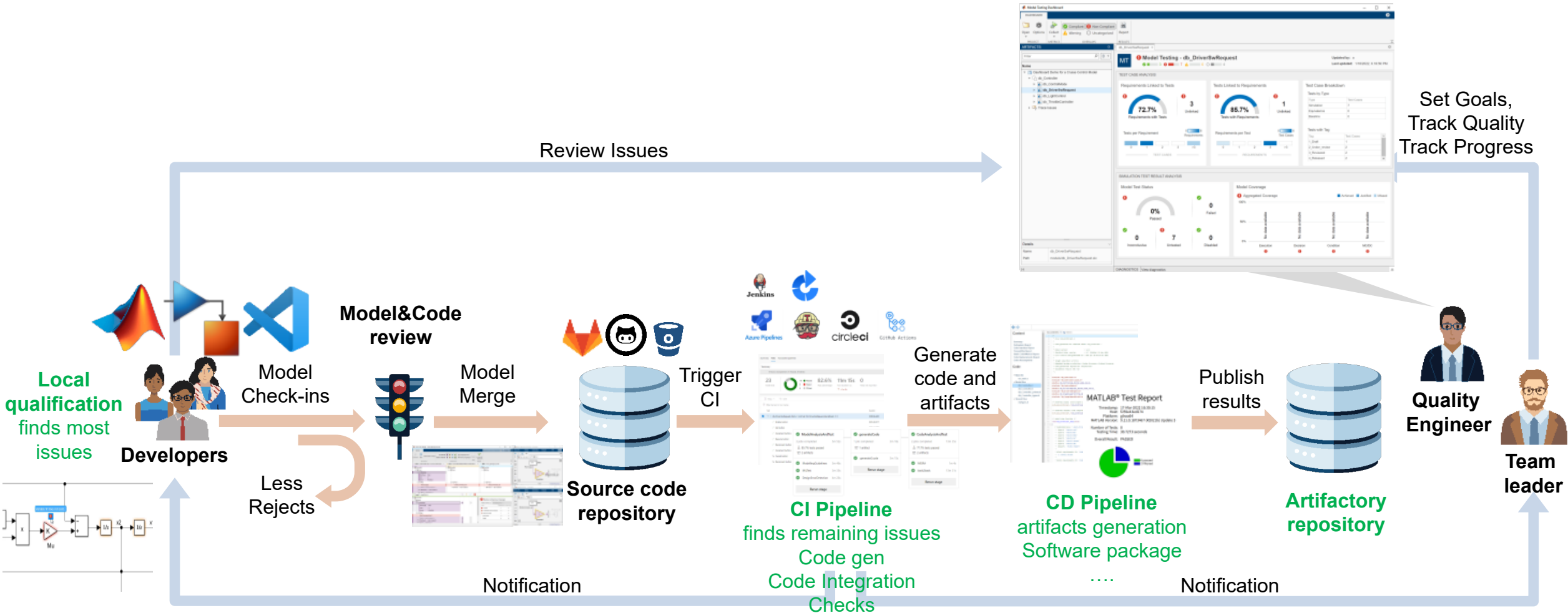
Figure 1. Issue counts for software releases before and after the adoption of Model-Based Design.

[Link to technical article](#)



# 基于模型设计的持续集成 workflow

# 将基于模型设计融入 CI 系统

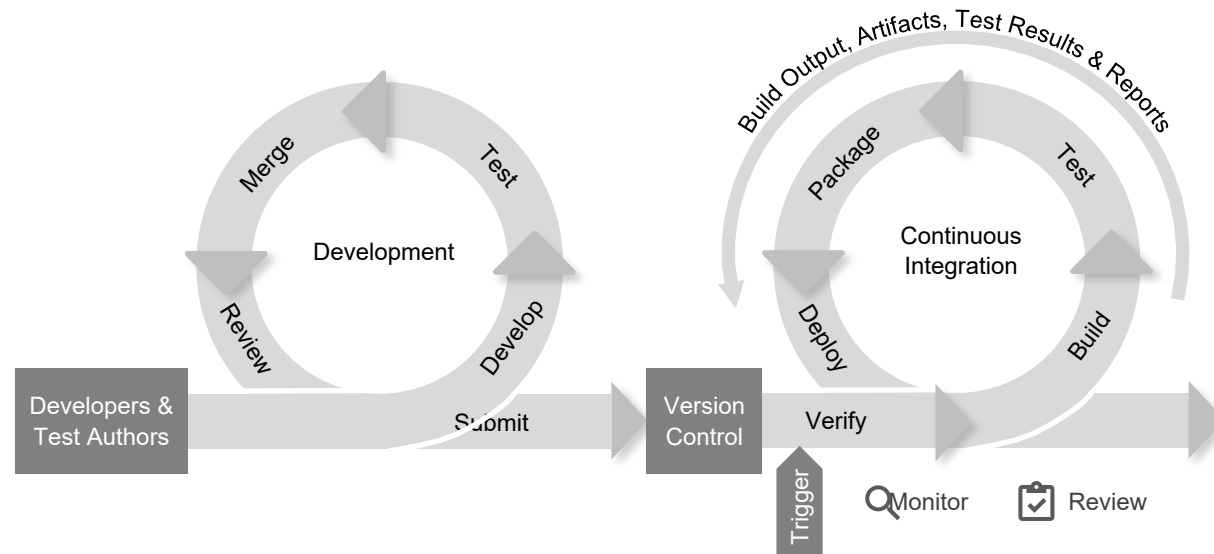


✓ **Direct Feedback**

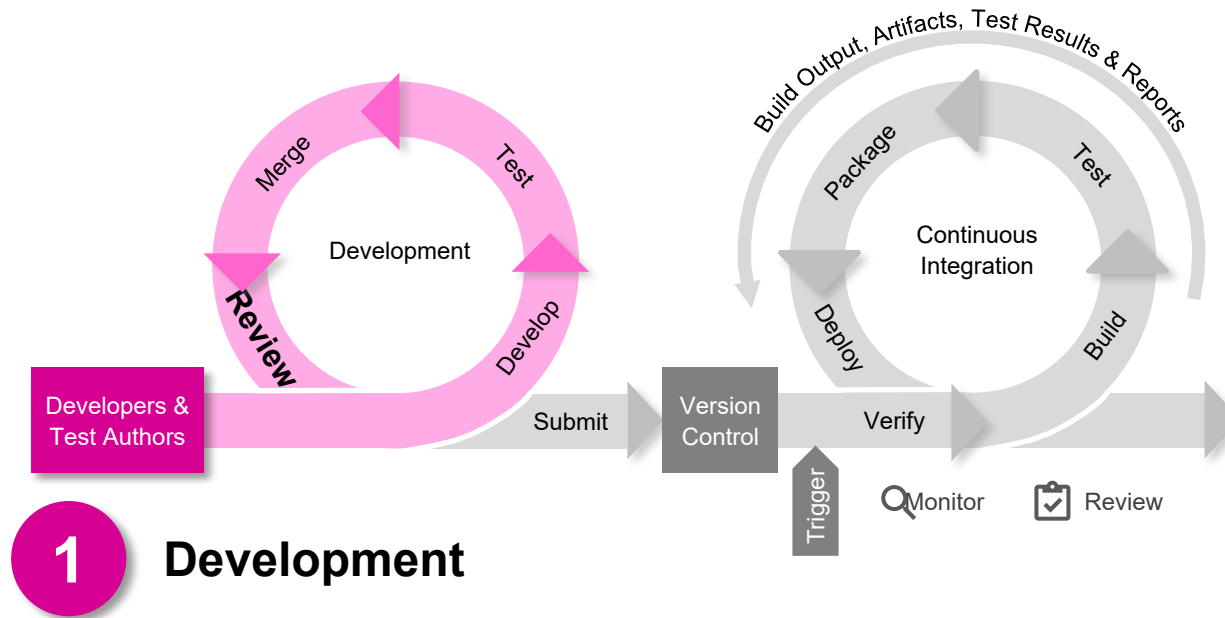
✓ **Flow / Automation**

✓ **Learning**

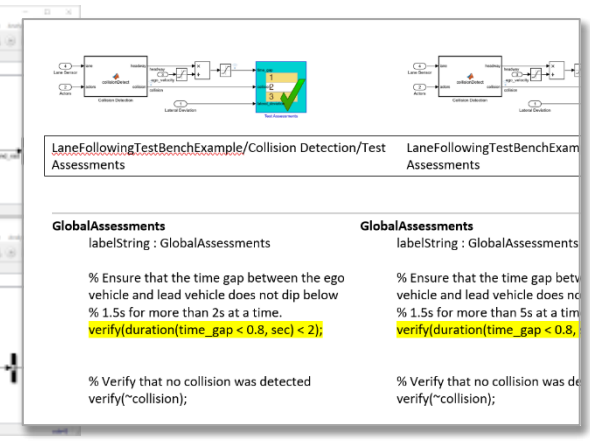
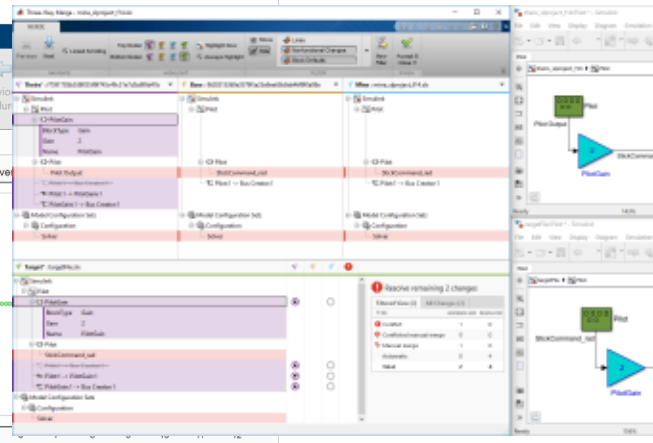
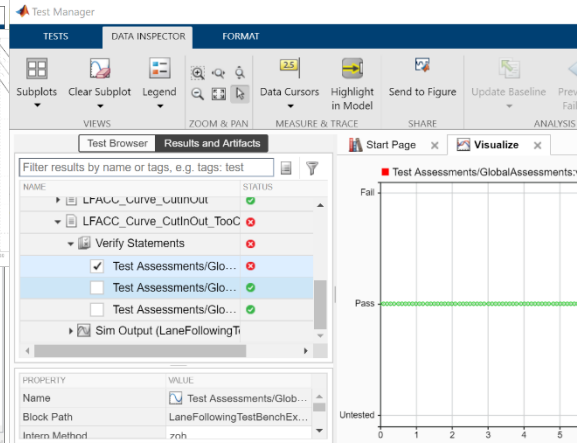
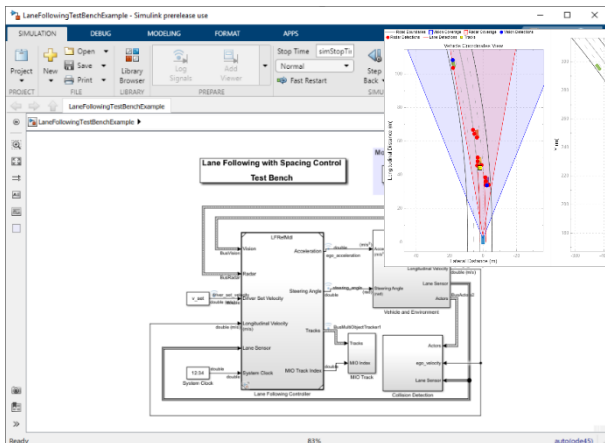
# 使用 MATLAB 和 Simulink 述说持续集成的故事



# 基于模型设计的持续集成工作流 开发阶段



## 1 Development



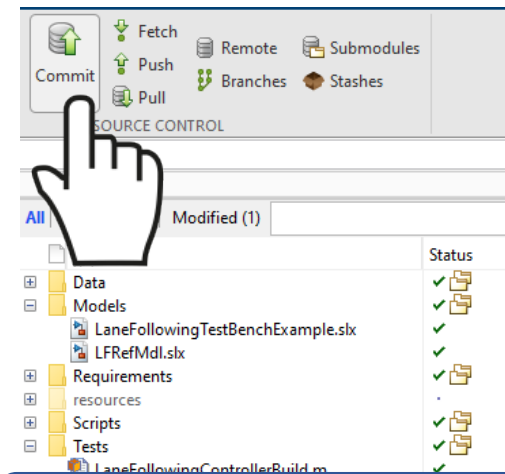
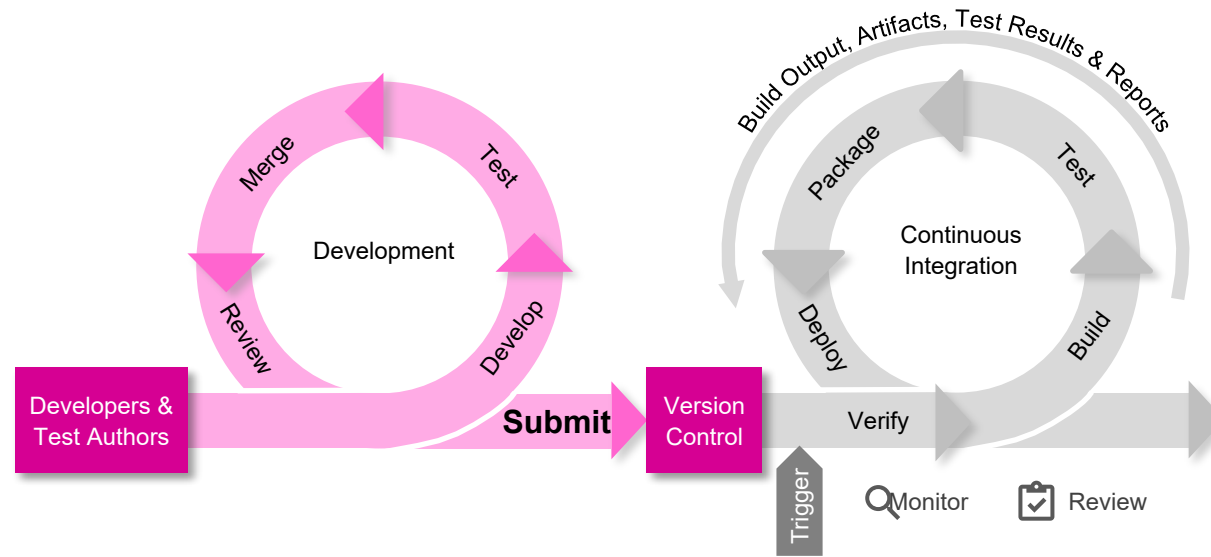
Modelling

Verification & Test

Compare & Merge

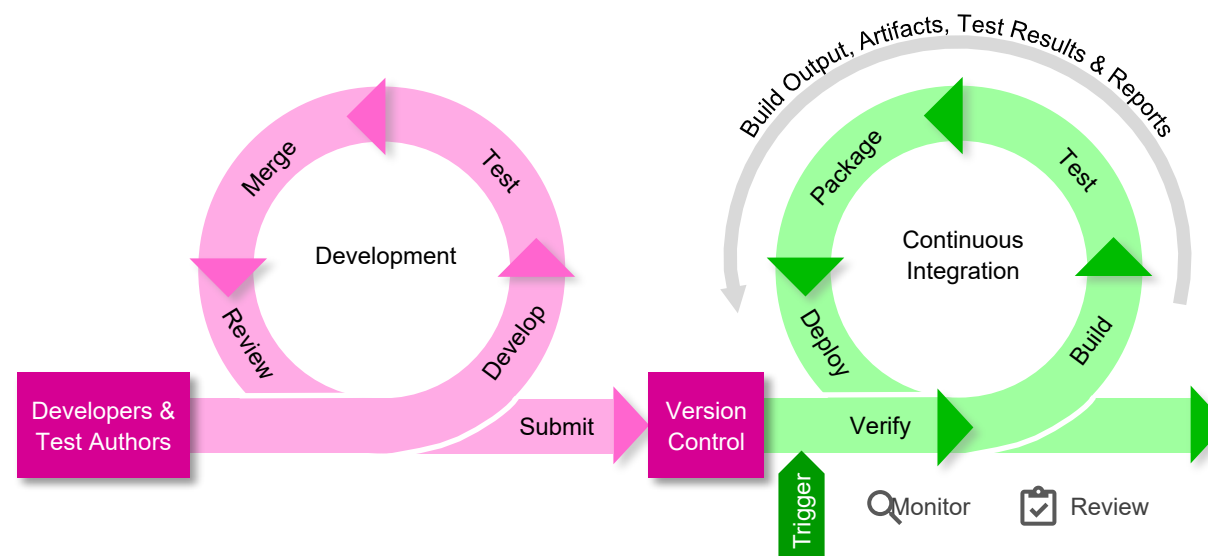
Gen Reports

# 基于模型设计的持续集成 workflow 推送



*Projects with SCM*

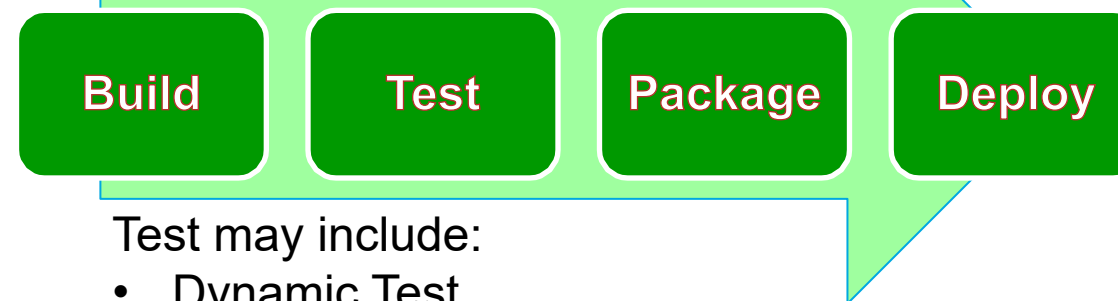
# 基于模型设计的持续集成 workflow



**1** Development

**2** Continuous Integration

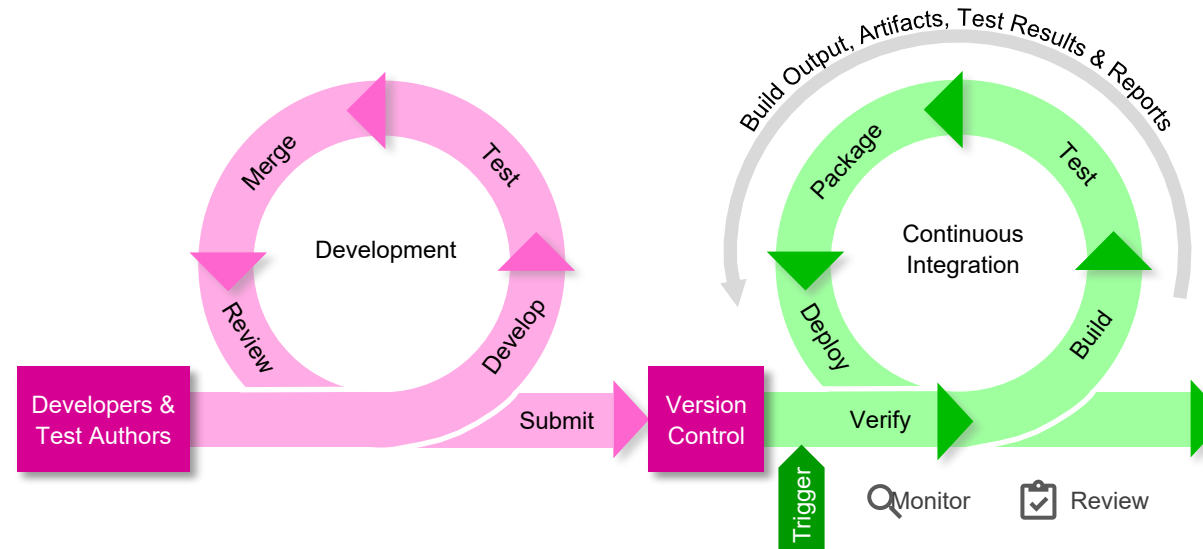
④ ④ Pipelines ④ ④



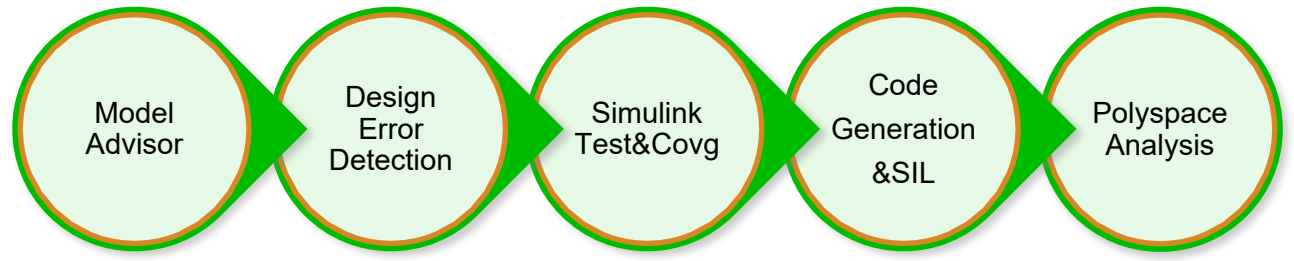
Test may include:

- Dynamic Test
- Static Checks
- Coverage...

# 基于模型设计的持续集成 workflow



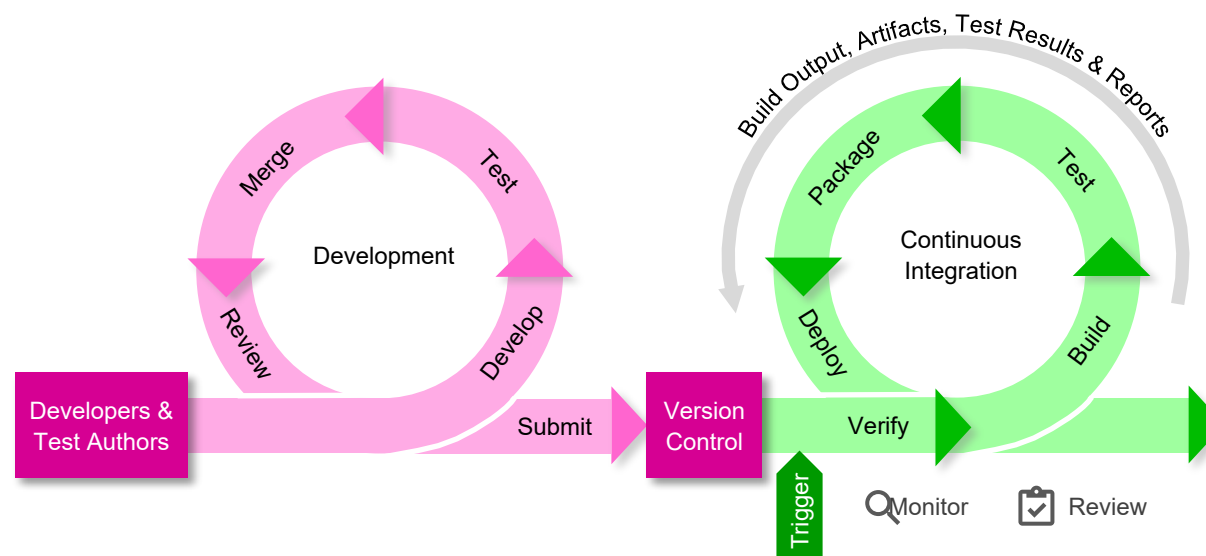
- 1** Development
- 2** Continuous Integration



Reports, Build Logs, Test Results, Code Coverage, Code Compliance, Runtime Errors

*Verification, Build and Test*

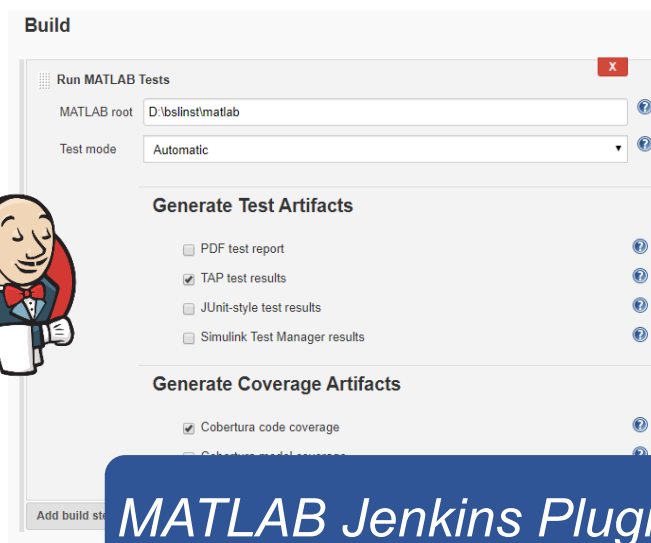
# 基于模型设计的持续集成 workflow



- 安装路径设置
- 过滤测试并运行
- 生成报告和其他产出物
- 提供pipeline接口

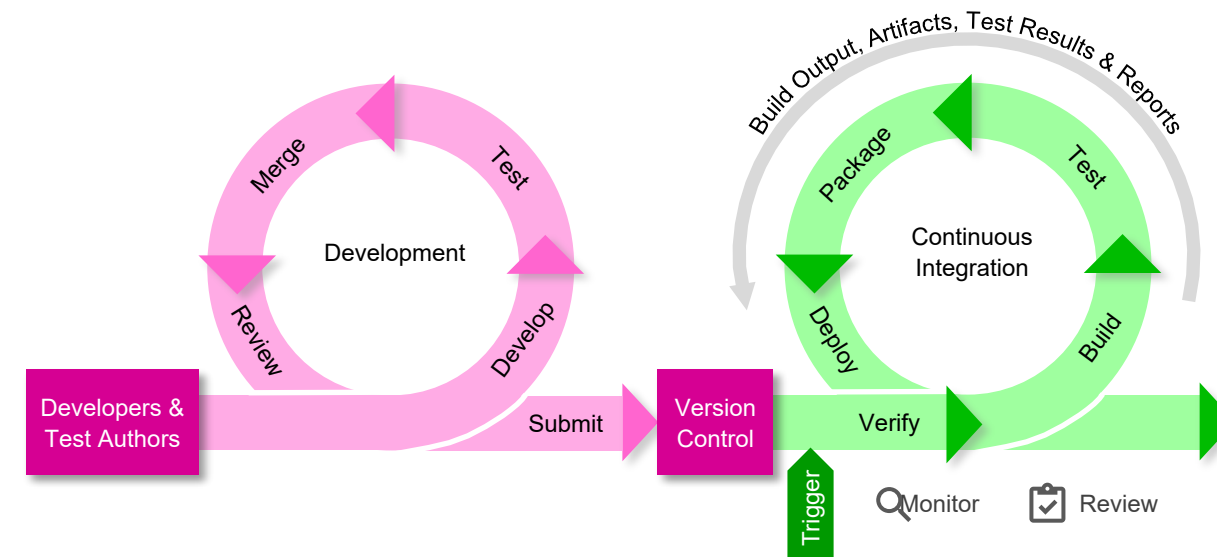
**1** Development

**2** Continuous Integration





# 基于模型设计的持续集成 workflow



**1** Development

**2** Continuous Integration

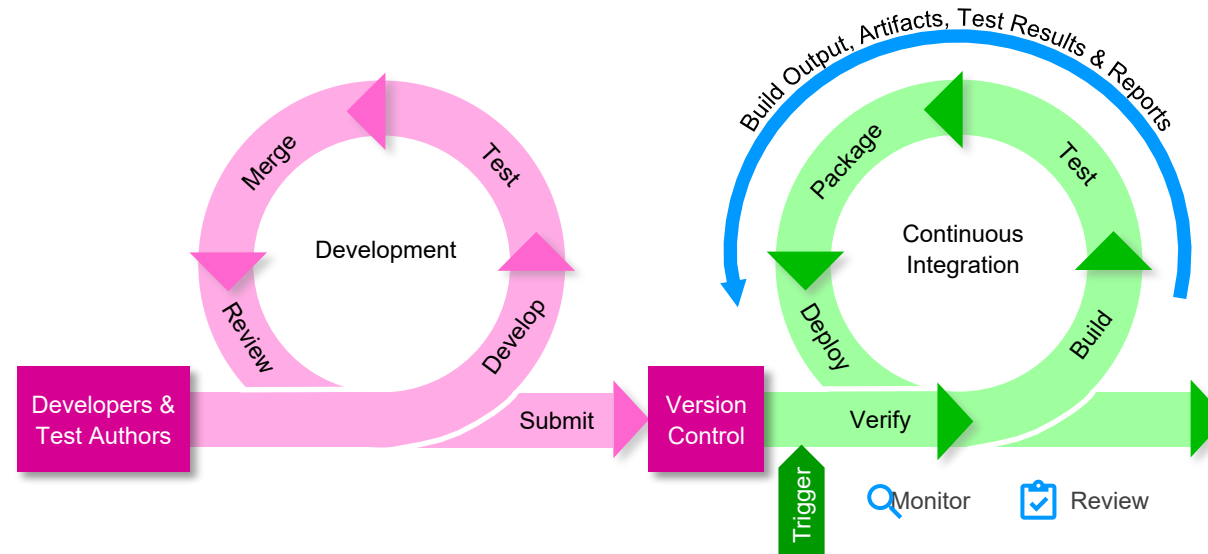


**matlab -batch**



*MATLAB -batch*

# 基于模型设计的持续集成 workflow



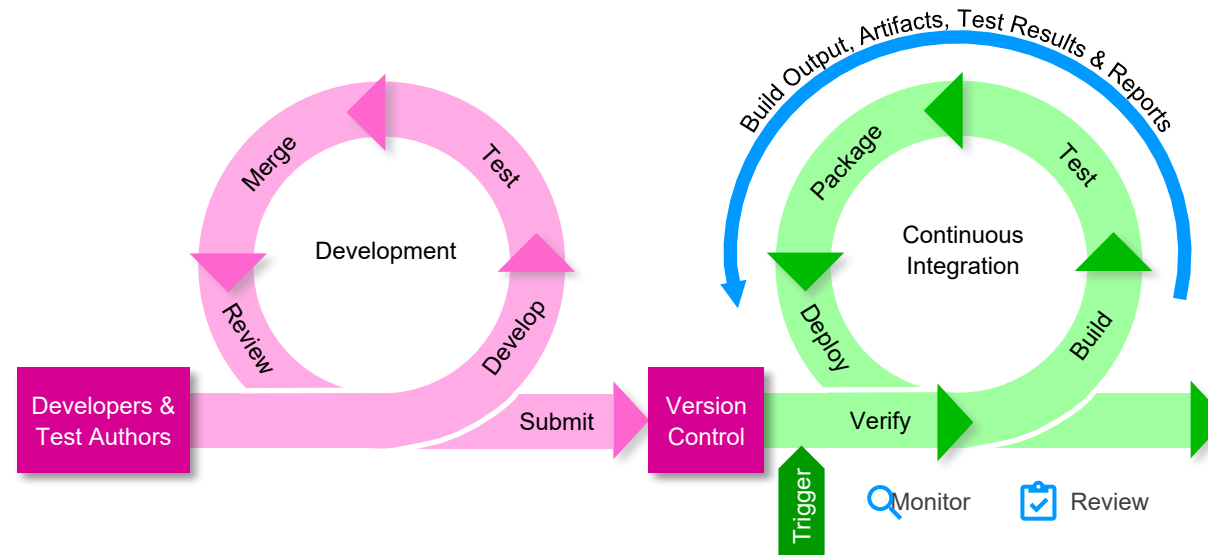
- 1 Development
- 2 Continuous Integration
- 3 Results Monitor and Review

## Failure Summary:

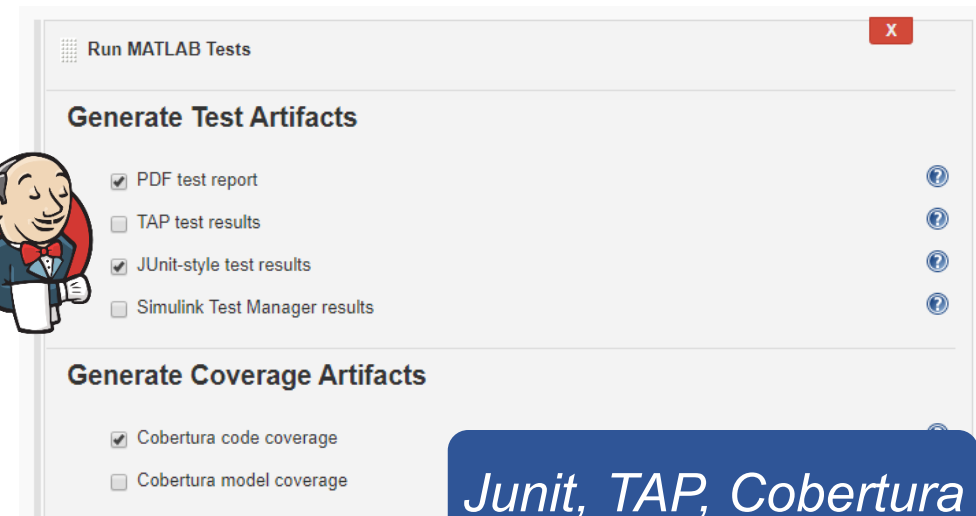
Name	Failed
----- LaneFollowingTestScenarios > Scenarios/LFACC_Curve_CutInOut_TooClose	X
ERROR: MATLAB error Exit Status: 0x00000001	
Build step 'Run MATLAB Tests' changed build result to FAILURE	
Finished: FAILURE	

*Text Output*

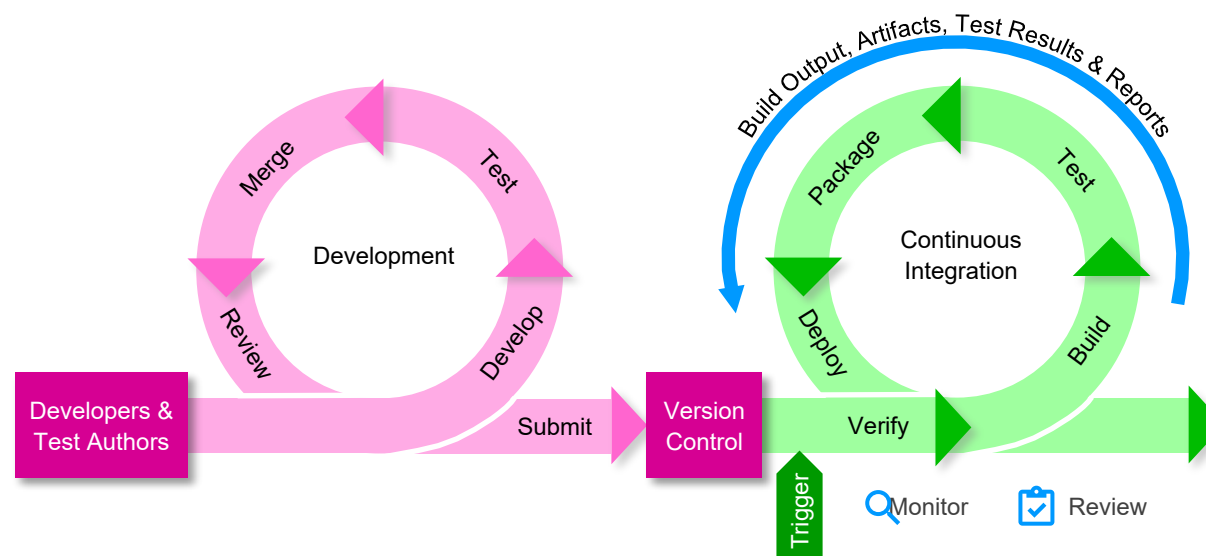
# 基于模型设计的持续集成 workflow



- 1 Development
- 2 Continuous Integration
- 3 Results Monitor and Review



# 基于模型设计的持续集成 workflow

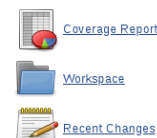


1 Development

2 Continuous Integration

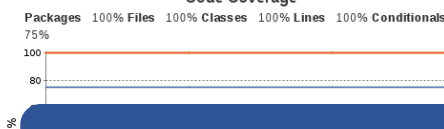
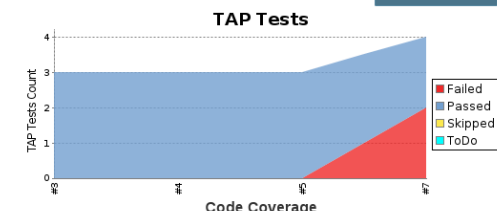
3 Results Monitor and Review

## Project UAV



### Permalinks

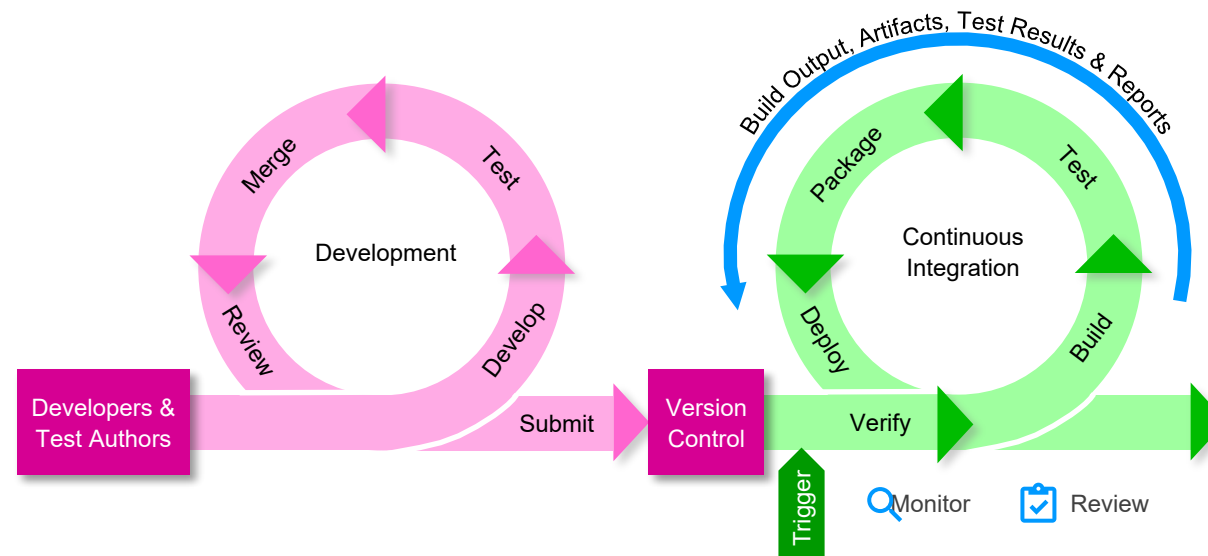
- [Last build \(#7\) . 3 min 26 sec ago](#)
- [Last stable build \(#5\) . 12 min ago](#)
- [Last successful build \(#7\) . 3 min 26 sec ago](#)
- [Last failed build \(#2\) . 1 hr 42 min ago](#)
- [Last unstable build \(#7\) . 3 min 26 sec ago](#)
- [Last unsuccessful build \(#7\) . 3 min 26 sec ago](#)
- [Last completed build \(#7\) . 3 min 26 sec ago](#)



Junit, TAP, Cobertura



# 基于模型设计的持续集成 workflow



- 1 Development
- 2 Continuous Integration
- 3 Results Monitor and Review

**Results: 2017-Jan-19 13:34:39**  
 Result Type: Result Set  
 Parent: None  
 Start Time: 2017-Jan-19 13:34:39  
 End Time: 2017-Jan-19 13:35:45  
 Outcome: Total: 2, Passed: 2

**Aggregated Coverage Results**

Analyzed Model	Sim Mode	Comp.	Decision	Condition	Execution	Relational Boundary
AHRS_voter	ModelRefSIL32	24%	100%	48%	80%	

[Back to Report Summary](#)

---

**AHRS\_voter\_SLDV\_TestCases**

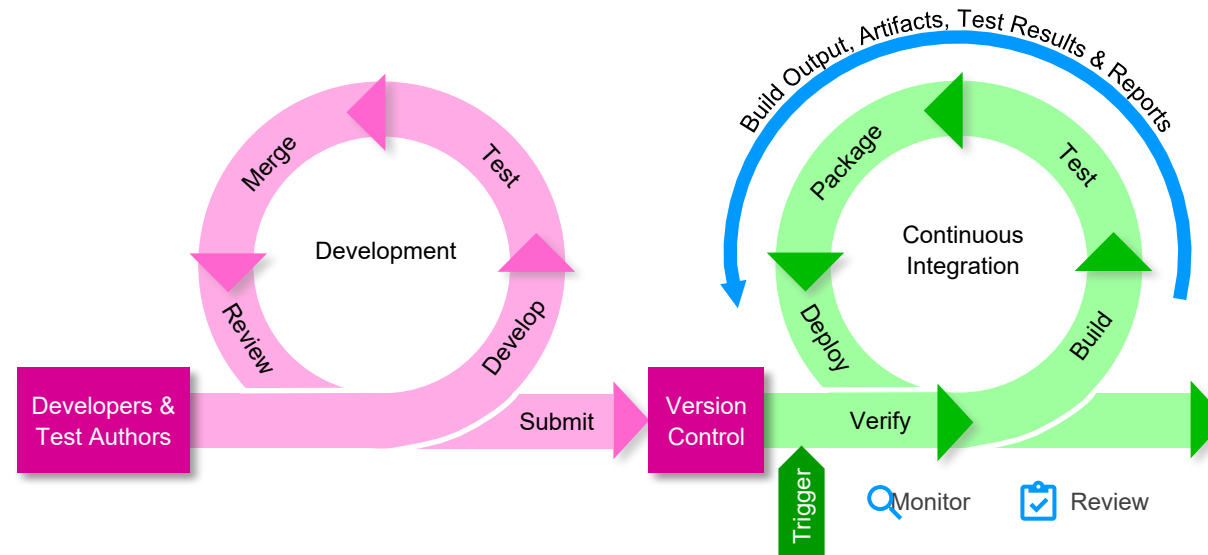
**Test Result Information**  
 Result Type: Test File Result  
 Parent: [Results: 2017-Jan-19 13:34:39](#)  
 Start Time: 2017-Jan-19 13:34:40  
 End Time: 2017-Jan-19 13:35:45  
 Outcome: Total: 2, Passed: 2

**Test Suite Information**  
 Name: AHRS\_voter\_SLDV\_TestCas

[Back to Report Summary](#)

*Publish Reports*

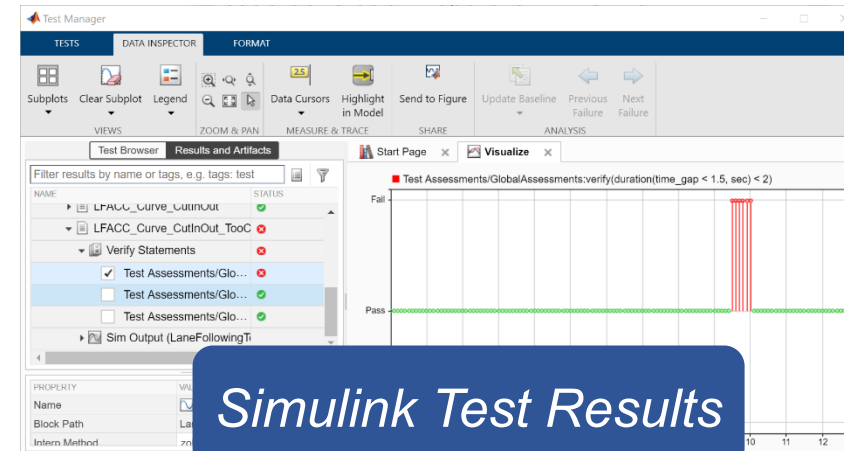
# 基于模型设计的持续集成 workflow



**1** Development

**2** Continuous Integration

**3** Results Monitor and Review



*Simulink Test Results*

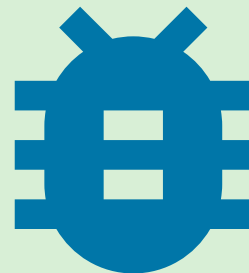
## 持续集成强化基于模型设计的流程



自动化和快速化



协同合作



提高质量



节约成本



审查&认证就绪

# 集成 Polyspace 到 CI 系统



# Polyspace 静态分析能力

## Bug Finder



→ High Quality, Secure, Compliant Code:

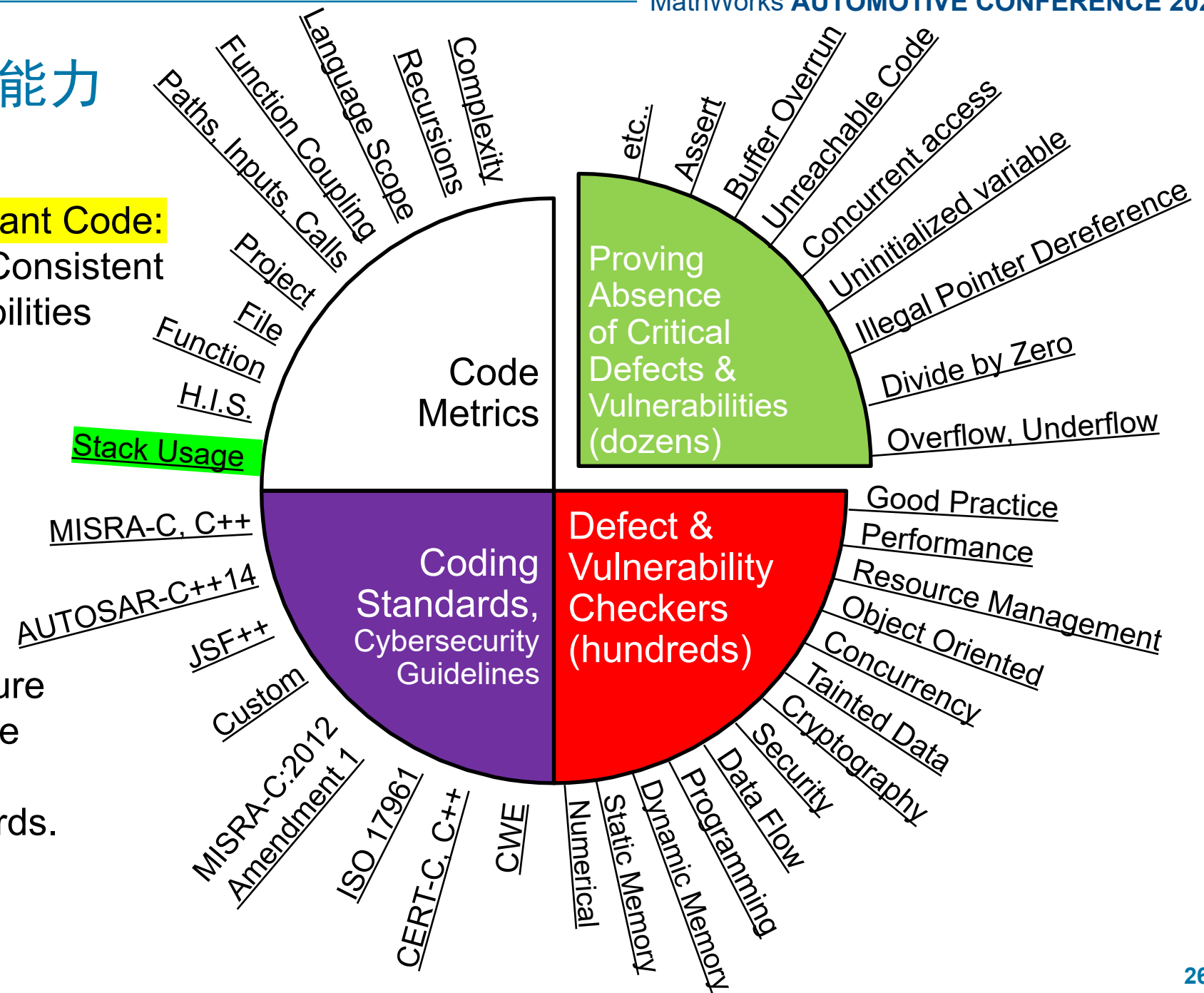
- Measurable, Maintainable, Consistent
- Very few defects or vulnerabilities
- Credits for functional safety, cybersecurity standards.

## Code Prover

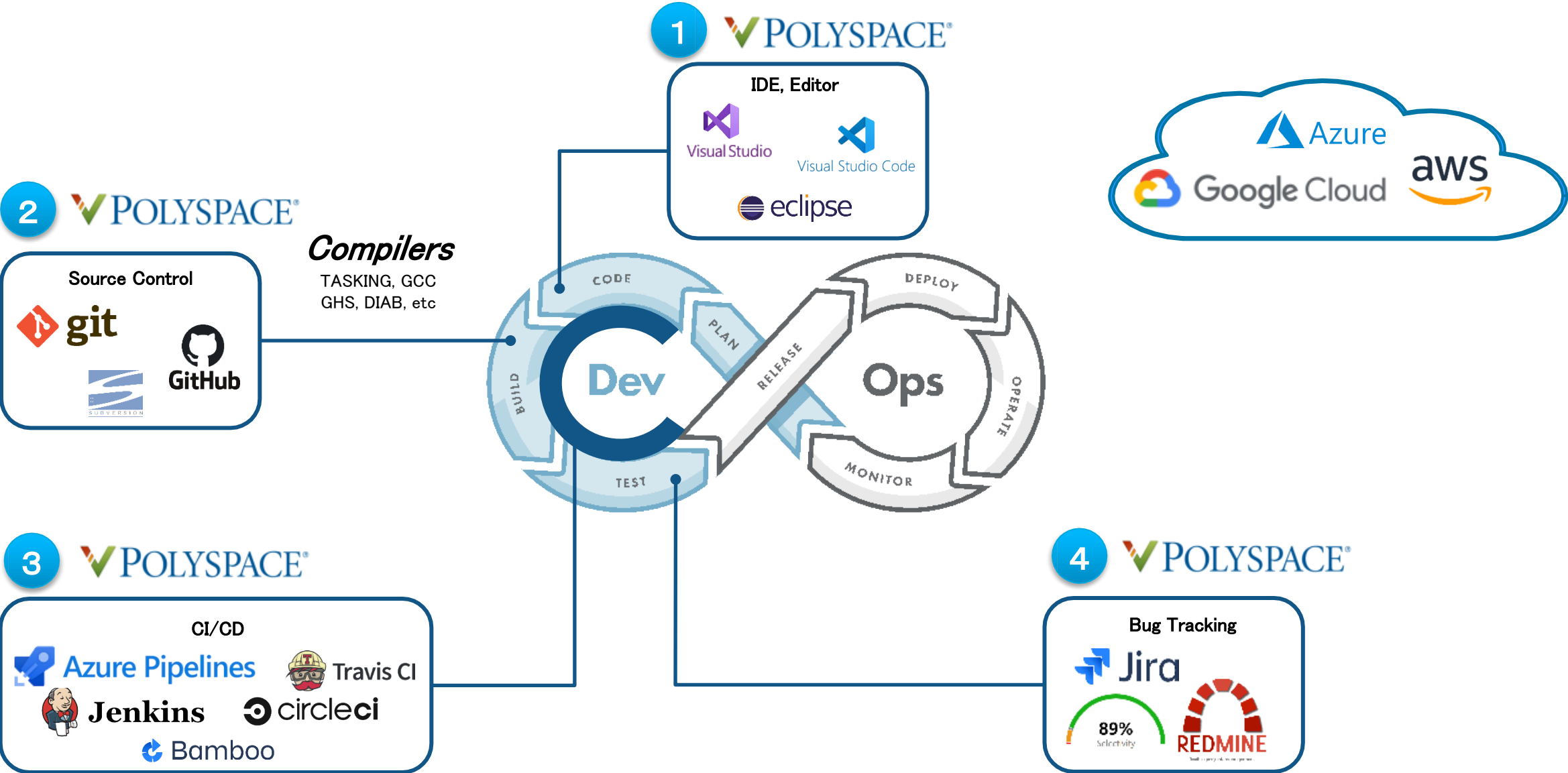


→ Fully Trusted Components:

- Reliable, Robust, Safe, Secure
- Proven free of critical runtime defects and vulnerabilities
- Additional credits for standards.



# 集成到 DevOps 中



# 集成到 DevOps 中

2 POLYSPACE

The screenshot shows the Polyspace IDE interface. On the left is the Project Explorer showing a project structure. The main window displays C++ code with annotations. A 'Results List' window is open at the bottom, showing a table of defects:

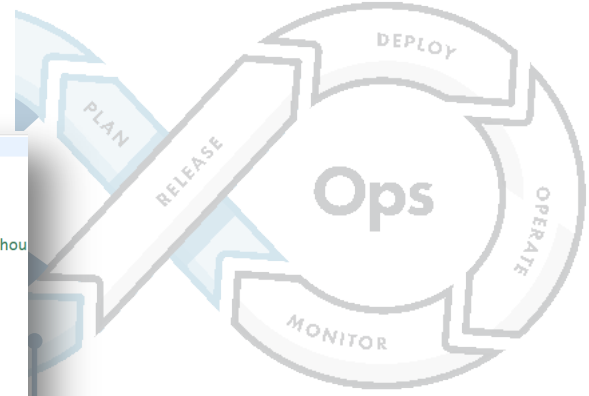
Family	ID	Type	Group	Missing
Defect	52	Performance	Missing	
AUTOSAR C++14	65	15 Exception handling	A15-4-4	
AUTOSAR C++14	67	15 Exception handling	A15-4-4	
AUTOSAR C++14	68	15 Exception handling	A15-4-4	
AUTOSAR C++14	69	15 Exception handling	A15-4-4	
AUTOSAR C++14	71	15 Exception handling	A15-4-4	
AUTOSAR C++14	72	15 Exception handling	A15-4-4	
AUTOSAR C++14	74	15 Exception handling	A15-4-4	
AUTOSAR C++14	11	2 Lexical conventions	A2-7-3	
AUTOSAR C++14	47	2 Lexical conventions	A2-7-3	

1 POLYSPACE

IDE, Editor

Visual Studio  
Visual Studio Code  
eclipse

Azure  
Google Cloud  
aws



4 POLYSPACE

Bug Tracking

Jira  
89% Selectivity  
REDMINE

# 集成到 DevOps 中

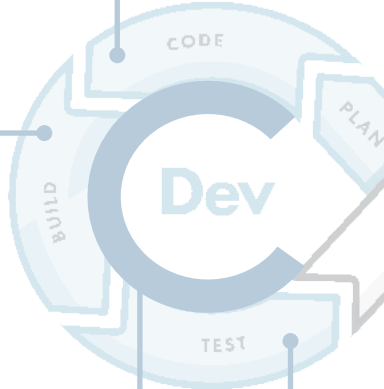
```
polyspace-configure -allow-build-error -allow-override -module -output-options-path options_path make -B
include/ src/idl_options.proto
-DBTHREAD_USE_FAST_PTHREAD_MUTEX -D__const__ = -D_GNU_SOURCE -DUSE_SYMBOLIZE -DNO_TCMALLOC -D_STDC_FORMAT_MACROS -D_
35\|2022-04-27T10:25:52+08:00\" -O2 -pipe -Wall -W -fPIC -fstrict-aliasing -Wno-invalid-offsetof -Wno-unused-parameter
r.cpp -o src/mcpack2pb/generator.o
```

```
y/esp_authenticator.o src/brpc/policy/file_naming_service.o src/brpc/policy/hasher.o src/brpc/policy/consul_naming_service.o src/
c/brpc/policy/redis_protocol.o src/brpc/policy/ubrpc2pb_protocol.o src/brpc/policy/auto_concurrency_limiter.o -Xlinker "-" -lgfla
t
> Copying to output/include
> Copying to output/lib
> Copying to output/bin
polyspace-configure: 416s: WARNING: Compilation units detected during your build in module libbrpc_a use different languages.
polyspace-configure: 416s: WARNING: Compilation units detected during your build in module libbrpc_so use different languages.
/home/.../workspace/brpc/incubator/brpc
```

## 2 POLYSPACE

Compilers  
TASKING, GCC  
GHS, DIAB, etc

Source Control



✓ Create project using build information

**Create project using build information**

Project has been successfully generated, please click on finish for opening the project in Polys...

Build command

Specify command used for building your source files  
C:\ti\ccs901\ccs\eclipse\ccstudio.exe

Specify working directory for running build command  
C:\Users\lehuahu\AppData\Local\Temp\Polyspace\build\FromBuildCmdDemo

Add advanced configure options

Run Stop

## 3 POLYSPACE

CI/CD

Azure Pipelines Travis CI  
Jenkins circleci  
Bamboo

Command output

```
2020-05-23 14:43:47.644:INFO:oejs.Server:Worker-13: Workbench early startup: Started @2
CCS HTTP adapter started! [ccs.port:9893] - Open 'localhost:9893/ide' in web browser to see
polyspace-configure: 117s: WARNING: Keeping potentially big build trace, remember to delete
polyspace-configure: 117s: WARNING: Keeping potentially big cache directory, remember to de
polyspace-configure: 0s: WARNING: Build command ignored (build deactivated)
polyspace-configure: 13s: INFO: Created project file C:\polyspace_workspace\FromBuildCmdD
polyspace-configure: 13s: WARNING: Keeping potentially big build trace, remember to delete it
```

# 集成到 DevOps 中

## 2 POLYSPACE

### Compilers

TASKING, GCC  
GHS, DIAB, etc

Source Control

## 3 POLYSPACE

### CI/CD

### 流水线

定义

Pipeline script

脚本 ?

```

59 {
60   sh label: "Cleanup", script: "make clean"
61   sh label: "Polyspace configure", script: "$configure -allow-build-error -allow-overwrite -lang
62 }
63 stage ("Analyze")
64 {
65   sh label: "Polyspace analysis",
66   script: "$analyze -options-file ${PROG}.opts -misra3 all -code-metrics -results-dir $RESULT"
67 }
68
69 // remove this paragraph if you do not use Polyspace Access

```

### 阶段视图

	Prepare	Checkout	Build	Polyspace Analysis	Upload to Access	Quality Gate	Email Notification
Average stage times: (Average full run time: ~40s)	374ms	2s	9s	12s	4s	8s	875ms
#41 Sep 13 15:35 No Changes	323ms	2s	10s	13s	4s	8s	1s
#40 Sep 10 23:08 No Changes	303ms	1s	9s	11s	5s	8s	1s
#39 Sep 10 22:52 No Changes	471ms	3s	10s	15s	4s	8s	1s

# 集成到 DevOps 中

### Remaining Findings

Remaining	98
New Remaining	7
Assigned To Me Remaining	0
Unassigned Remaining	98

### Code Metrics

Sub-project(s)	0
File(s)	6
Line(s)	429
Cyclomatic complexity	6

### Quality Objectives

95.0%

Threshold: SQ04

Remaining: 24

### Run-time Check

Remaining: 30

Red	5
Orange	20
Gray	6
Green	219

### Coding Rules

Remaining: 60

To Do	60
Done	4

### Result Details

Status: Unreviewed

Severity: Unset

Assigned to: Type username or...

Track issue Create ticket

### Create Issue

Summary\*: Possibly unintended evaluation of expression because of

Type\*: Bug

Component\*: Personal

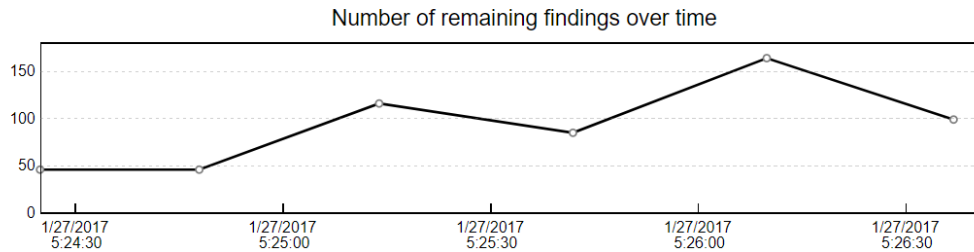
SubComponent: Unknown

Problem: Found in /mathworks/GNB/devel/sandbox/cpreve/Metrics-results/Bug\_Finder\_Example-Trends/sources/programming.c

- Go to Polyspace finding here: <https://ah-access-ipolyspace:9443/metrics/index.html?>



Trends



### CI/CD

Azure Pipelines

Travis CI

Jenkins

circleci

Bamboo

4

## POLYSPACE®

### Bug Tracking

Jira

REDMINE

89%  
Selectivity

jenkins / ScriptedPipeline\_Example x +

demo-polyspace-ci.gnb.mathworks.com:8080/blue/organizations/jenkins/ScriptedPipeline\_Example/detail/ScriptedPipeline\_Example/52/pipeli...

ScriptedPipeline Example < 52

Pipeline Changes Tests Artifacts

Branch: — <1s No changes

Commit: — - Started by user anonymous

```

graph LR
  Start((Start)) --> Checkout((Checkout))
  Checkout --> Configure((Configure))
  Configure --> Analyze((Analyze))
  Analyze --> PublishResults((Publish Results))
  PublishResults --> AssignFindings((Assign new findings))
  AssignFindings --> QualityGate((Quality Gate))
  QualityGate --> MakePDFReport((Make PDF Report))
  MakePDFReport --> Notification((Notification))
  Notification --> End((End))
  
```

Queued Waiting for run to start

Started "ScriptedPipeline\_Example" #52 [OPEN](#) ×

demo-polyspace-ci.gnb.mathworks.com:8080/blue/organizations/jenkins/.../pipeline

云端部署持续集成，用于大规模系统



# Azure Pipeline 的持续集成 workflow 使用参考架构进行快速设置

Dockerfile

Contents History Compare Blame

Press F11 to exit full screen

```
1 # Copyright 2019 - 2021 The MathWorks, Inc.
2
3 # To specify which MATLAB release to install in the container, edit the value of the MATLAB_RELEASE argument.
4 # Use lower case to specify the release, for example: ARG MATLAB_RELEASE=r2020a
5 #ARG MATLAB_RELEASE=r2021b
6 ARG MATLAB_RELEASE
7
8 # When you start the build stage, this Dockerfile by default uses the Ubuntu-based matlab-deps image.
9 # To check the available matlab-deps images, see: https://hub.docker.com/r/mathworks/matlab-deps
10 FROM mathworks/matlab-deps:${MATLAB_RELEASE}
11
12 # Declare the global argument to use at the current build stage
13 ARG MATLAB_RELEASE
14
15 # Install rpm dependencies
16 RUN export DEBIAN_FRONTEND=noninteractive && apt-get update && \
17     apt-get install --no-install-recommends --yes \
18         wget \
19         unzip \
20         ca-certificates && \
21     apt-get clean && apt-get autoremove
22
23 # Run rpm to install MATLAB in the target location and delete the rpm installation afterwards
24 RUN wget -q https://www.mathworks.com/mpm/glnxa64/mpm && \
25     chmod +x mpm && \
26     ./mpm install \
27         --release=${MATLAB_RELEASE} \
28         --destination=/opt/matlab \
29         --products MATLAB Simulink Stateflow Simulink_Requirements MATLAB_Coder Simulink_Coder Embedded_Coder Simulink_Check Simulink_Test Simulink_Coverage Simulink_Design_Verifier && \
30     rm -f mpm /tmp/mathworks_root.log && \
31     ln -s /opt/matlab/bin/matlab /usr/local/bin/matlab
32
33 # Add "matlab" user and grant sudo permission.
34 RUN adduser --shell /bin/bash --disabled-password --gecos "" matlab && \
35     echo "matlab ALL=(ALL) NOPASSWD: ALL" > /etc/sudoers.d/matlab && \
36     chmod 0440 /etc/sudoers.d/matlab
37
38 # One of the following 2 ways of configuring the license server to use must be
39 # uncommented.
40
41 ARG LICENSE_SERVER
42 # Specify the host and port of the machine that serves the network licenses
43 # if you want to bind in the license info as an environment variable. This
44 # is the preferred option for licensing. It is either possible to build with
45 # something like --build-arg LICENSE_SERVER=27000@MyServerName, alternatively
46 # you could specify the license server directly using
47 ENV MLM_LICENSE_FILE=27000@netIn-server.internal.cloudapp.net
48 #ENV MLM_LICENSE_FILE=$LICENSE_SERVER
49
50 # Alternatively you can put a license file into the container.
51 # You should fill this file out with the details of the license
52 # server you want to use and uncomment the following line.
53 # COPY network.lic /opt/matlab/licenses/
54
55 # Set user and work directory
56 USER matlab
57 WORKDIR /home/matlab
58 ENTRYPOINT ["matlab"]
59 CMD [""]
```

Dependencies

# Azure Pipeline 的持续集成 workflow 使用参考架构进行快速设置

IT / MATLAB Admins



- Dockerfile
- YAML

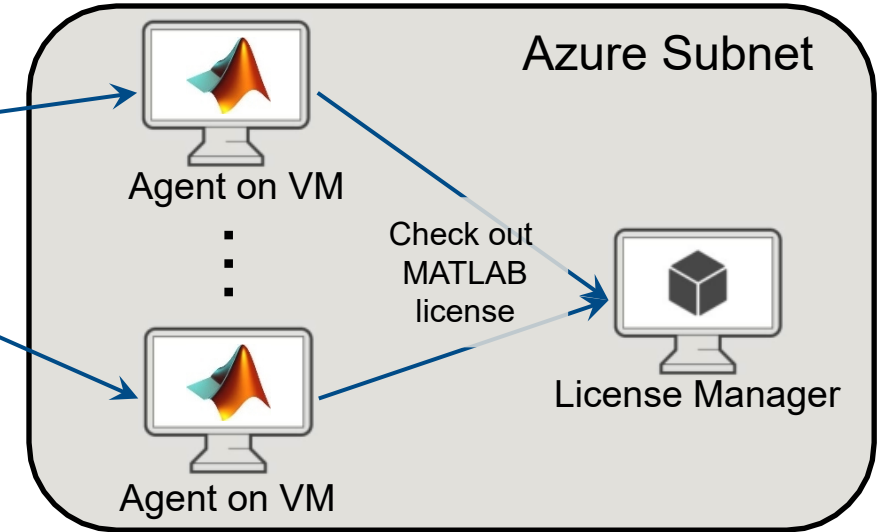


Push  
MATLAB  
image



Container  
Registry

Instantiate  
MATLAB  
container



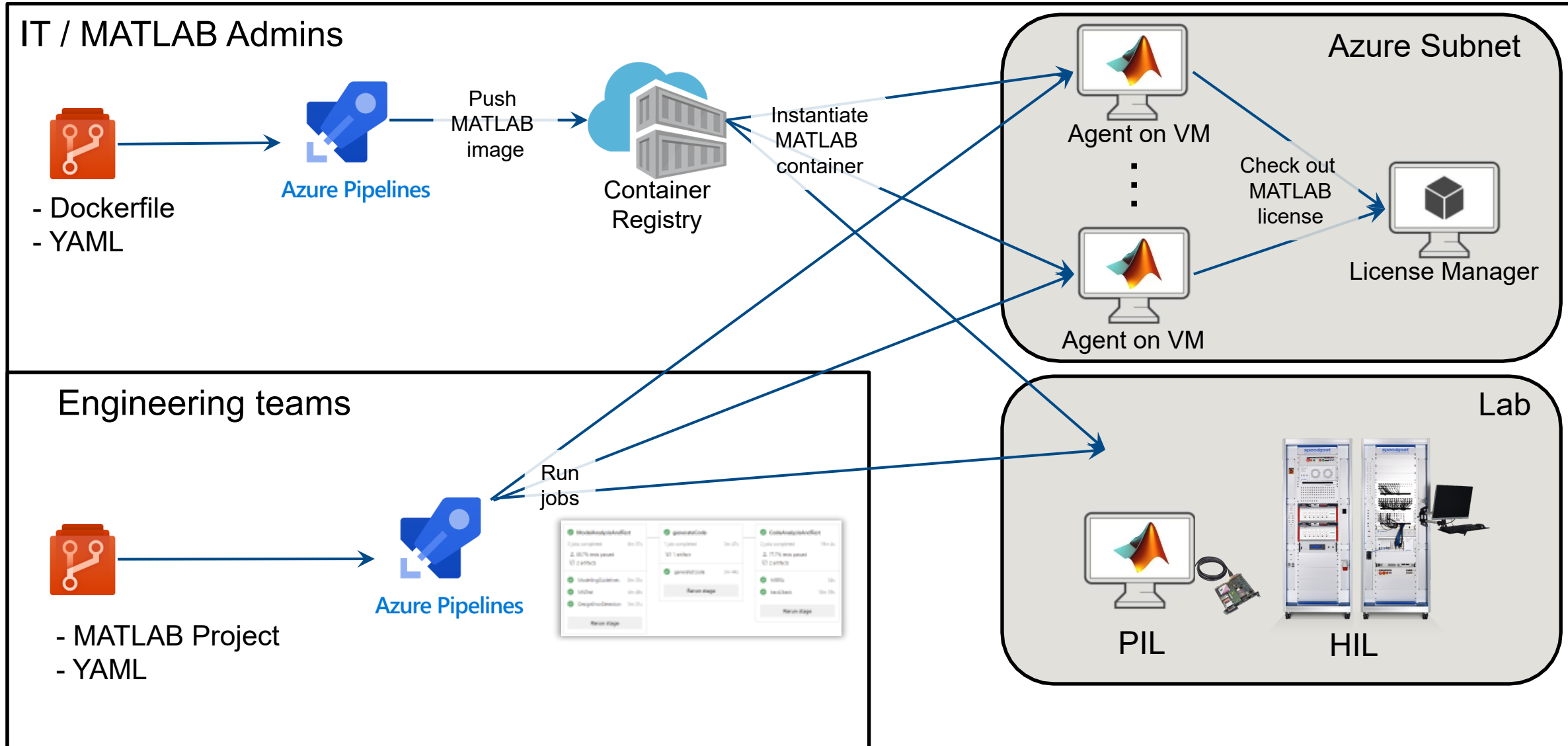
# Pipeline 的持续集成 workflow 推送变更到云端触发MBD的云端构建

The screenshot displays the MATLAB R2021a environment with a project named 'CruiseDashboardExample.prj'. The interface is divided into several panes:

- Current Folder:** Shows the file structure of the project, including folders like 'work', 'scripts', 'requirements', 'models', 'generated\_reports', 'data', 'CI', and files like 'Jenkinsfile', 'README.md', 'license.txt', 'azure-pipelines-docker.yml', 'azure-pipelines.yml', and '.gitlab-ci.yml'.
- Project - Demo Project including requirements based unit tests:** A table listing files and their status and classification.
- Workspace:** Shows the current workspace contents, including a project named 'prj'.
- Command Window:** Displays the MATLAB prompt 'fx >>'.

Name	Status	Classification	Git	Review Status	Owner
github	✓		•		
CI	✓		•		
data	✓		•		
generated_reports	✓		•		
models	✓		•		
db_Controller	✓		•		
db_ControlMode	✓		•		
harnesses	✓		•		
unit_tests	✓		•		
db_ControlMod...	✓	Derived	•		
db_ControlMod...	✓	Design	•		
db_ControlMod...	✓	Derived	•		
db_DriverSwRequ...	✓		•		
db_TargetSpeedT...	✓		•		
requirements	✓		•		
scripts	✓		•		
gitattributes	✓		•		
.gitignore	✓		•		
.gitlab-ci.yml	✓		•		
azure-pipelines.yml	✓		•		

# 架构图示例



# 持续集成实践中的建议

# 建立成熟的问题检测和处置流程

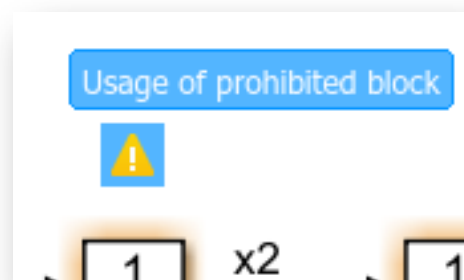
Maturity	Description of static analysis
Level 0	No analysis performed
Level 1	Analysis once per release
Level 2	Analysis multiple times per release, critical issues added to backlog
Level 3	Fully automated & frequent analysis, all critical issues resolved
Level 4	New issues break the build, critical and high issues resolved, IDE-plugins are used



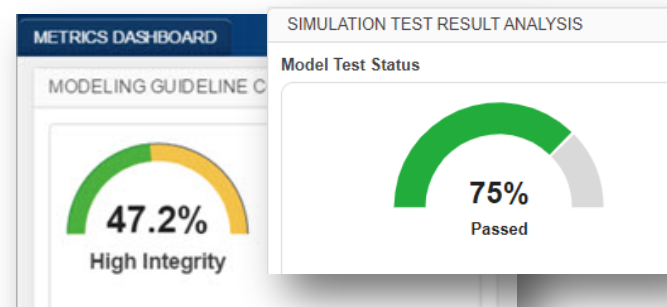
CI pipeline

Process

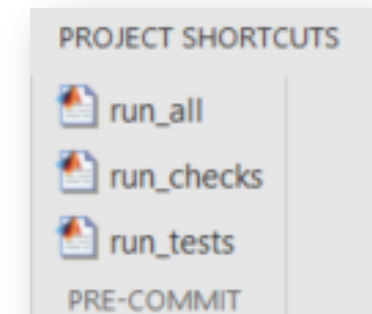
Pre-commit checks



Edit time checks



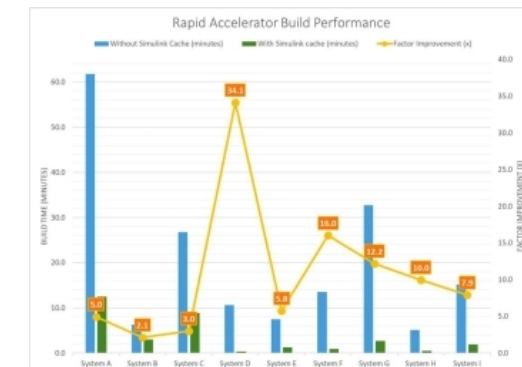
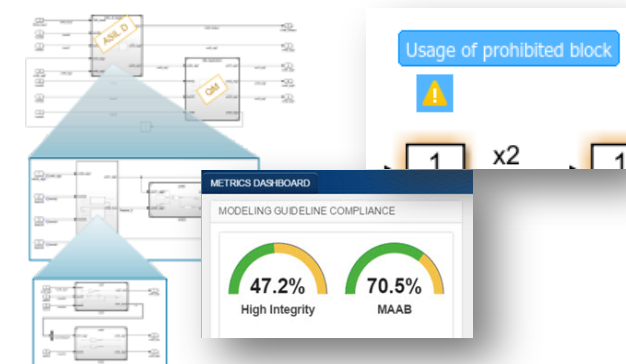
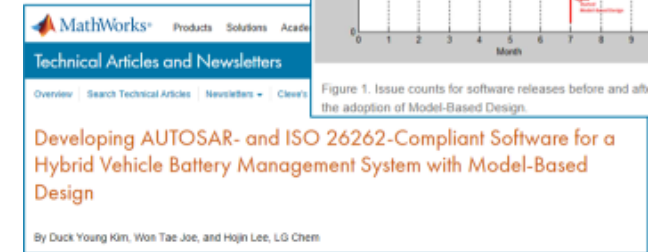
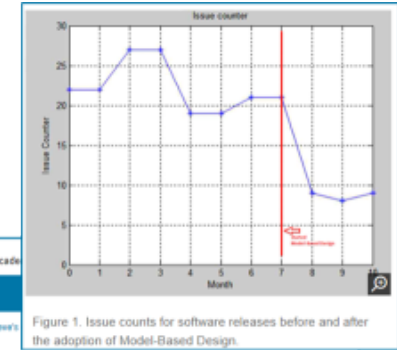
Quality Dashboards



MATLAB Project shortcuts

# 保持沟通联络 – 利用MathWorks的CI和DevOps专业知识

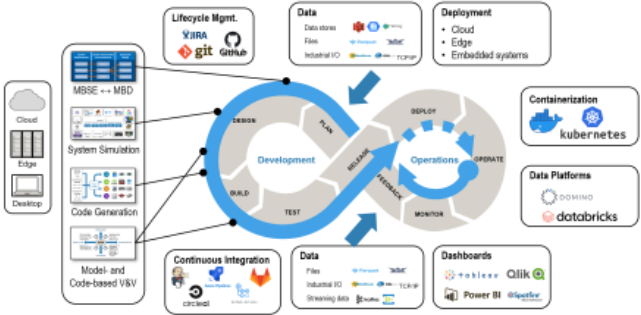
- 提高自动化程度和CI成熟度
  - 将MBD与CI系统、版本管理系统等其他系统集成
  - 尽早检查错误，建立成熟的问题检测和处置流程
- 改进系统架构，掌控复杂度
  - 为测试自动化和功能安全和网络安全进行模块化
- 解决性能瓶颈



# 要点回顾

- 将基于模型设计和Polyspace融入CI和DevOps应对软件开发转型的挑战
- 在云上部署CI，以提高MBD的灵活性
- 保持联络以强化CI流程和最佳实践

DORA metrics impact	Throughput		Stability	
	Lead Time For Change	Deployment Frequency	Change Failure Rate	Mean Time to Recovery
CI/CD pipelines	X	X	X	X
toolchain upgrades	X	X		X
system architecture		X	X	
cache files on CI	X	X		





感谢您的聆听！