



Consistent architecture diagrams for C++ projects

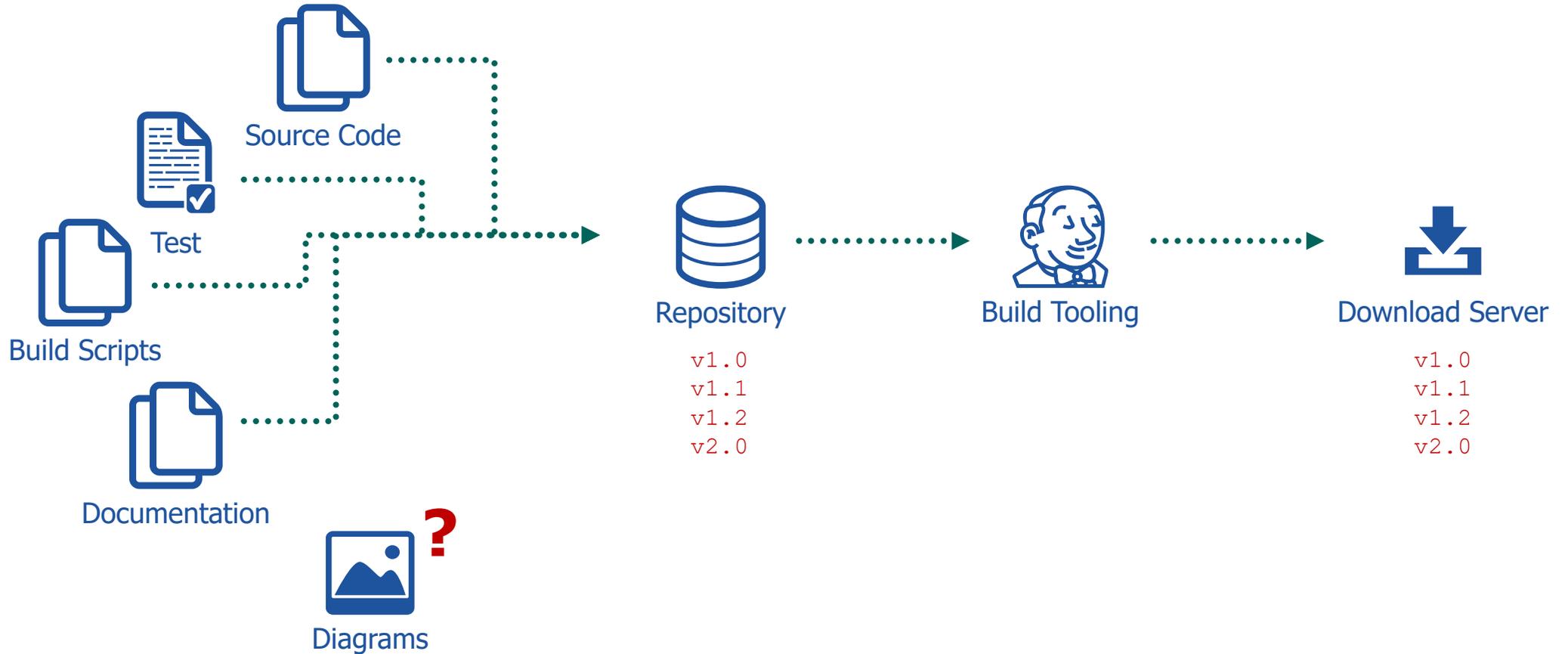
Marius Feilhauer

About me



- M.Sc. Automotive Engineering @ University Stuttgart
- PhD thesis *Simulation-based validation of advanced driver assistance systems* @ HLRS University Stuttgart (Prof. Resch) and ETAS GmbH
- Working experience @ Robert Bosch GmbH, Aston Martin Ltd. and Porsche AG
- Software developer of C++ based simulation models @ ETAS GmbH
- Lectures @ University Stuttgart

Introduction and Overview



Agenda

- Good documentation
- Architecture and code visualization
- Consistency
- Diagrams as code
- Takeaways & Outlook



Good Documentation

Golden nuggets

Guiding principles by atlassian



Keep it brief



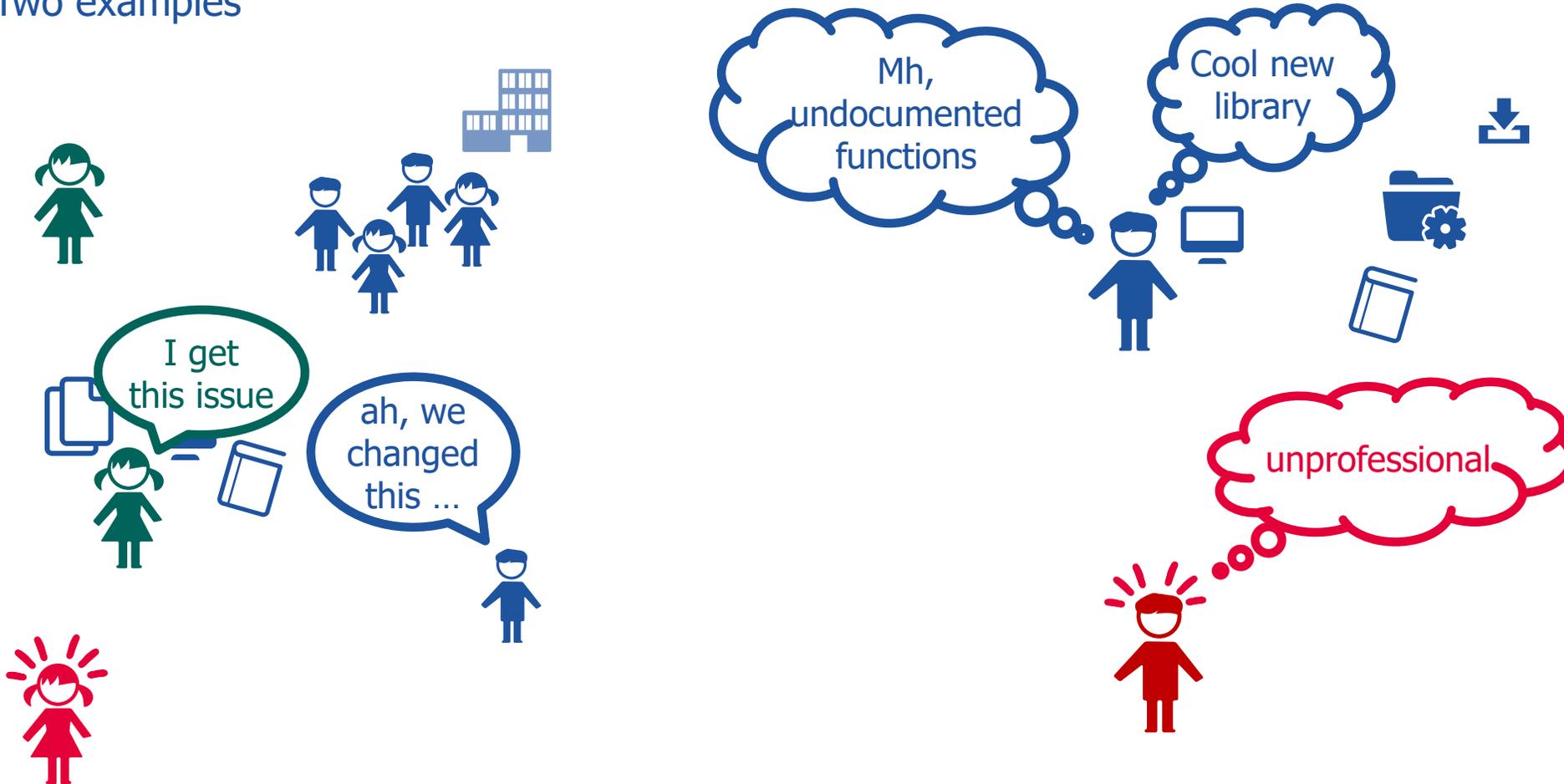
Visuals are key



Know your audience

Documentation use cases

Two examples



Adam Scott, www.oreilly.com/ideas/the-eight-rules-of-good-documentation

The eight rules of good documentation

1. Write documentation that is inviting and clear
2. Write documentation that is comprehensive, detailing all aspects of the project
3. Write documentation that is skimmable
4. Write documentation that offers examples of how to use the software
5. Write documentation that has repetition, when useful
6. Write documentation that is up-to-date
7. Write documentation that is easy to contribute to
8. Write documentation that is easy to find



Fast understanding
of your intentions



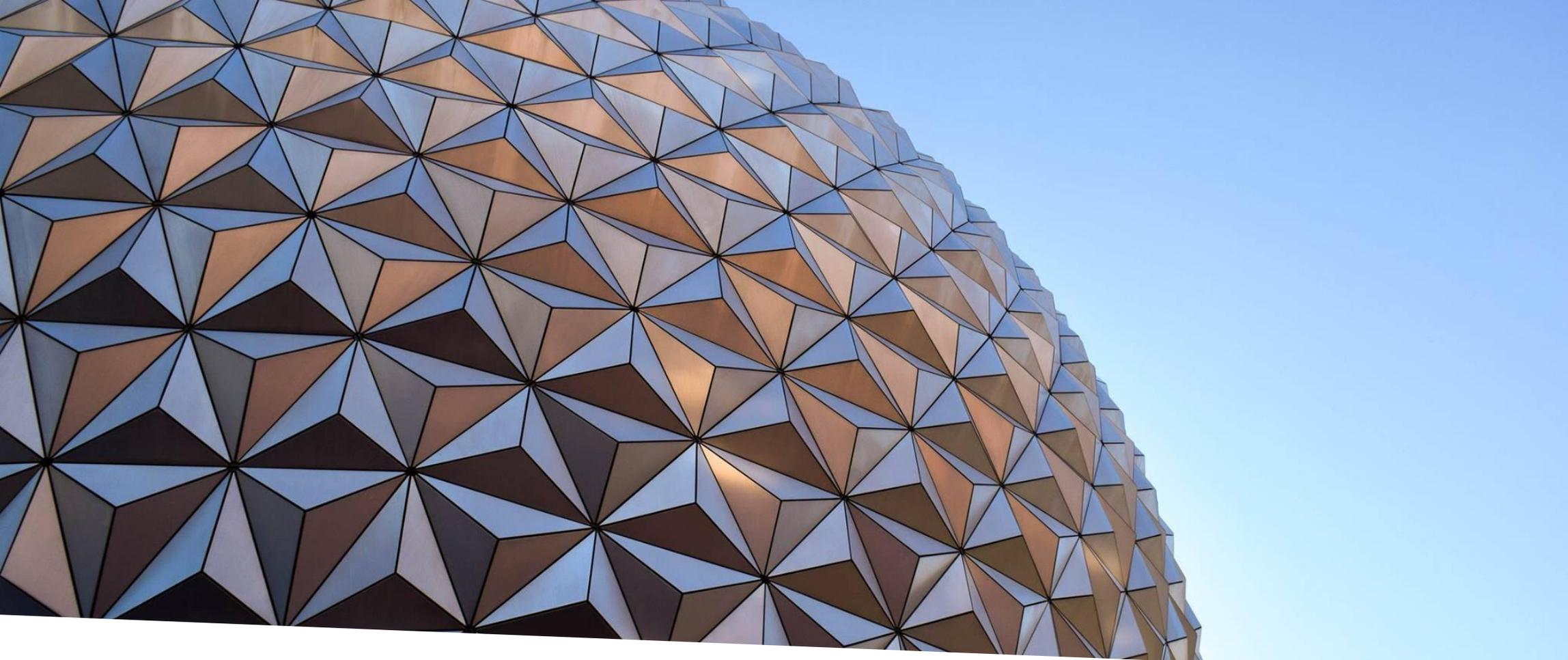
Professional
public image



Enforces to express
intentions precisely

Additional benefits

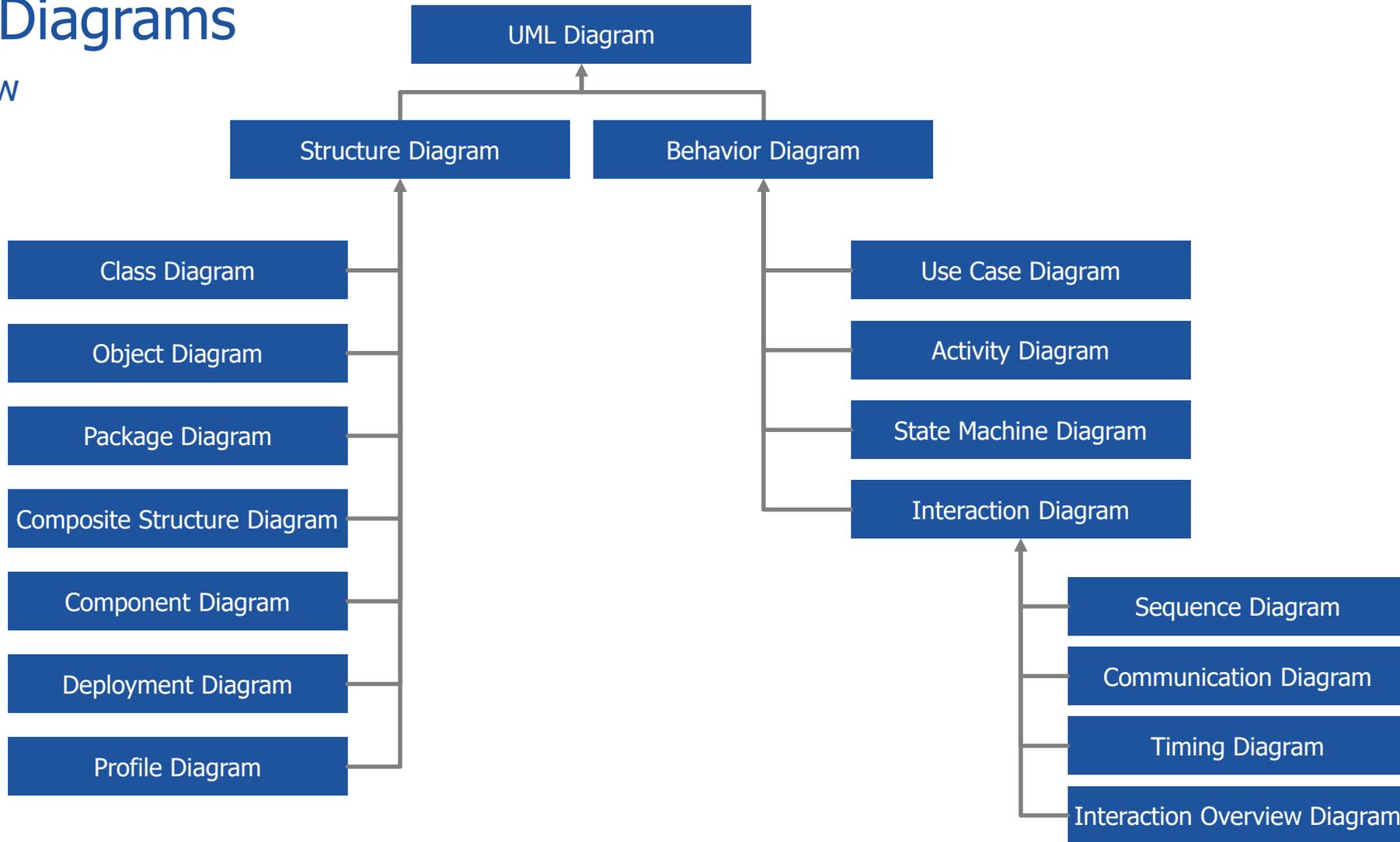
when using consistent diagrams within your documentation



Diagrams and code visualization

UML Diagrams

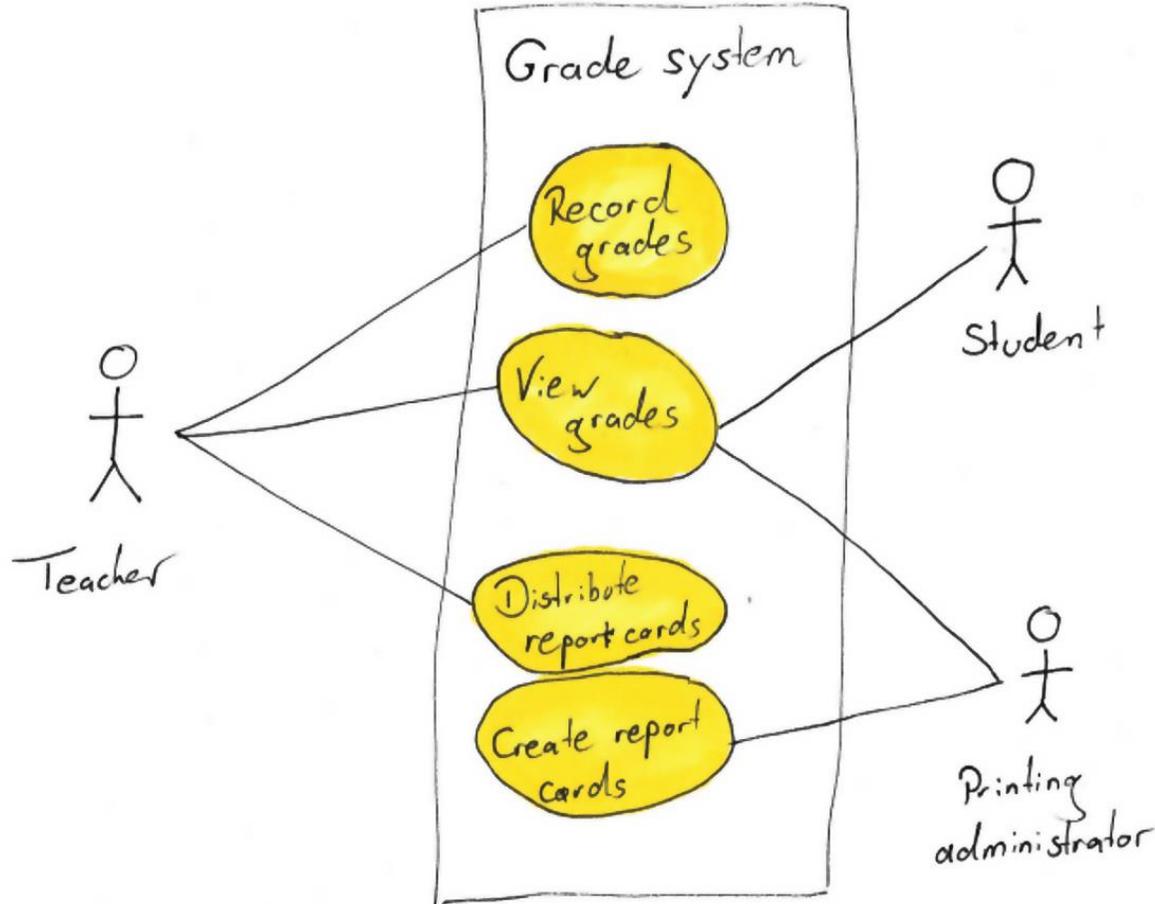
Overview



based on www.uml-diagrams.org/uml-25-diagrams.html

Grade System Example

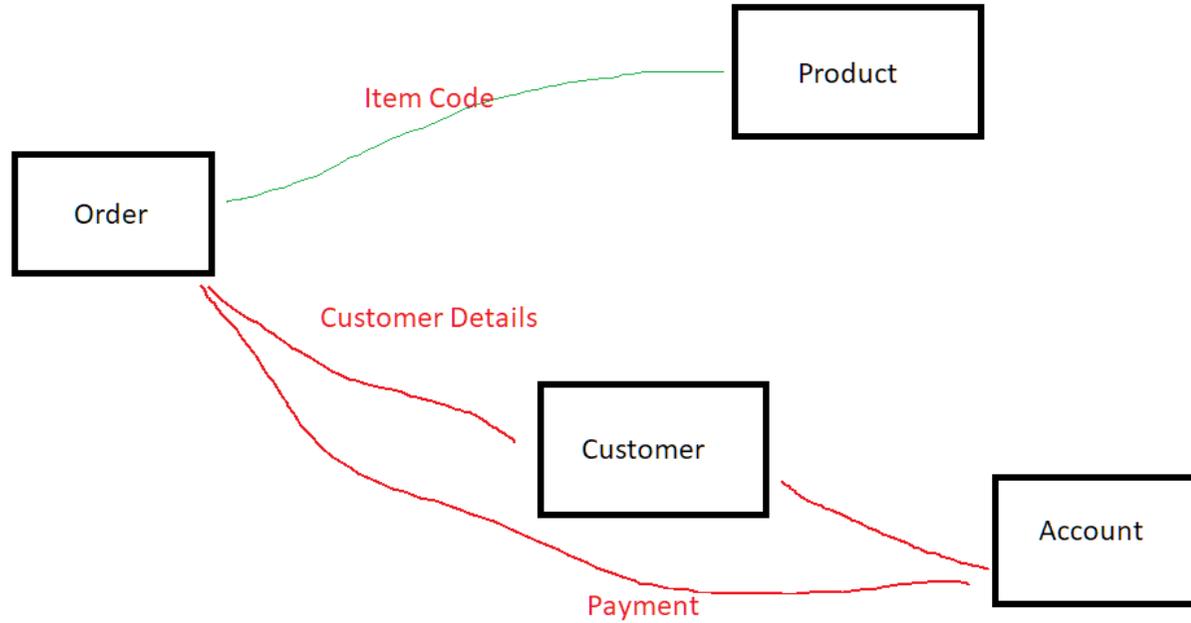
Use Case Diagram



Madeline Adair, Use Case Diagrams, www.slideplayer.com/slide/677748

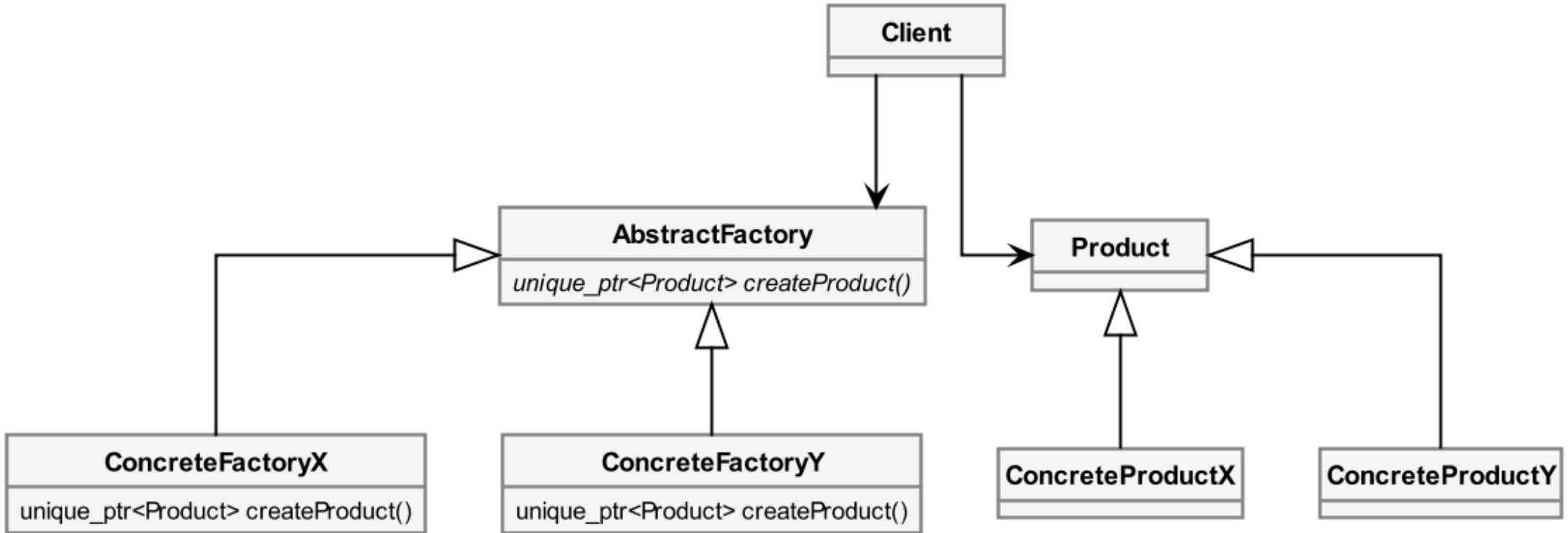
Shopping Example

UML Component Diagram Example



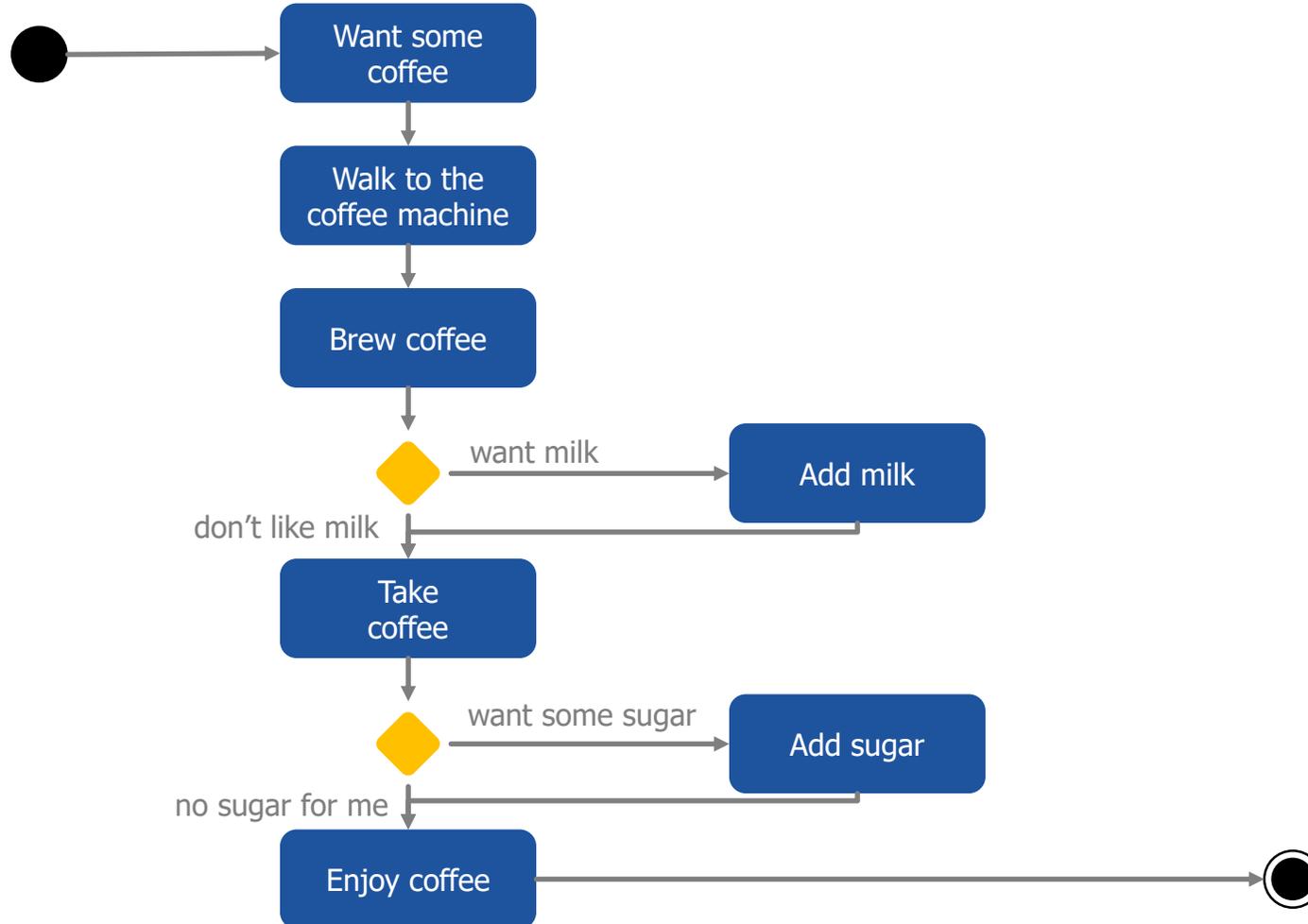
Abstract Factory Design Pattern

UML Class Diagram Example



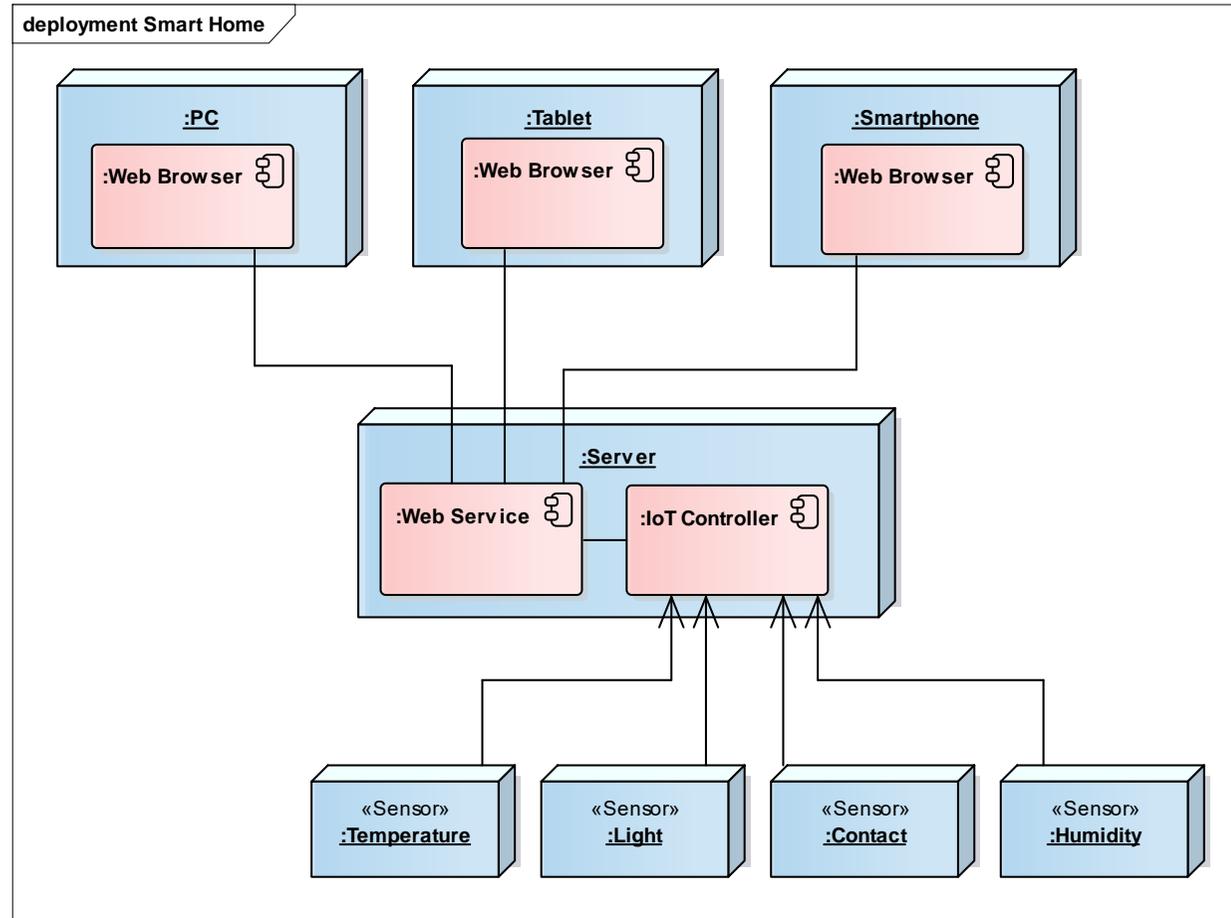
Let's get a coffee

Activity Diagram



Smart Home Example

Deployment Diagram

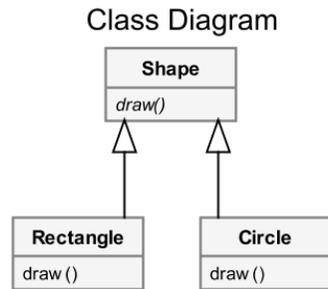
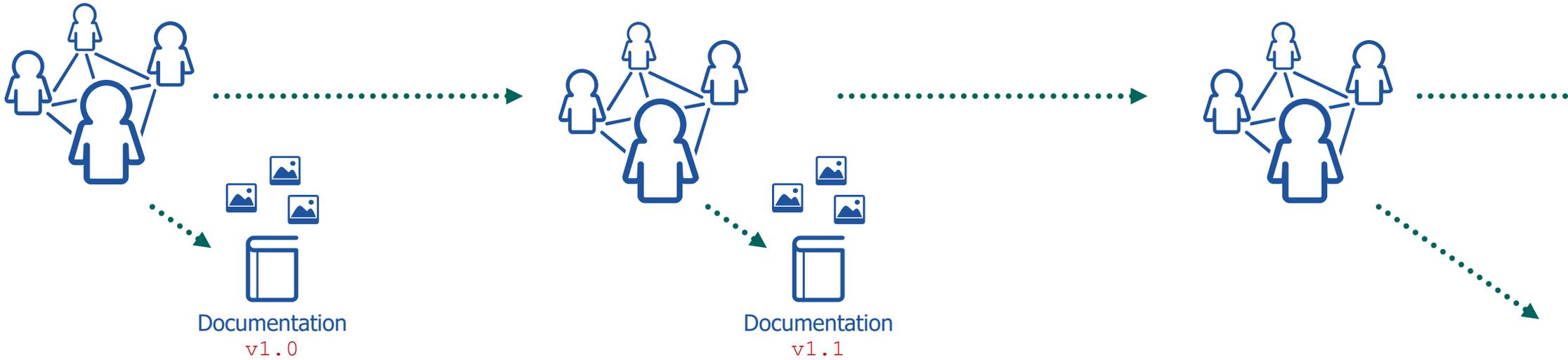


Diagrams allow to transport information easily
and there are many different options available how to draw them

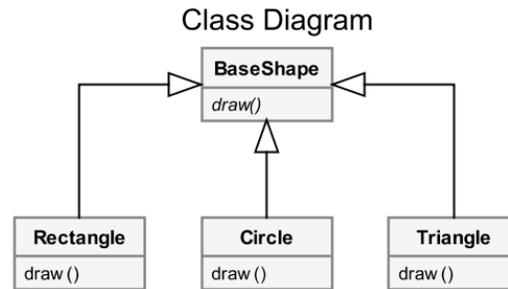


Consistency

Consistency



Meeting C++ example



Meeting C++ example

Draw diagrams by using code

Source code of diagrams has to be kept in-sync with the rest of the code and its documentations



Diagrams as code

Overview

Diagrams as code

- Available tools allow automated generation of diagrams
Doxygen, CMake, PlantUML
- Easy integration into existing build toolchains
- Automatic approaches vs. scripted diagrams

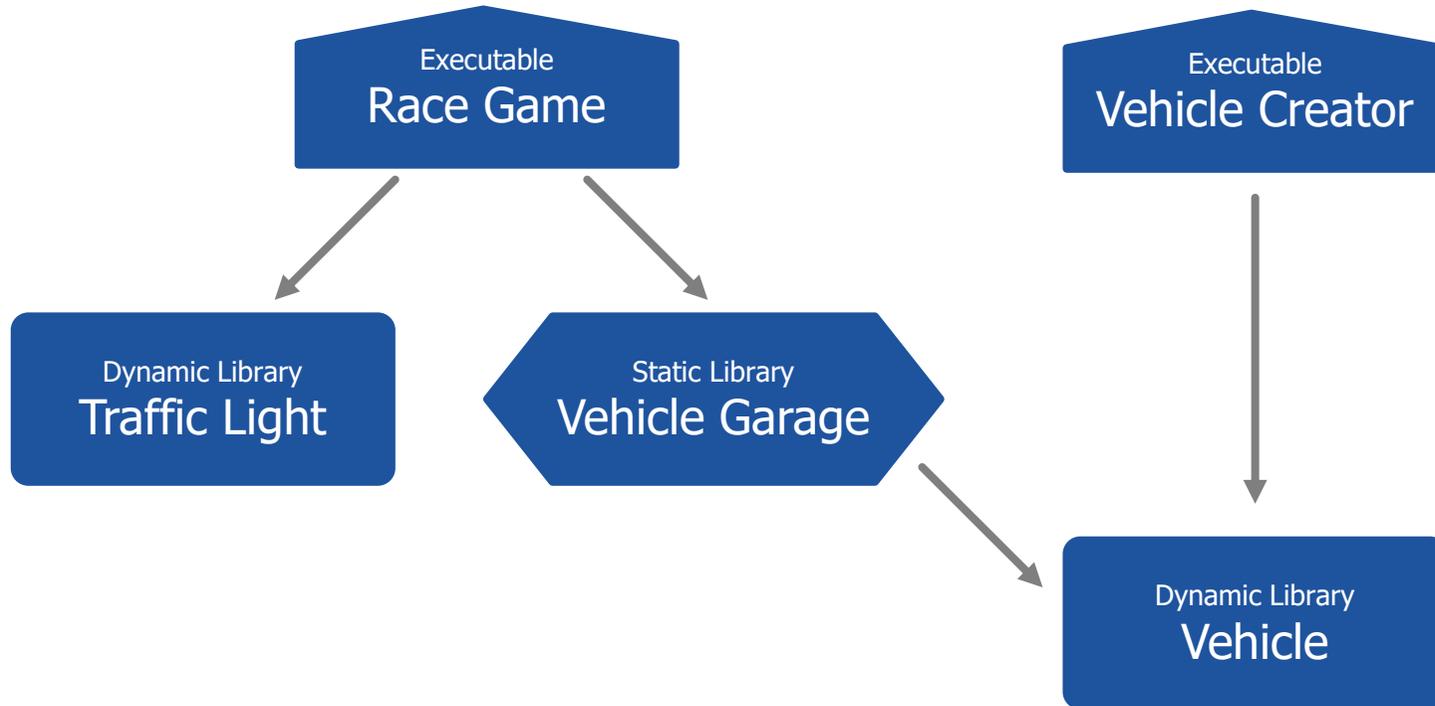
Example project

Directory structure



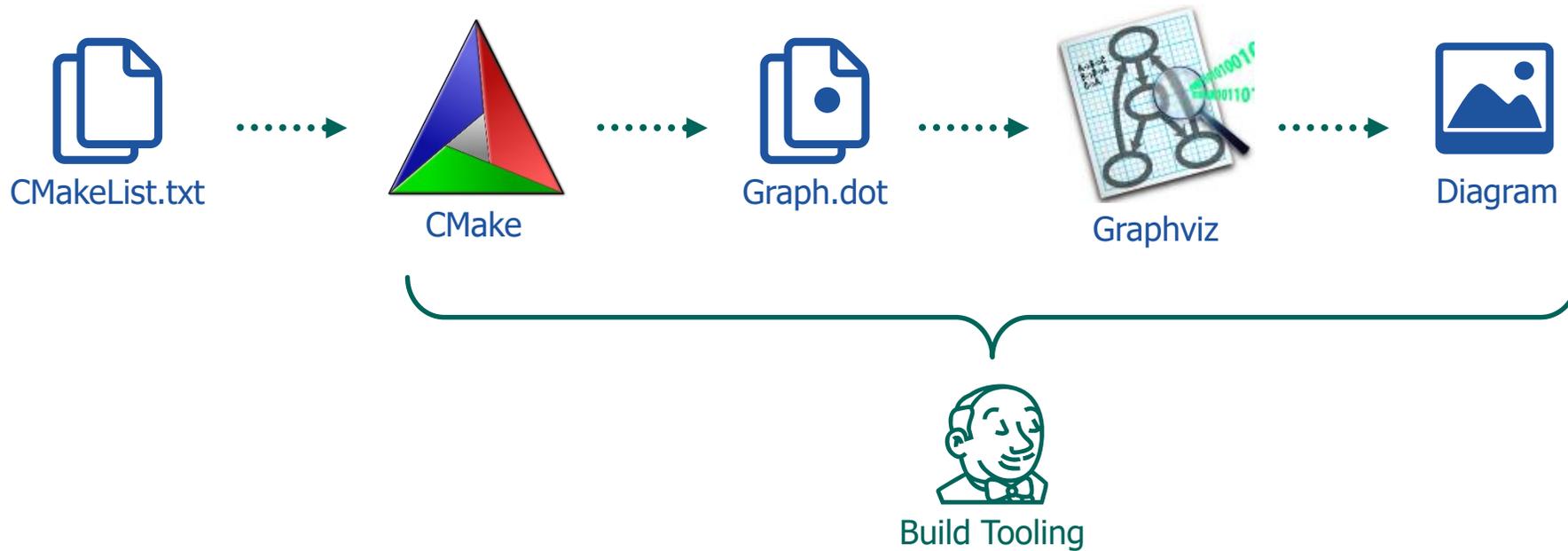
Example project

CMake targets



Dependency Graphs

Cmake & Graphviz



Dependency Graphs

Cmake & Graphviz

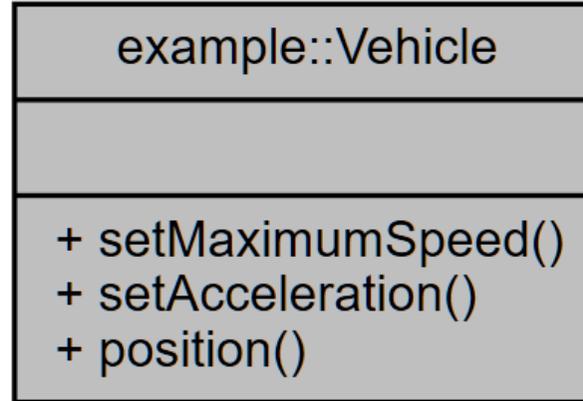
```
add_custom_target(DependencyGraph
    COMMAND cmake --graphviz=DependencyGraph.dot .
    COMMAND dot -Tsvg DependencyGraph.dot -o DependencyGraph.svg
    WORKING_DIRECTORY ${CMAKE_BINARY_DIR})
```

CMakeLists.txt

 [CMake Live Demo](#)

Collaboration Diagrams

Doxygen

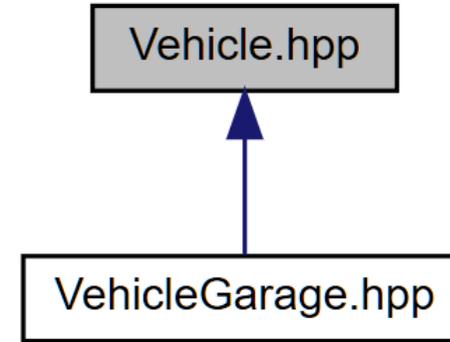
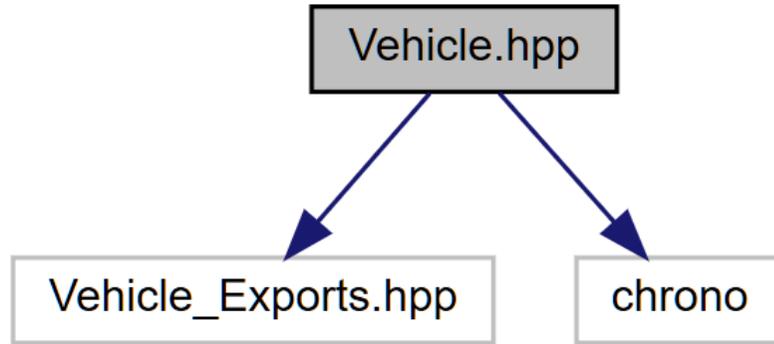


COLLABORATION_GRAPH = YES

doxyfile

Include Graphs

Doxygen

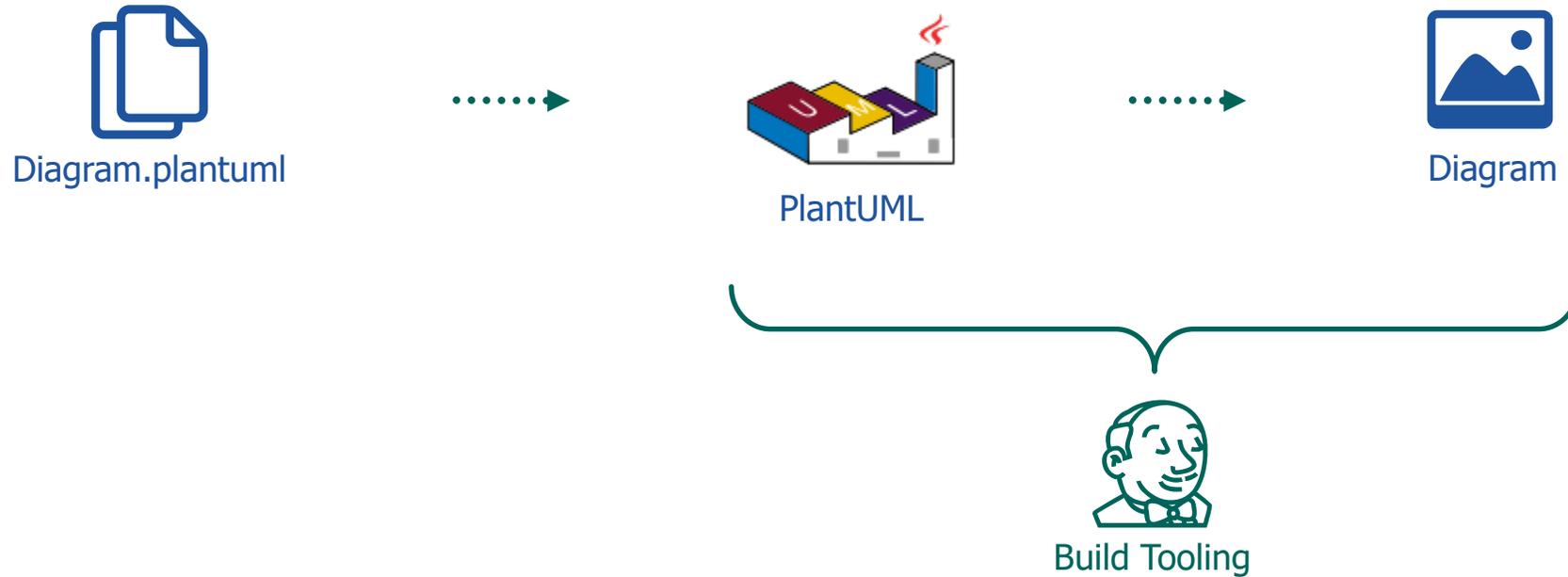


```
INCLUDE_GRAPH      = YES  
INCLUDED_BY_GRAPH = YES
```

doxyfile

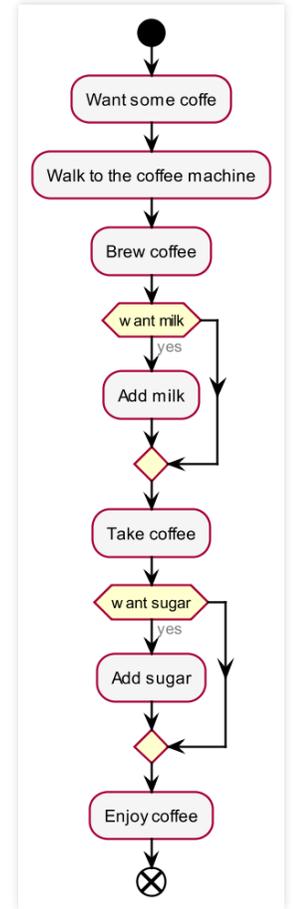
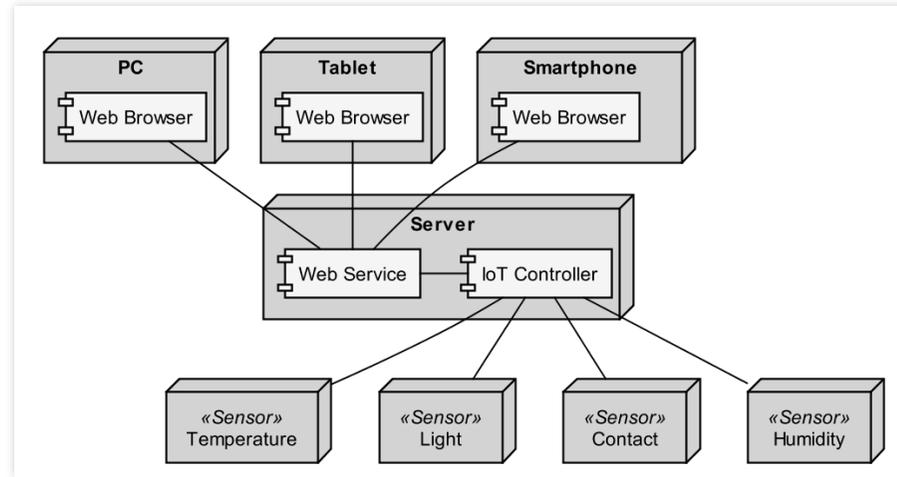
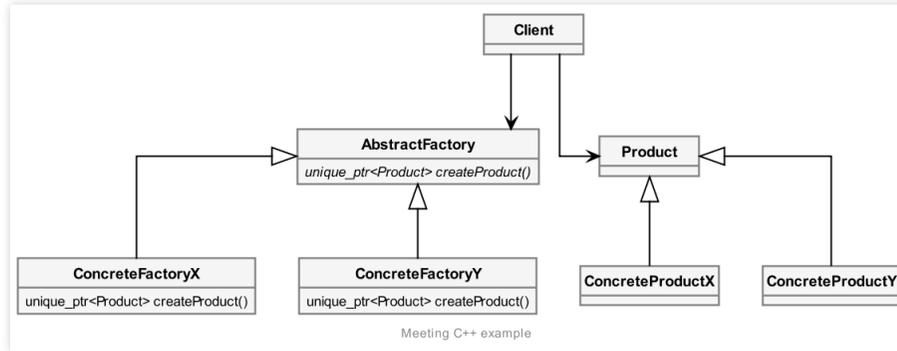
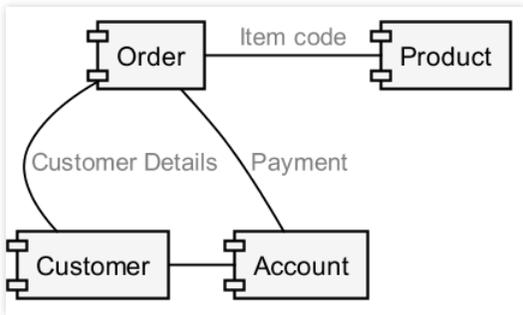
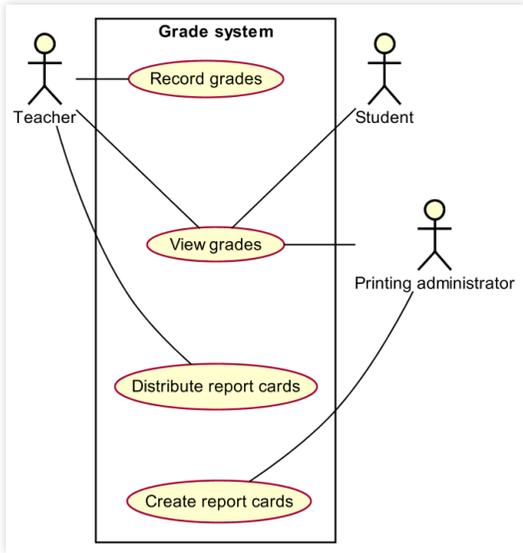
UML Diagrams

PlantUML



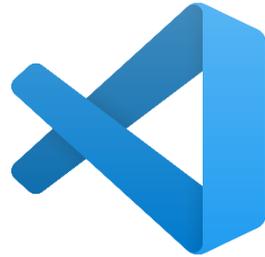
PlantUML

Some Examples



PlantUML

Live Demo in Visual Studio Code



Visual Studio Code

&



PlantUML Extension

➔ PlantUML Live Demo

PlantUML

Doxygen & Markdown Integration

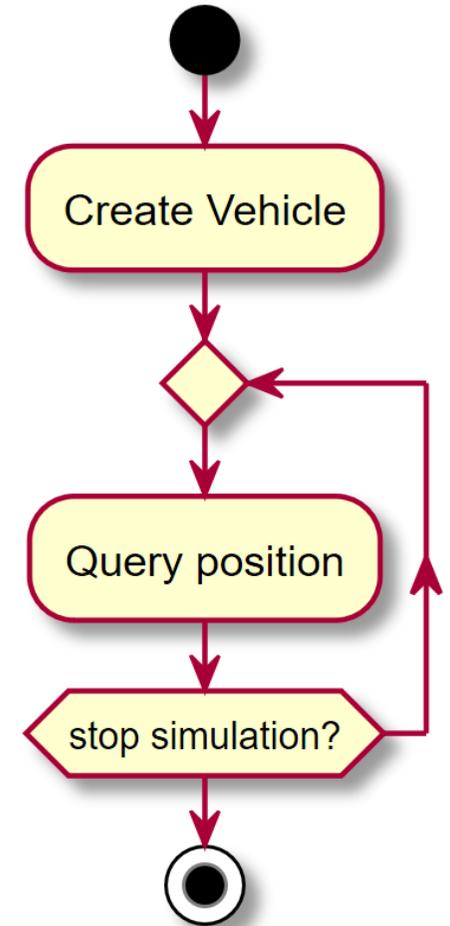
```
# This is a vehicle model
```

This example shows a PlantUML diagram within a Markdown file to be processed by doxygen.

```
@startuml "Vehicle Simulation"
```

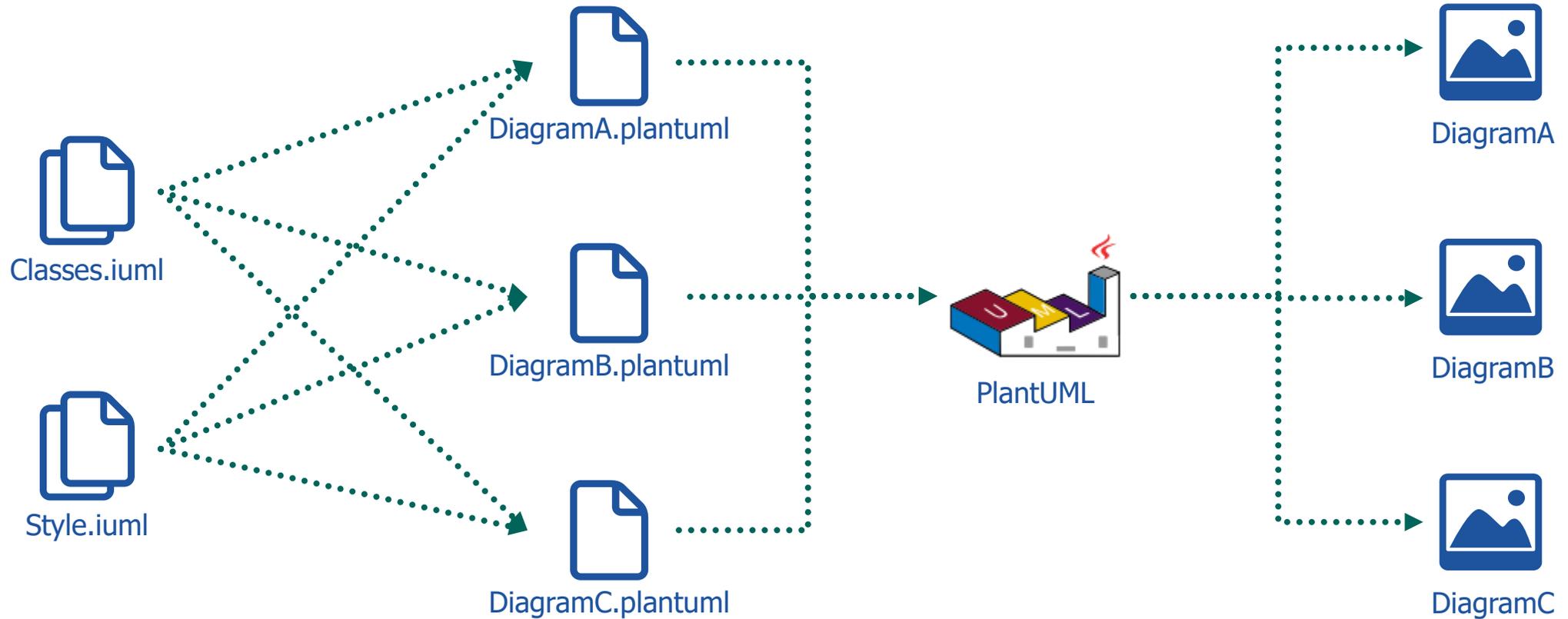
```
start
:Create Vehicle;
repeat
    :Query position;
repeat while (stop simulation?)
stop
```

```
@enduml
```



PlantUML

Consistency



➔ Doxygen & PlantUML consistency demo

Great tooling for drawing
diagrams as code is already available!
and it even integrates nicely with CMake and Doxygen



Takeaways & Outlook

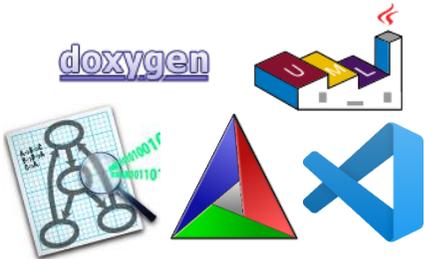
Takeaways & Outlook



Visuals are key



Diagrams as code



Great tooling available



Single source
& version controlled



Fun & easy
to work with

- Doxygen name resolution
- Reference of header files

 [Doxygen link demo](#)

A photograph of a wooden desk with a white sticky note and a black marker. The sticky note has the word 'thanks!' written in black cursive. The marker is a black Tombow ABT Acid Free N15. A red line with a circular graphic element is overlaid on the bottom left of the image.

thanks!

Thank you