



# Continuous Modeling with MATLAB and Microsoft Azure DevOps

Wojciech Halabis

Test and Validation Engineer

**CLAAS**







1. Introduction
2. Motivation
3. Solution
4. Microsoft Azure DevOps
5. Use Cases
6. Conclusion

# Introduction about myself

I am 33 years old.

Degree in Electronics (B.Eng.) and Engineering Physics (M.Sc.).

Professional software development since 2013.

Current position: Test Automation in IoT (Architectural design, Test specification and implementation, continuous testing...).

Working at CLAAS since August 2018.

5+ years experience in modular software design using “git flow”.



# CLAAS Company presentation





# CLAAS E SYSTEMS Company presentation



Connectivity &  
Data Management



Steering, Precision  
Applications & Terminals



Process  
Automation



Base Components & Architecture



1. Introduction
2. Motivation
3. Solution
4. Microsoft Azure DevOps
5. Use Cases
6. Conclusion

# Who may be interested in this presentation

Are your models complex?



Are there 10+ people working on the same model?



Are those people working at different locations, even countries?



Do you feel that your teams develop several times similar stuff, e.g. libraries?

Have you experienced problems during integration of MATLAB Simulink models?

Do you want to adopt agile methodologies?



# Best practices for modular mode design

Use model reference for unit-level models.

Pick a strategy for grouping units into features.

Classify models according to safety requirements.

Avoid algorithmic content at the integration level.

Use model metrics to monitor unit complexity.

Define the role of model architects and integrators.



[5]



[6]





1. Introduction
2. Motivation
3. **Solution**
4. Microsoft Azure DevOps
5. Use Cases
6. Conclusion

# Taking the approach of other industries

---

These problems are not new, there are a lot of engineers that have been struggling for years with them.

So... why don't we use some of their approaches?

Do you know who I'm talking about?

**SOFTWARE ENGINEERING**



1. Introduction
2. Motivation
3. Solution
4. Microsoft Azure DevOps
5. Use Cases
6. Conclusion



# Covering the entire application lifecycle with Microsoft Azure DevOps

It provides version control.



It provides reporting and requirements management.



[16]

It provides project management, for both agile software development and waterfall teams.



[4]

It provides automated build, testing and release management.



[17]

# Snapshots – Home page

**Azure DevOps** DefaultCollection / MathModelsCLAAS / Overview / Wiki

MathModelsCLAAS +

Wikis > MathModelsCLAAS.wiki

## Home

Halabis, Wojciech 1/31/2019 Revisions

Follow 0 Edit page + New page More

### Welcome to this project!

### Our Goal

The main purpose of this wiki is to enable a seamless transition into version control for MATLAB models. All the information related to this topic will be found here, you don't need to search anywhere else! We hope you get enough information, transparency and if not, you can tell us or much better, feel free to apporn new entries to this wiki!

Filter pages by title

**Pages**

- Home
  - Introduction to git
  - How To's!
    - Installation of dSpace
    - Git process
    - TFS Admins page
    - Gitignore explained
  - Installation of git
  - Git using Matlab
  - Git using Git Extensions
  - Git using Visual Studio
- Meeting notes
  - MATLAB\_Simulink\_Styleguide

Project settings

# Snapshots – Git repositories

The screenshot displays the Azure DevOps interface for a repository named "MathModelsCLAAS". The breadcrumb trail shows the path: DefaultCollection / MathModelsCLAAS / Repos / Files / Name of the repository. The left sidebar contains navigation options: Overview, Boards, Repos (highlighted with a green arrow), Files, Commits, Pushes, Branches, Tags, Pull requests, Pipelines, Test Plans, and Artifacts. The main content area shows the repository's commit history, with a table listing commit hashes, messages, and authors. The table has columns for Commit, Message, and Author. The commit history is filtered to show "Simple history" by "Author" from a specific date to another. The commit hashes listed are: 49dba66f, acfc9044, 354735a5, 45a5744b, 3d9400a3, bb337071, 3216c667, a941d784, 1e10b6fc, 8e4f8496, 19453108, and 525b8ca8. Two callout boxes are present: one pointing to the commit history table with the text "„Commits“ or history of the model repository" and another pointing to the Author column with the text "Responsible of the commit".



# Snapshots – Knowledge base

The screenshot displays the Azure DevOps interface for a Wiki page titled "Enable Source Control in MATLAB". The page content includes a navigation breadcrumb (DefaultCollection / MathModelsCLAAS / Overview / Wiki), a search bar, and a list of actions (Follow, Edit page, New page, More). The main content area features a text instruction: "Open MATLAB-> Preferences, go to MATLAB->General->Source Control select 'Enable MathWorks source Control Integration'". Below this instruction is a screenshot of the MATLAB Preferences dialog box, specifically the "MATLAB General Source Control Preferences" section. In this dialog, the "Enable MathWorks source control integration" radio button is selected. A green arrow points to the "Wiki" menu item in the left sidebar. Another green arrow points to the "Enable Source Con..." page link in the right-hand "Pages" list.

DefaultCollection / MathModelsCLAAS / Overview / Wiki

Search

MathModelsCLAAS +

Overview

Summary

Dashboards

Analytics views\*

Wiki ←

Boards

Repos

Pipelines

Test Plans

Artifacts

Project settings <<

Wikis > MathModelsCLAAS.wiki v

## Enable Source Control in MATLAB

Halabis, Wojciech 5/7/2020 Revisions

Follow 0 Edit page + New page More v

Open MATLAB-> Preferences, go to MATLAB->General->Source Control select "Enable MathWorks source Control Integration"

Preferences

MATLAB General Source Control Preferences

Source control

Enable MathWorks source control integration

None

OK Cancel Apply Help

Filter pages by title

Pages

- Home
  - Introduction to git
  - How To's!
    - Installation of dSpace
    - Git process
    - TFS Admins page
    - Gitignore explained
    - Installation of git
    - Git using Matlab
      - ← Enable Source Con... \*\*\*
      - Typical git operations
      - Typical git operations (f...
    - Git using Git Extensions
    - Git using Visual Studio
  - Meeting notes
  - MATLAB\_Simulink\_Styleguide

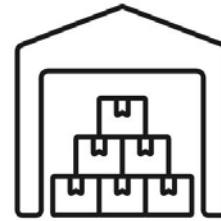
# Putting your models in version control system

Encourage collaboration.



[7]

Maintain properly our versions.



[8]

Restore previous versions.



[9]

Understand all the time what is happening.

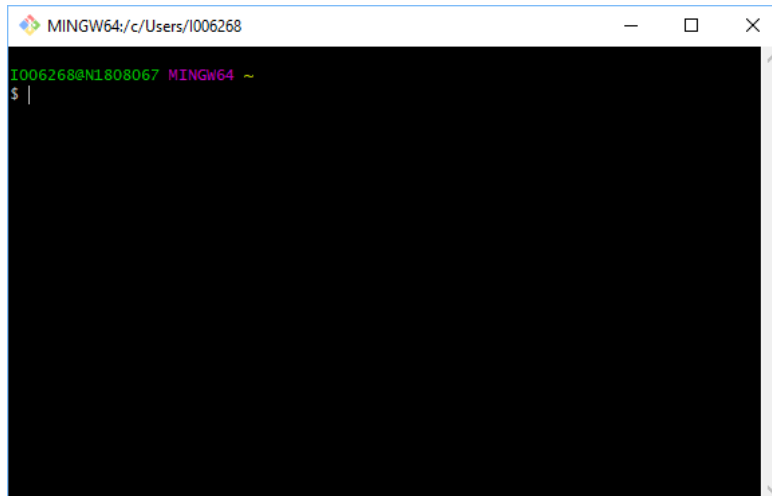


[10]

**ALL OF THAT JUST BY  
INTEGRATING PROPERLY A  
VERSION CONTROL SYSTEM IN  
YOUR COMPANY**

# Faster development with appropriate git tool

As a command line tool



With a git GUI

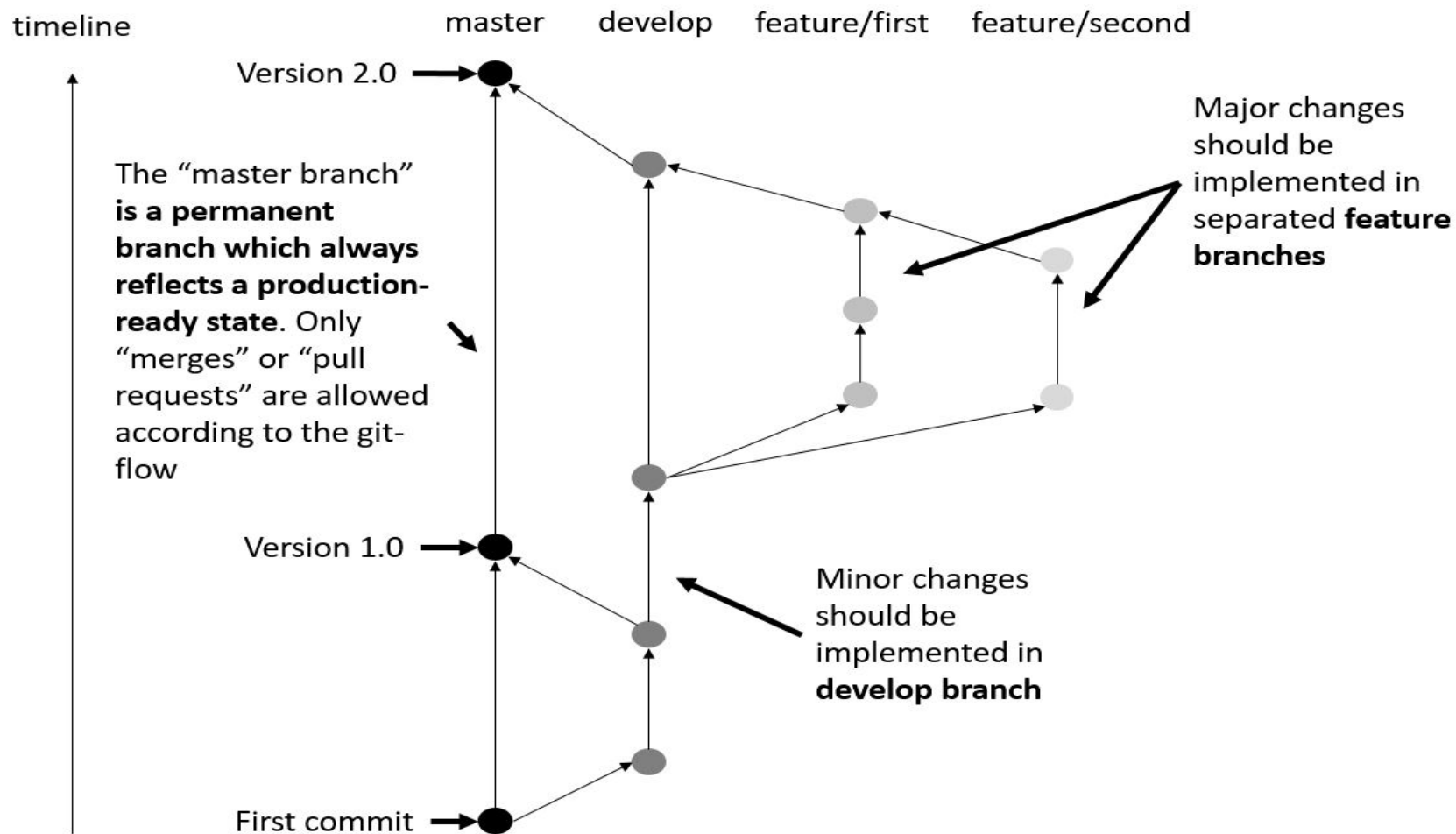


Integrated





# Developing models using the “git flow”

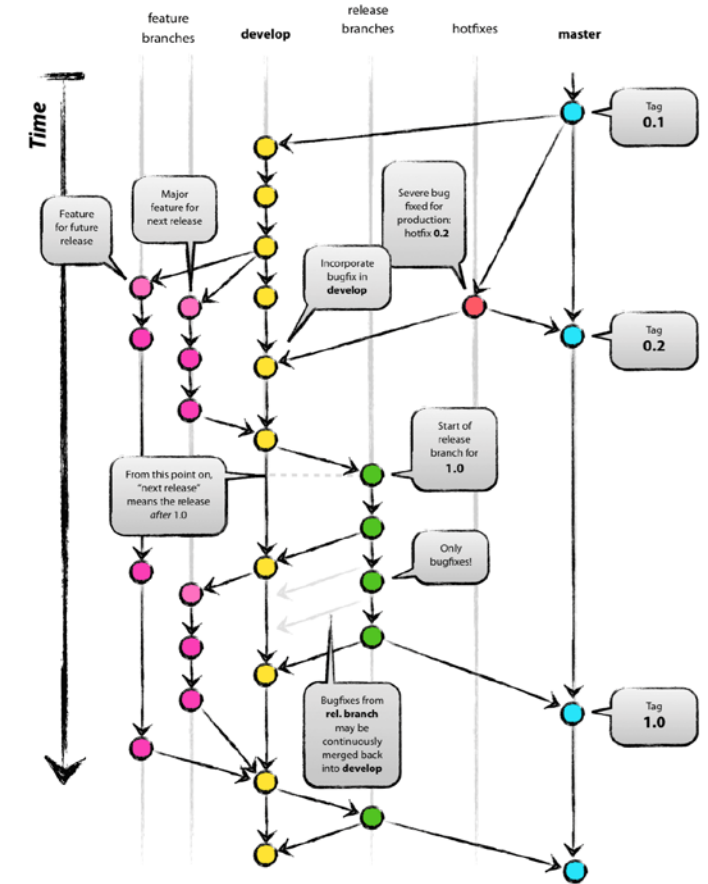


# Traceability out of the box

REQ: a function that calculates max of two values



Automatic creation of feature-branches for the model, important for almost automated release notes creation





1. Introduction
2. Motivation
3. Solution
4. Microsoft Azure DevOps
5. Use Cases
6. Conclusion



# Example of modular design with git submodules

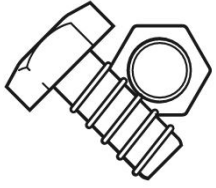
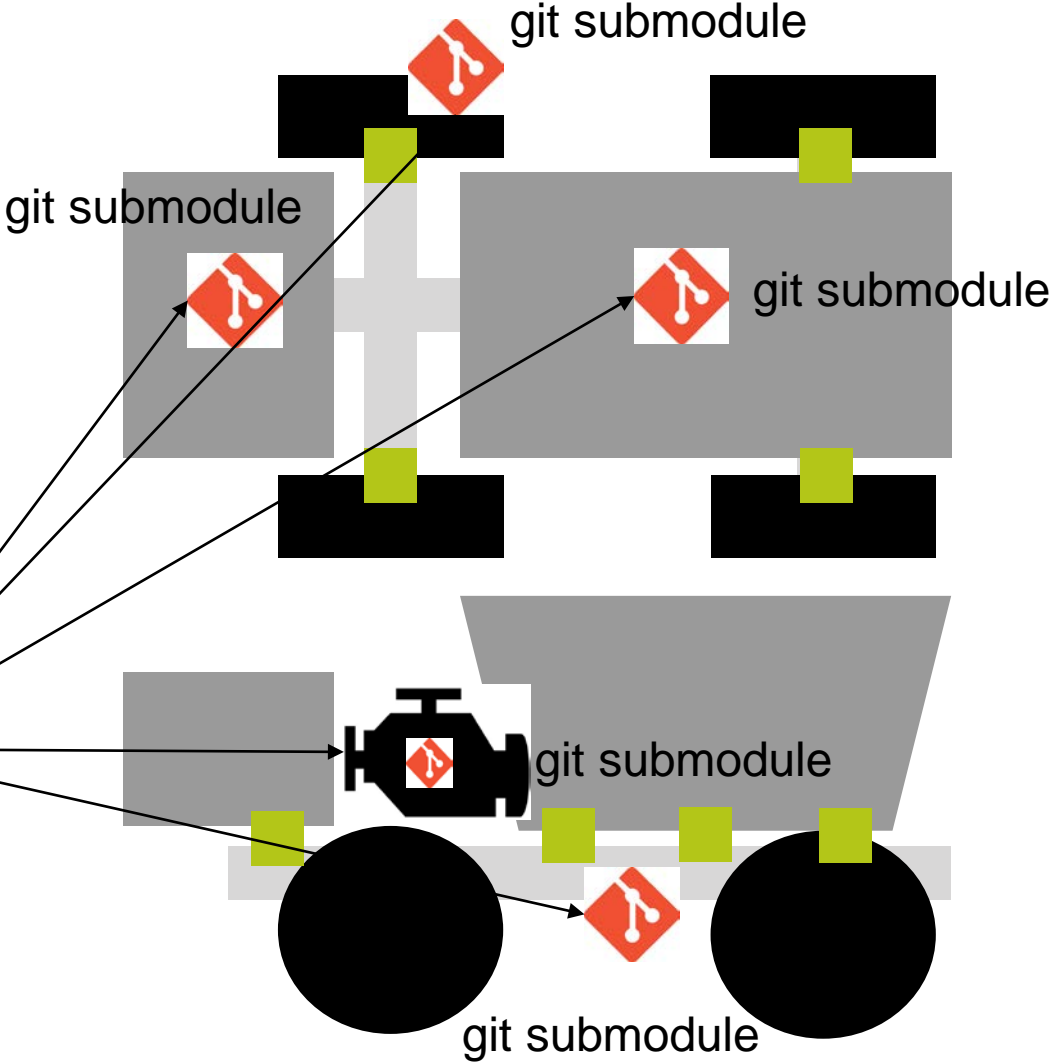


[14]

“Welding” for your interfaces



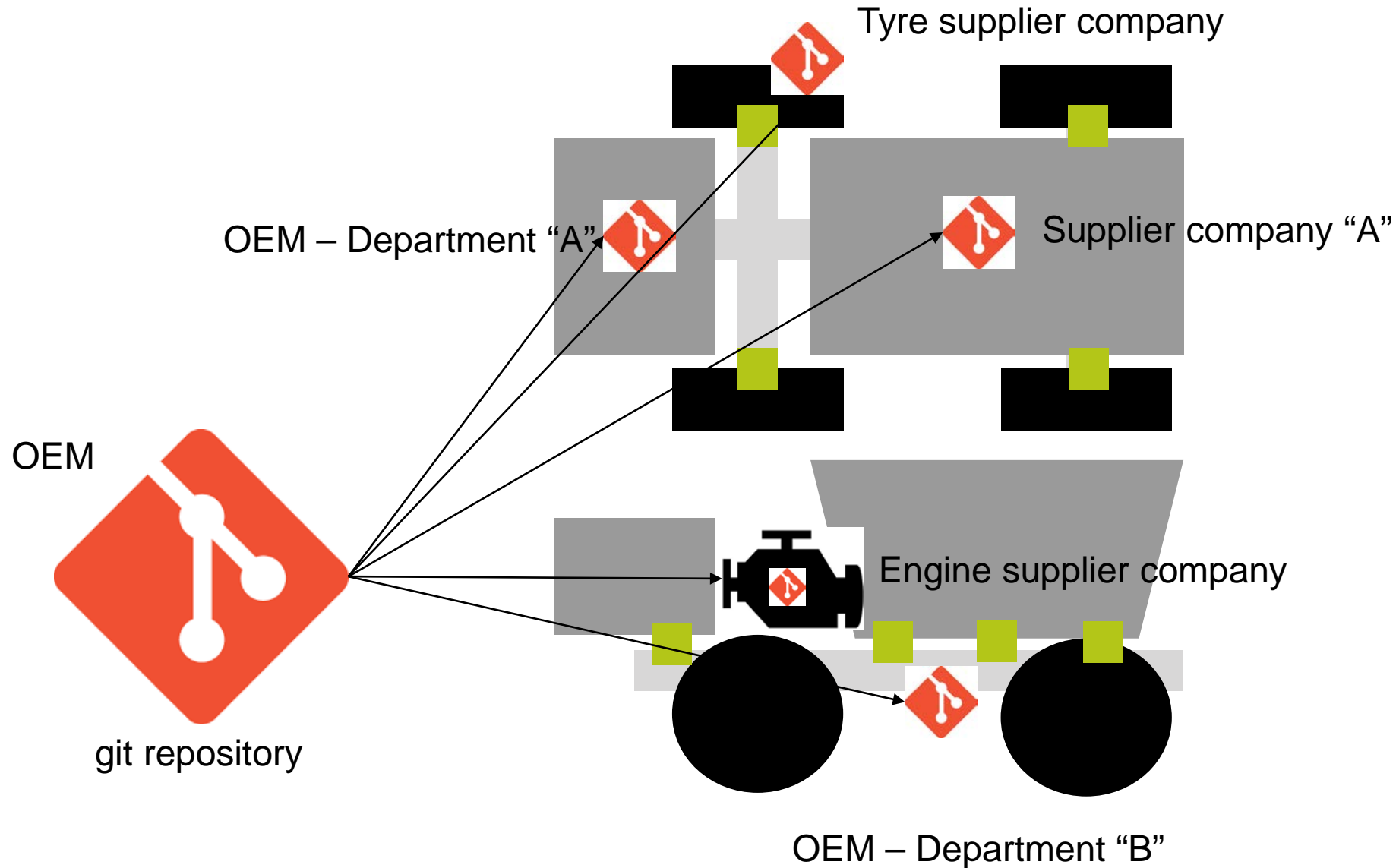
git repository



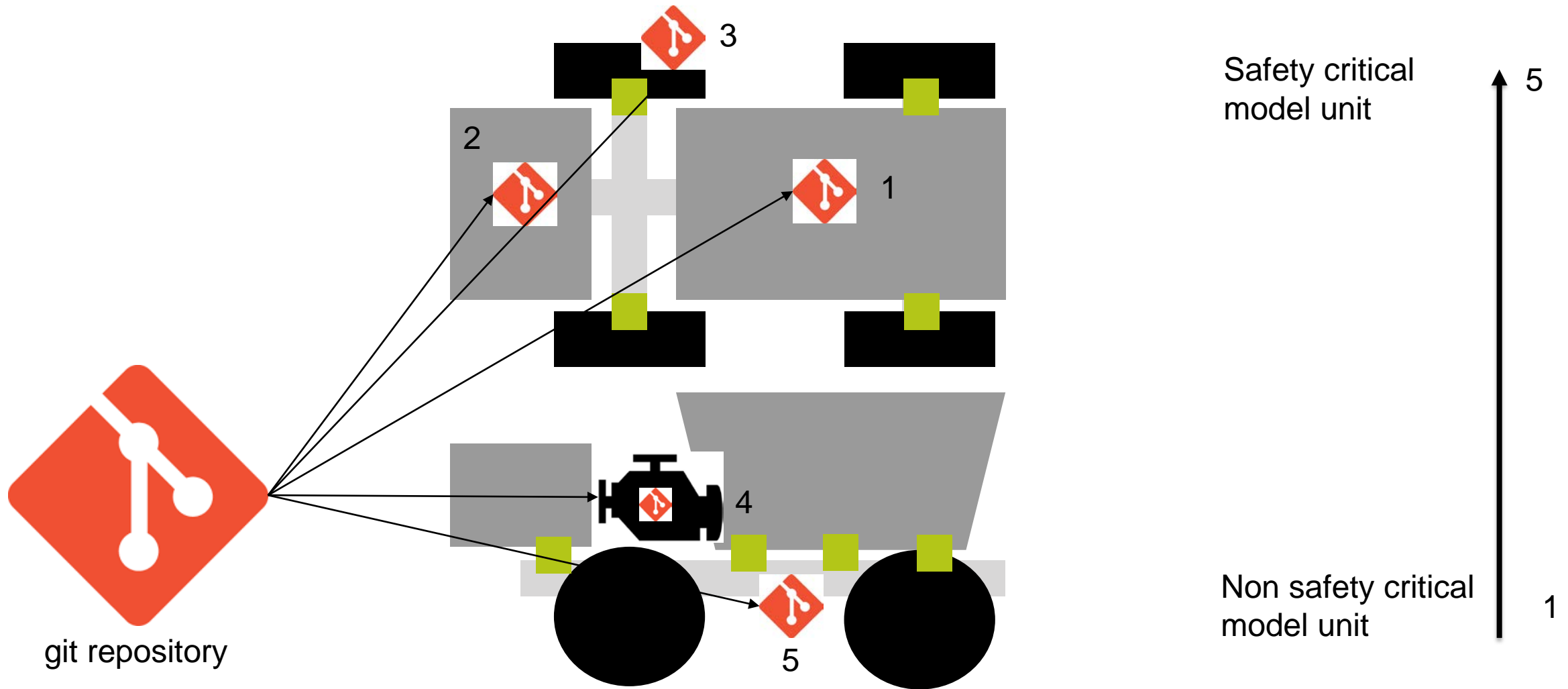
[15]

“Nuts and bolts” for your interfaces

# Externalization and reuse of model units



# Development efforts based on safety requirements of model units







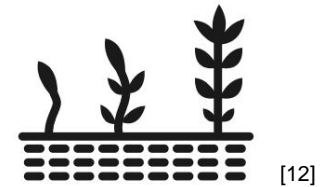
1. Introduction
2. Motivation
3. Solution
4. Microsoft Azure DevOps
5. Use Cases
6. Conclusion

# Where are we now at CLAAS

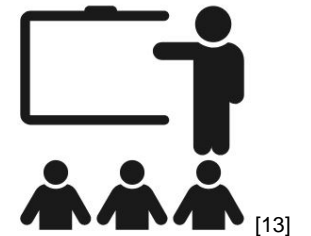
Different departments are already working using this methodology.



Growing number of machine Simulink models are being built on top of git submodules.



Departments that have developed a modelling process without git, are being trained and will adopt this methodologies in the near future.



# References

---

[1] <https://thenounproject.com/weltenraser/>

[2] <https://thenounproject.com/rshashank19/>

[3] <https://thenounproject.com/nhor/collection/global-business>

[4] <https://thenounproject.com/floicons/>

[5] <https://thenounproject.com/popcornartsgfx>

[6] <https://thenounproject.com/lastspark>

[7] <https://thenounproject.com/icondesign018>

[8] <https://thenounproject.com/victorulerz>

[9] <https://thenounproject.com/rshashank19/>

[10] <https://thenounproject.com/mirhashemihamed>

[11] <https://thenounproject.com/hatti27789service>

[12] <https://thenounproject.com/IconTrack/>

[13] <https://thenounproject.com/chatsudastock>

[14] <https://thenounproject.com/Flatart>

[15] <https://thenounproject.com/mnhendricks11>

[16] <https://thenounproject.com/AlfredoCreates>

[17] [https://thenounproject.com/coquet\\_adrien](https://thenounproject.com/coquet_adrien)



think e-farming.

Thank you for your attention.

More information: [ces-frontoffice@claas.com](mailto:ces-frontoffice@claas.com)

[wojciech.halabis@claas.com](mailto:wojciech.halabis@claas.com)