MathWorks
AUTOMOTIVE CONFERENCE 2022
North America

MBD meets CI: I connected Simulink with my CI system, what’s next?

Dr. Tjorben Gross, MathWorks

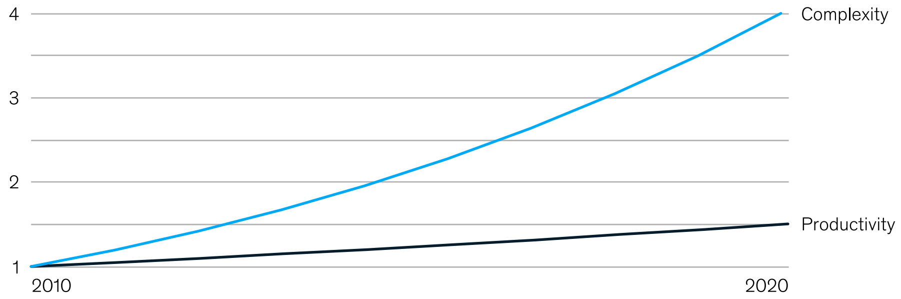
MathWorks®

The challenge: Mastering The Digital Transformation

- “Overall, the goal is to **accelerate development speed and enable early testing**.”
- *“Areas of concern include **agile practices, continuous integration, and automated testing**.“*
- *“The introduction of a standardized, **state-of-the-art development toolchain** is a key enabler to **unlock 30% to 40% of productivity** potentials from automated testing and agile methods.”*

Software complexity is increasing more quickly than productivity.

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



Source: McKinsey's SoftCoster embedded software project database

McKinsey
& Company

Source: https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/when-code-is-king-mastering-automotive-software-excellence

Model-Based Design enables high DevOps performance

- DevOps Goal: „**Reduce the time** between committing a change and placing it in production, **while ensuring high quality and compliance**„

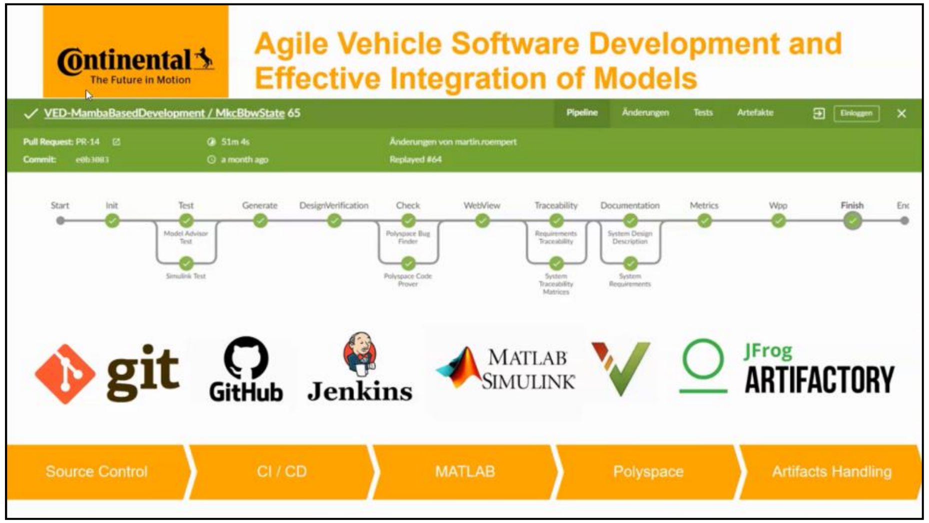
ISO 26262 – Functional Safety

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)



Continental The Future in Motion

Agile Vehicle Software Development and Effective Integration of Models

VED-MambaBasedDevelopment / MkcBbwState 65

Pipeline | Änderungen | Tests | Artefakte | Einloggen

Pull Request: PR-14

Commit: e0b3883

51m 4s

a month ago

Änderungen von martin.roempert

Replayed #64

Online Panel Discussion: Agile Vehicle Software Development and Effective Integration of Models

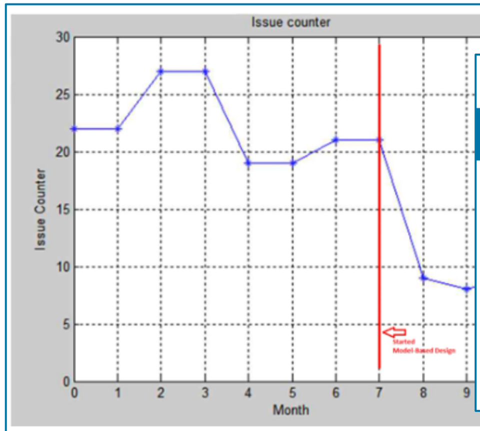


Figure 1. Issue counts for software releases before and after the adoption of Model-Based Design.

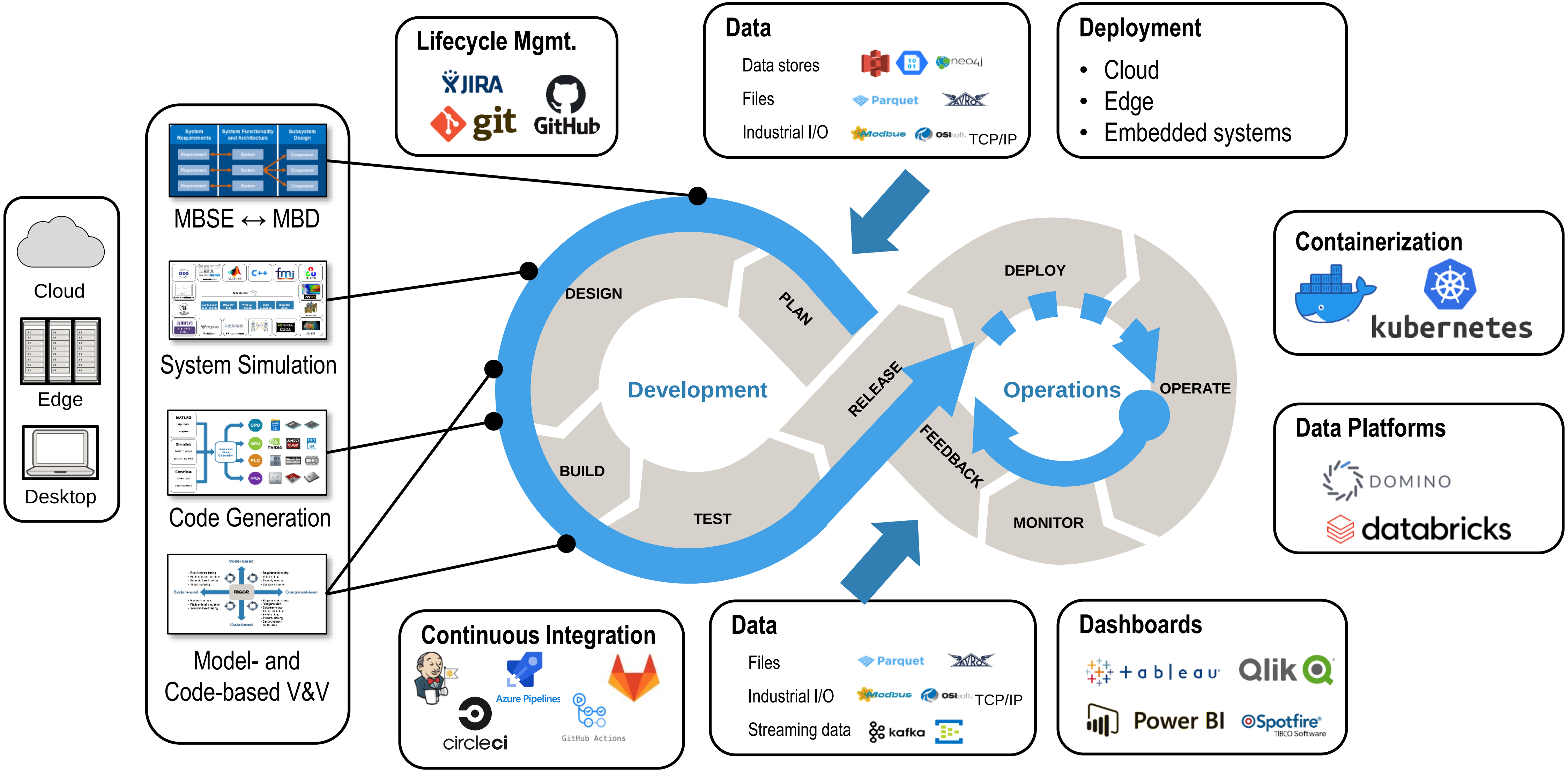
Technical Articles and Newsletters

Developing AUTOSAR- and ISO 26262-Compliant Software for a Hybrid Vehicle Battery Management System with Model-Based Design

By Duck Young Kim, Won Tae Joe, and Hojin Lee, LG Chem

Link to technical article

Example for DevOps building blocks for embedded production SW



Cloud

Edge

Desktop

MBSE ↔ MBD

System Simulation

Code Generation

Model- and Code-based V&V

Lifecycle Mgmt.

JIRA git GitHub

Data

Data stores neo4j

Files Parquet AVRO

Industrial I/O Modbus OSIsoft TCP/IP

Deployment

- Cloud
- Edge
- Embedded systems

Containerization

kubernetes

Data Platforms

DOMINO

databricks

Continuous Integration

Azure Pipelines circleci GitHub Actions

Data

Files Parquet AVRO

Industrial I/O Modbus OSIsoft TCP/IP

Streaming data kafka

Dashboards

tableau Qlik

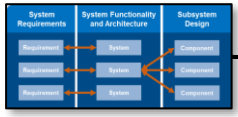
Power BI Spotfire TIBCO Software

Focus Today: Continuous Integration for embedded production SW

Cloud

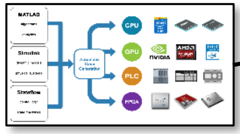
Edge

Desktop

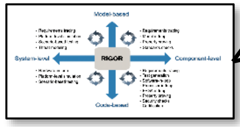


MBSE ↔ MBD

System Simulation



Code Generation



Model- and Code-based V&V

Lifecycle Mgmt.

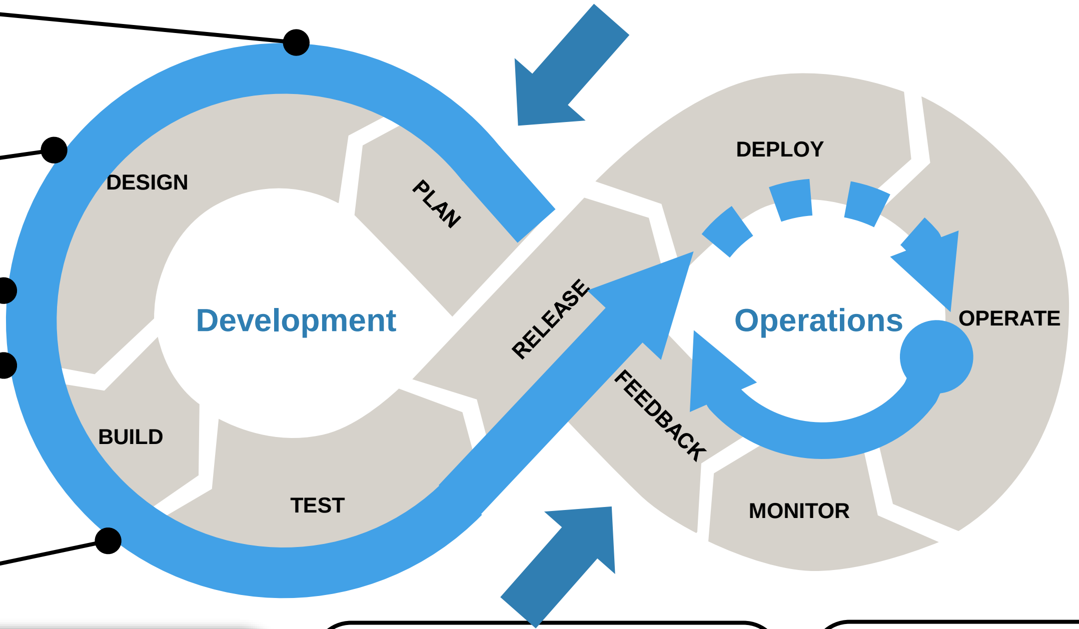
JIRA git GitHub

Data

Data stores	neo4j
Files	Parquet AVRO
Industrial I/O	Modbus OSIsoft TCP/IP

Deployment

- Cloud
- Edge
- Embedded systems



Containerization

kubernetes

Data Platforms

DOMINO

databricks

Continuous Integration

Azure Pipelines

circleci

GitHub Actions

Data

Files	Parquet AVRO
Industrial I/O	Modbus OSIsoft TCP/IP
Streaming data	kafka

Dashboards

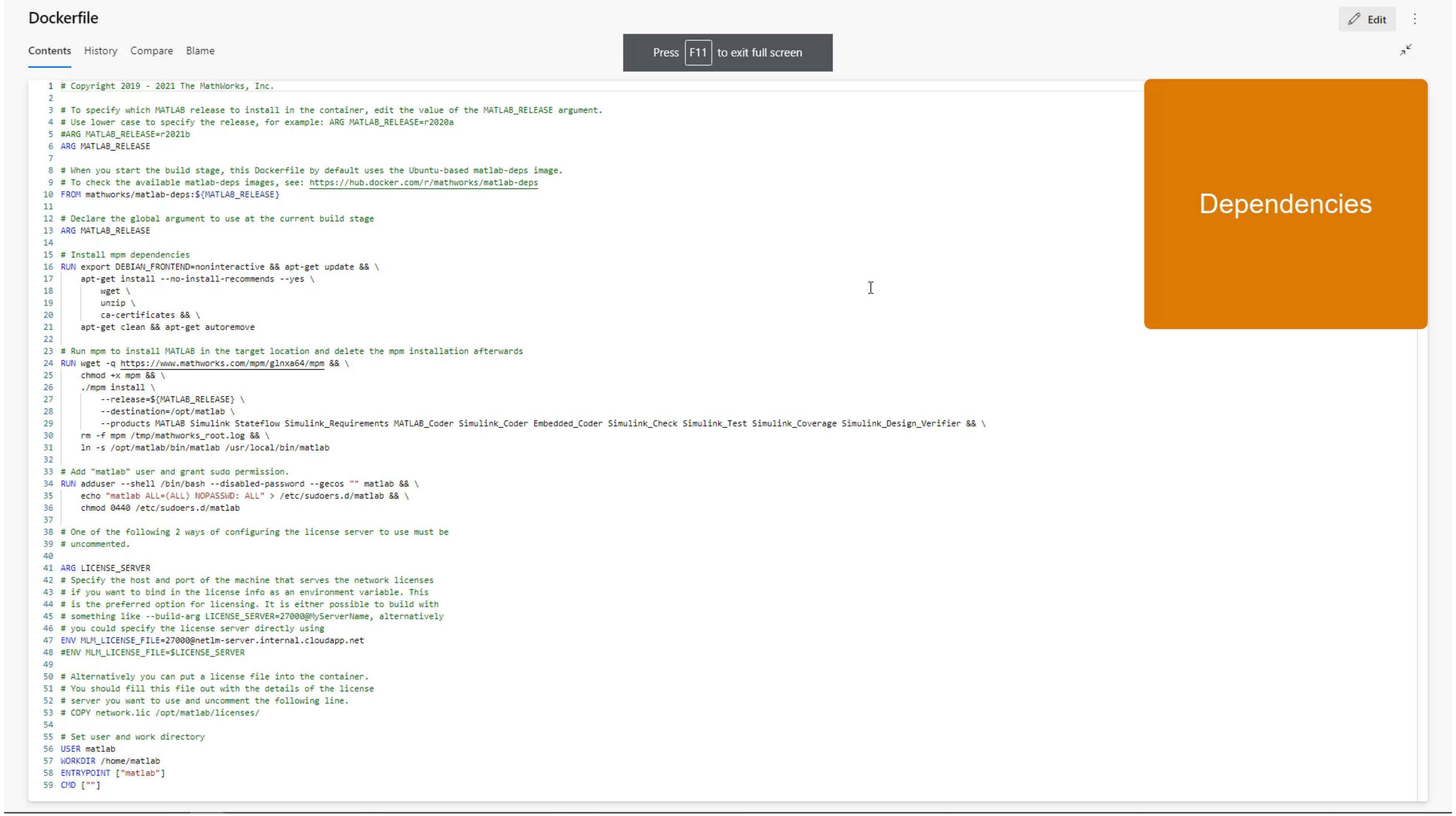
tableau Qlik

Power BI Spotfire TIBCO Software

Agenda

- Contribution of CI and MBD to DevOps
- Example: Scale up CI in the cloud
- Best Practice
- Summary

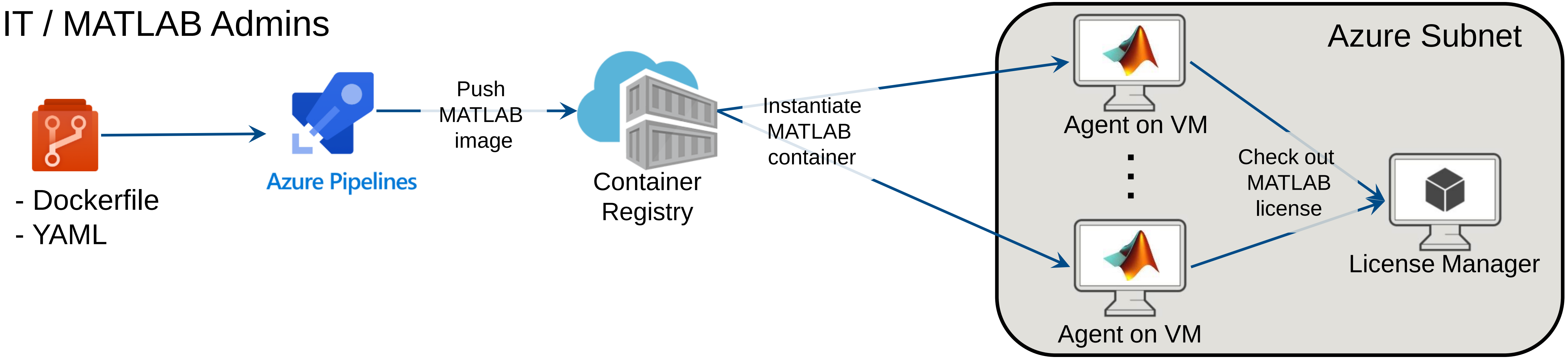
[Video] Example CI Workflow using Azure Pipelines - IT sets up the architecture



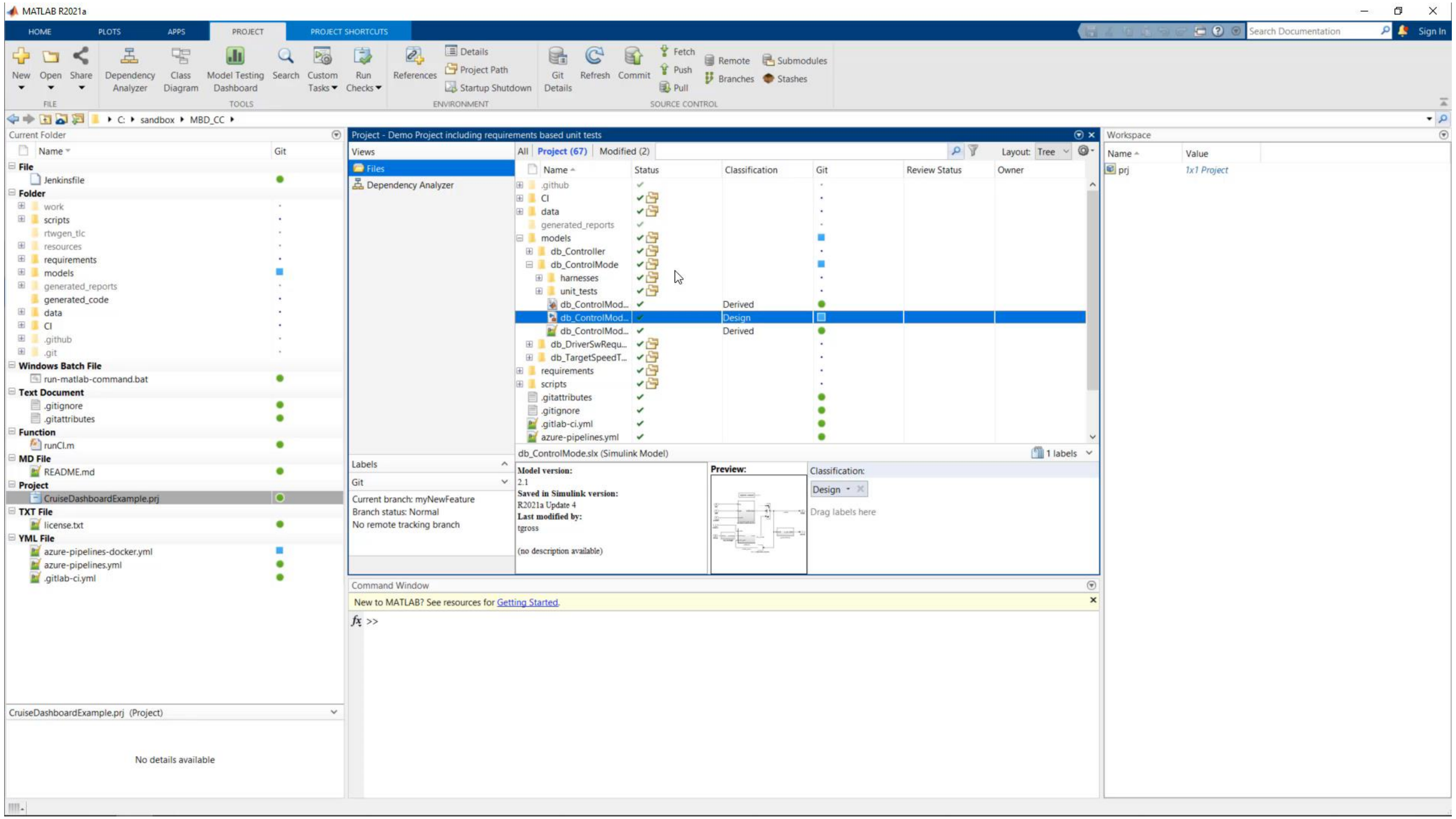
```dockerfile
# Copyright 2019 - 2021 The MathWorks, Inc.

# To specify which MATLAB release to install in the container, edit the value of the MATLAB_RELEASE argument.
# Use lower case to specify the release, for example: ARG MATLAB_RELEASE=r2020a
#ARG MATLAB_RELEASE=r2021b
ARG MATLAB_RELEASE

# When you start the build stage, this Dockerfile by default uses the Ubuntu-based matlab-deps image.
# To check the available matlab-deps images, see: https://hub.docker.com/r/mathworks/matlab-deps
FROM mathworks/matlab-deps:${MATLAB_RELEASE}

# Declare the global argument to use at the current build stage
ARG MATLAB_RELEASE

# Install mpm dependencies
RUN export DEBIAN_FRONTEND=noninteractive && apt-get update && \
    apt-get install --no-install-recommends --yes \
        wget \
        unzip \
        ca-certificates && \
    apt-get clean && apt-get autoremove

# Run mpm to install MATLAB in the target location and delete the mpm installation afterwards
RUN wget -q https://www.mathworks.com/mpm/glnxa64/mpm && \
    chmod +x mpm && \
    ./mpm install \
        --release=${MATLAB_RELEASE} \
        --destination=/opt/matlab \
        --products MATLAB Simulink Stateflow Simulink_Requirements MATLAB_Coder Simulink_Coder Embedded_Coder Simulink_Check Simulink_Test Simulink_Coverage Simulink_Design_Verifier && \
    rm -f mpm /tmp/mathworks_root.log && \
    ln -s /opt/matlab/bin/matlab /usr/local/bin/matlab

# Add "matlab" user and grant sudo permission.
RUN adduser --shell /bin/bash --disabled-password --gecos "" matlab && \
    echo "matlab ALL=(ALL) NOPASSWD: ALL" > /etc/sudoers.d/matlab && \
    chmod 0440 /etc/sudoers.d/matlab

# One of the following 2 ways of configuring the license server to use must be
# uncommented.

ARG LICENSE_SERVER
# Specify the host and port of the machine that serves the network licenses
# if you want to bind in the license info as an environment variable. This
# is the preferred option for licensing. It is either possible to build with
# something like --build-arg LICENSE_SERVER=27000@MyServerName, alternatively
# you could specify the license server directly using
ENV MLM_LICENSE_FILE=27000@netlm-server.internal.cloudapp.net
#ENV MLM_LICENSE_FILE=$LICENSE_SERVER

# Alternatively you can put a license file into the container.
# You should fill this file out with the details of the license
# server you want to use and uncomment the following line.
# COPY network.lic /opt/matlab/licenses/

# Set user and work directory
USER matlab
WORKDIR /home/matlab
ENTRYPOINT ["matlab"]
CMD [""]
```

Dependencies

Example architecture for containerized CI pipeline



[Video] Example CI Workflow using Azure Pipelines - Engineers leverage containerized MBD pipeline



Name	Git
File	
Jenkinsfile	●
Folder	
work	·
scripts	·
rtwgen_tlc	·
resources	·
requirements	·
models	■
generated_reports	·
generated_code	·
data	·
CI	·
.github	·
.git	·
Windows Batch File	
run-matlab-command.bat	●
Text Document	
.gitignore	●
.gitattributes	●
Function	
runCI.m	●
MD File	
README.md	●
Project	
CruiseDashboardExample.prj	●
TXT File	
license.txt	●
YML File	
azure-pipelines-docker.yml	■
azure-pipelines.yml	●
.gitlab-ci.yml	●

CruiseDashboardExample.prj (Project)

No details available

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	✓		●		

db_ControlMode.slx (Simulink Model) 1 labels

Model version:
2.1
Saved in Simulink version:
R2021a Update 4
Last modified by:
tgross

(no description available)

Preview:

Classification:
Design
Drag labels here

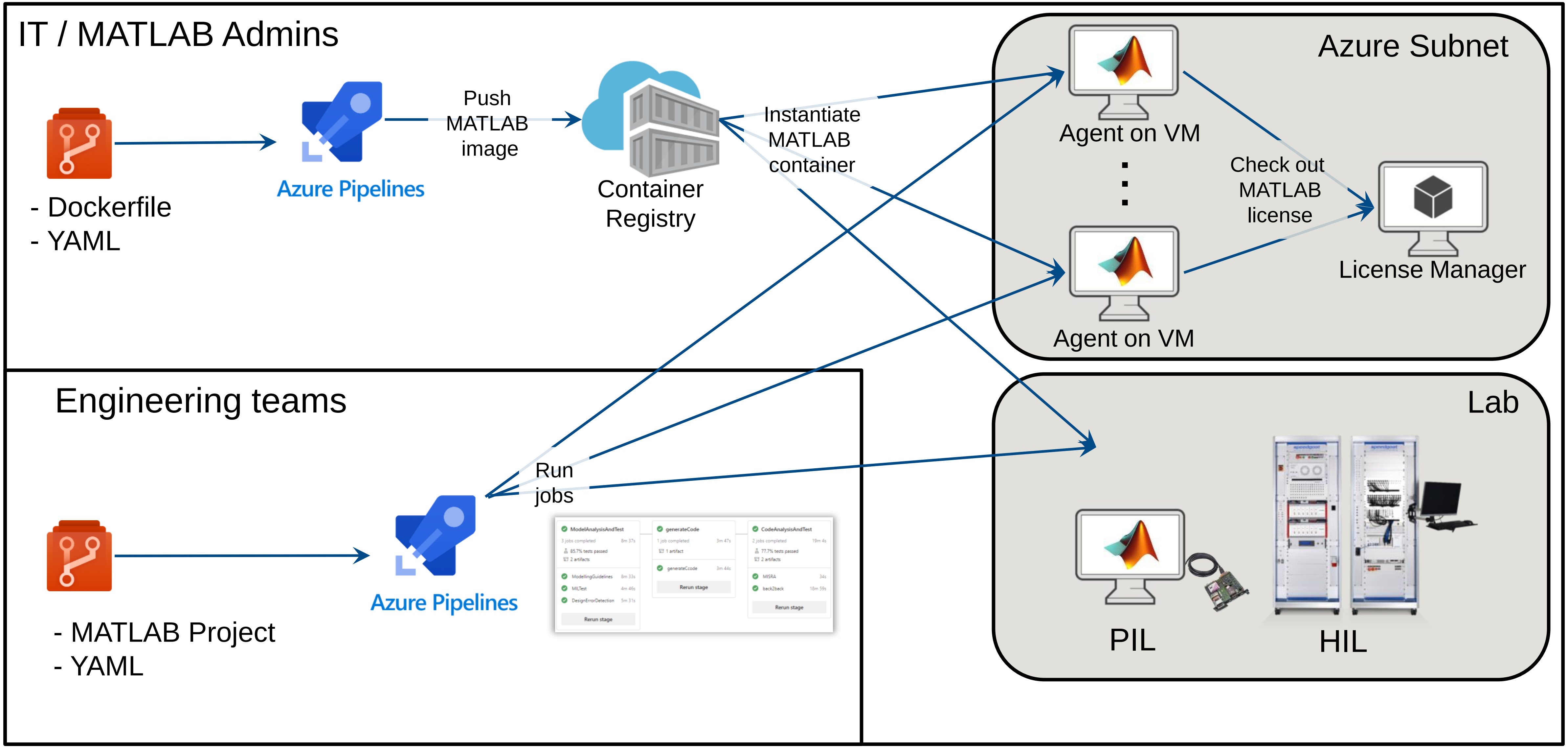
Current branch: myNewFeature
Branch status: Normal
No remote tracking branch

Name	Value
prj	1x1 Project

New to MATLAB? See resources for Getting Started.

```
fx >>
```

Example architecture for containerized CI pipeline



ModelAnalysisAndTest
3 jobs completed 8m 37s
85.7% tests passed
2 artifacts
ModellingGuidelines 8m 33s
MILTest 4m 46s
DesignErrorDetection 5m 31s
Rerun stage

generateCode
1 job completed 3m 47s
1 artifact
generateCcode 3m 44s
Rerun stage

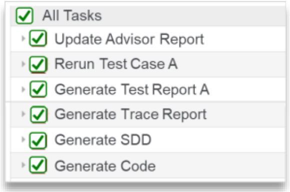
CodeAnalysisAndTest
2 jobs completed 19m 4s
77.7% tests passed
2 artifacts
MISRA 34s
back2back 18m 59s
Rerun stage

Agenda

- Contribution of CI and MBD to DevOps
- Example: Scale up CI in the cloud
- Best Practice
- Summary

Detect errors earlier & improve – Establish a mature issue detection process

CI support package



☑ All Tasks
☑ Update Advisor Report
☑ Rerun Test Case A
☑ Generate Test Report A
☑ Generate Trace Report
☑ Generate SDD
☑ Generate Code

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

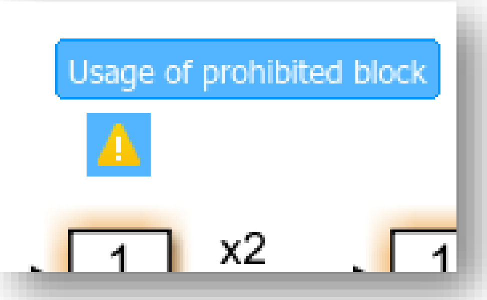
SAFe® PROVIDED BY SCALED AGILE®

https://www.scaledagileframework.com/accelerating-flow-with-devsecops-and-the-software-factory/

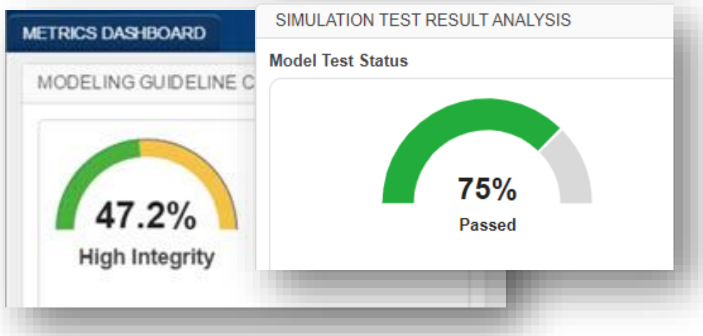
CI pipeline

Process

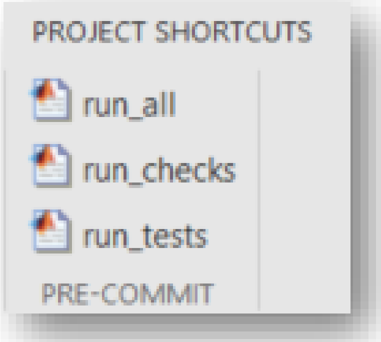
Pre-commit checks



Edit time checks

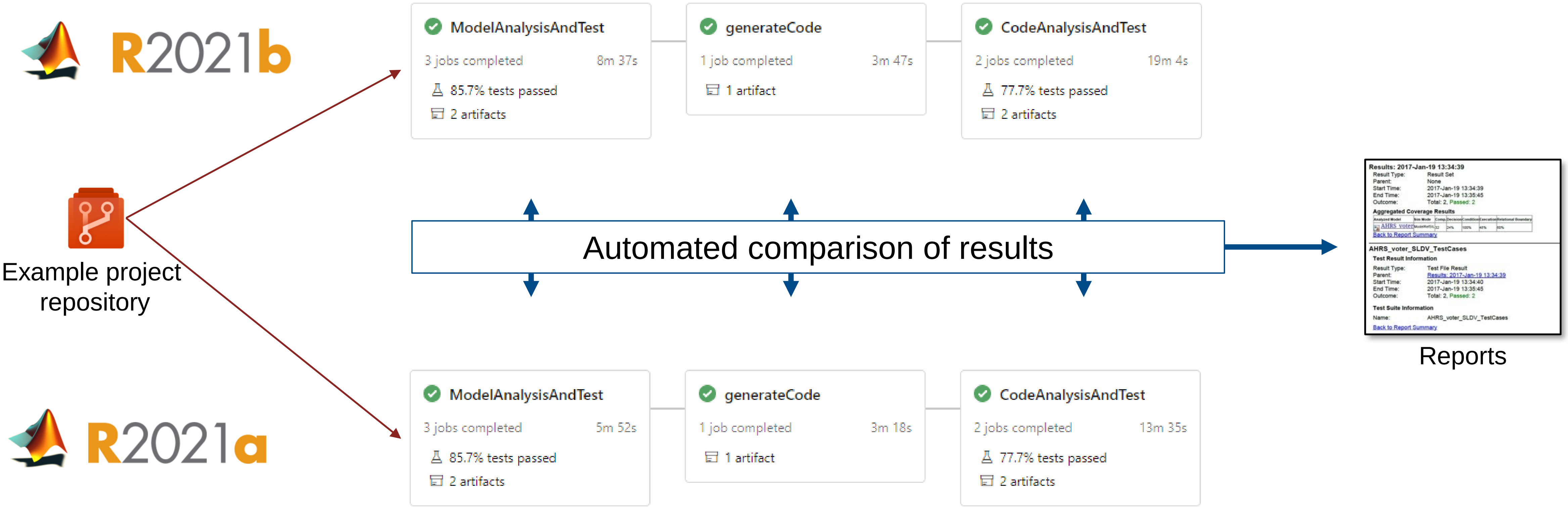


Quality Dashboards



MATLAB Project shortcuts

Enable continuous software updates – keep your toolchain upgradeable



R2021b pipeline:

- ModelAnalysisAndTest: 3 jobs completed, 8m 37s; 85.7% tests passed; 2 artifacts
- generateCode: 1 job completed, 3m 47s; 1 artifact
- CodeAnalysisAndTest: 2 jobs completed, 19m 4s; 77.7% tests passed; 2 artifacts

R2021a pipeline:

- ModelAnalysisAndTest: 3 jobs completed, 5m 52s; 85.7% tests passed; 2 artifacts
- generateCode: 1 job completed, 3m 18s; 1 artifact
- CodeAnalysisAndTest: 2 jobs completed, 13m 35s; 77.7% tests passed; 2 artifacts

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	ModelRefSIL	32	24%	100%	45%	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_TestCases

Back to Report Summary

Get in touch – Leverage MathWorks expertise for CI and DevOps

- Increase Automation and CI maturity:
  - Integrate with your CI platform, version control and other systems
  - Detect errors earlier & improve – establish a mature issue detection process
- Enable Model-Based development toolchain upgrades
- Enhance system architecture to master complexity
  - Enable modularization for test automation and safety/cybersecurity compliance
- Address performance bottle-necks
  - Speed up simulations 2x-34x to get faster feedback with cache files created on CI

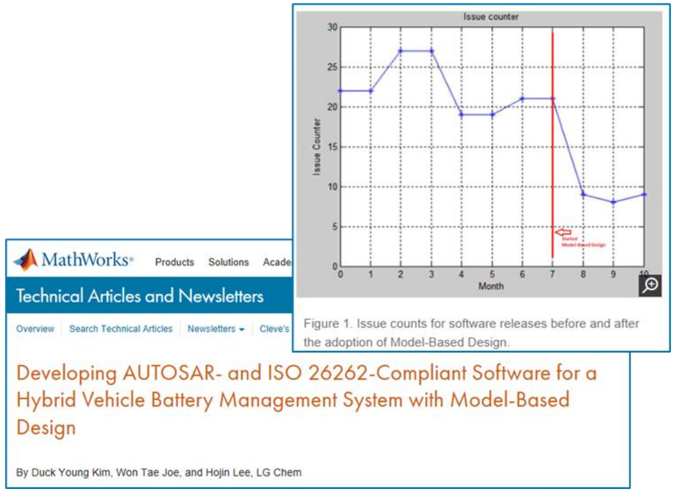
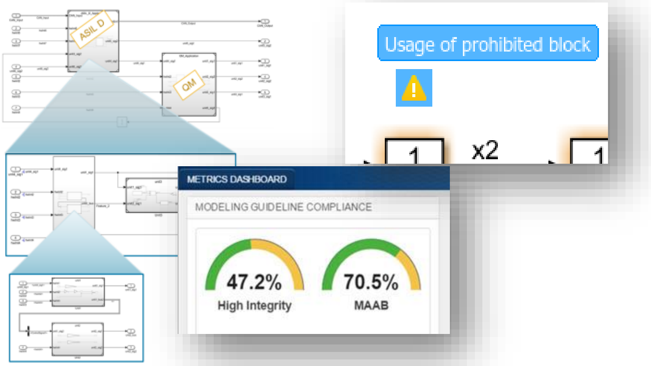
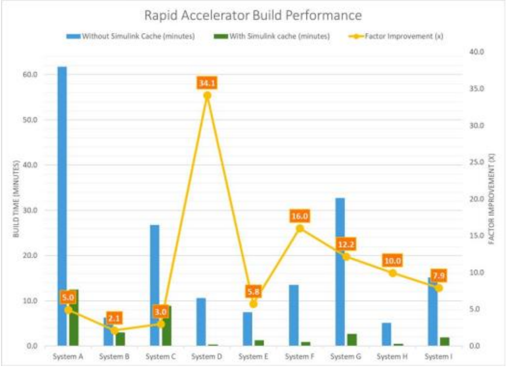


Figure 1. Issue counts for software releases before and after the adoption of Model-Based Design.

Developing AUTOSAR- and ISO 26262-Compliant Software for a Hybrid Vehicle Battery Management System with Model-Based Design

By Duck Young Kim, Won Tae Joe, and Hojin Lee, LG Chem





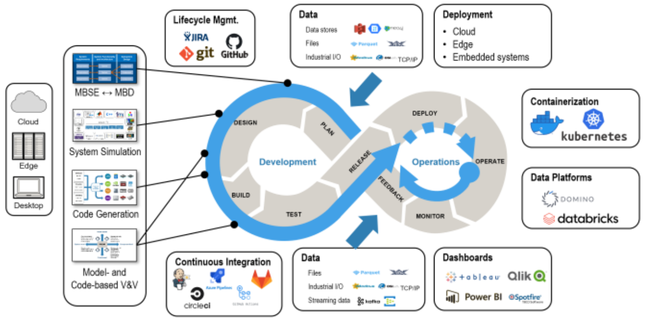
Key Takeaways

- Use DevOps and CI with Model-Based Design tackle software transformation challenges

	Throughput		Stability	
DORA metrics impact	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		

- Gain flexibility by scaling CI for MBD in the cloud

- Get in touch to enhance your CI workflow and implement best practice
  - AMER: Bernard Johnson (bjohnson@mathworks.com)
  - EMEA: Tjorben Gross (tgross@mathworks.com)


Lifecycle Mgmt.
JIRA
git
GitHub
Data
Data stores
Files
Industrial I/O
TCP/IP
Deployment
• Cloud
• Edge
• Embedded systems
Cloud
Edge
Desktop
MBSE ↔ MBD
System Simulation
Code Generation
Model- and Code-based V&V
DESIGN
PLAN
BUILD
TEST
Development
RELEASE
FEEDBACK
DEPLOY
OPERATE
MONITOR
Operations
Containerization
kubernetes
Data Platforms
DOMINO
databricks
Continuous Integration
circleci
Data
Files
Industrial I/O
TCP/IP
Streaming data
kafka
Dashboards
+tableau
Qlik
Power BI
Spotfire