



For a Better Tomorrow

AISIN GROUP

Virtual based Automated Testing for automotive Body ECUs

QTronic User Conference 2019
Virtual ECUs and Applications
2nd of December, Berlin, Germany

Hiroshi Ueda

Ubiquitous AI Corporation
SPQA Division

Kazumasa Matoba

AISIN SEIKI Co., Ltd.
Software Engineering Dept.

1. Introduction

2. Background

3. HILS by Silver

4. Python Scripts Management

5. Summary



1. Introduction

2. Background

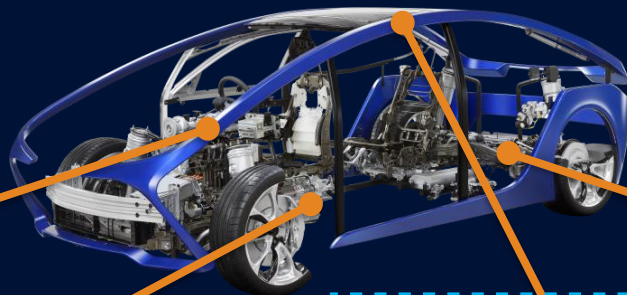
3. HILS by Silver

4. Python Scripts Management

5. Summary



1-1. AISIN Group Main Products



ICT & Electronics

Powertrain

Body

Vehicle Safety System



Sensor



Automatic transmission



Power sliding door



Parking assist system



Car navigation system



CVT



Sunroof



Power tilt & telescopic



ECU



Electric water pump



Power seat



Electric parking brake

1-2. Features of the Body System

Features of the body system

- There is a direct operation request from the user.
- The number of actuators to be driven is relatively small. (1 ~ 3 pieces)
- The requirement for responsiveness is low



1. Introduction

2. Background

3. HILS by Silver

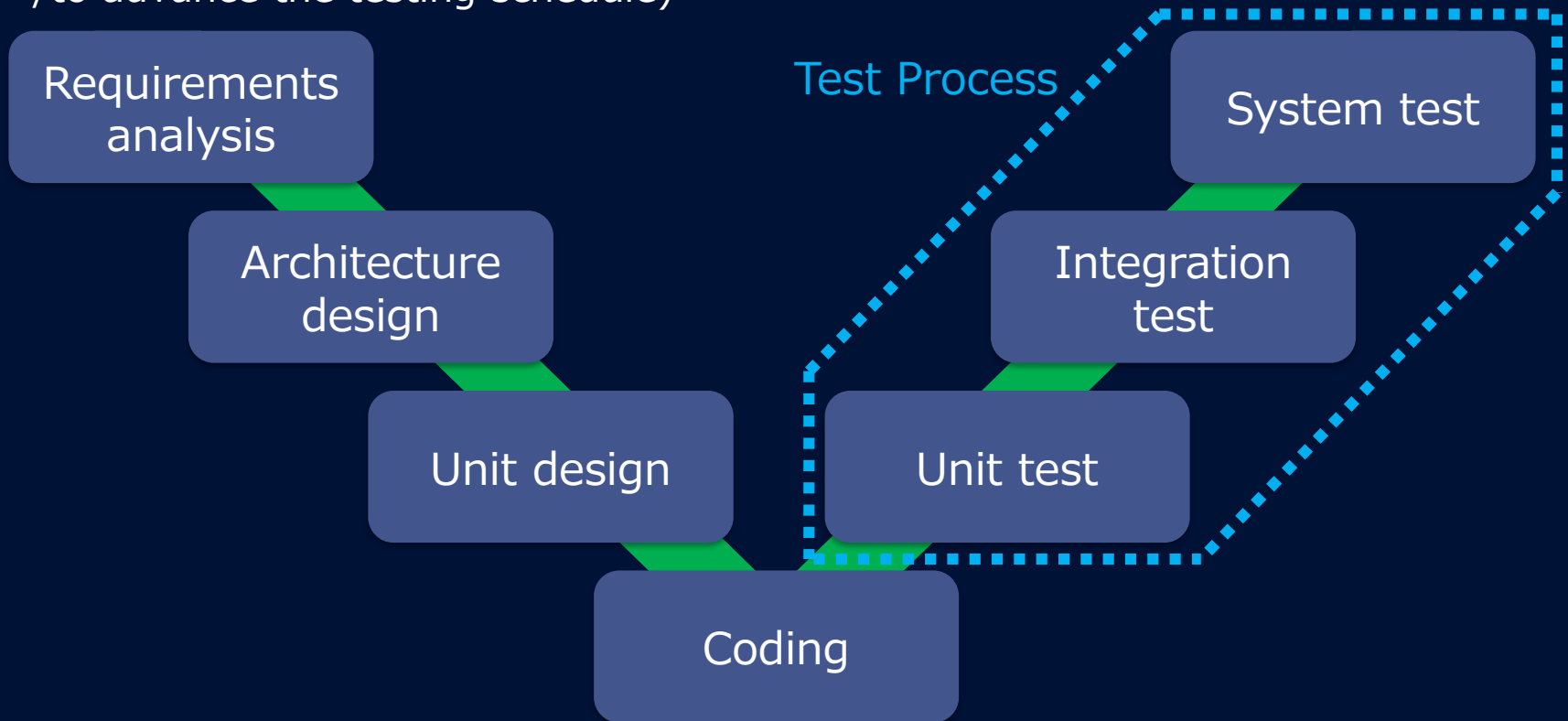
4. Python Scripts Management

5. Summary



2-1. Testing Embedded Software

- Increase in test cases due to the development of larger-scale in-vehicle software
- Increased demand for simulation testing
(Reduced hardware and other provisioning costs
/easy and replicable test environments
/to advance the testing schedule)

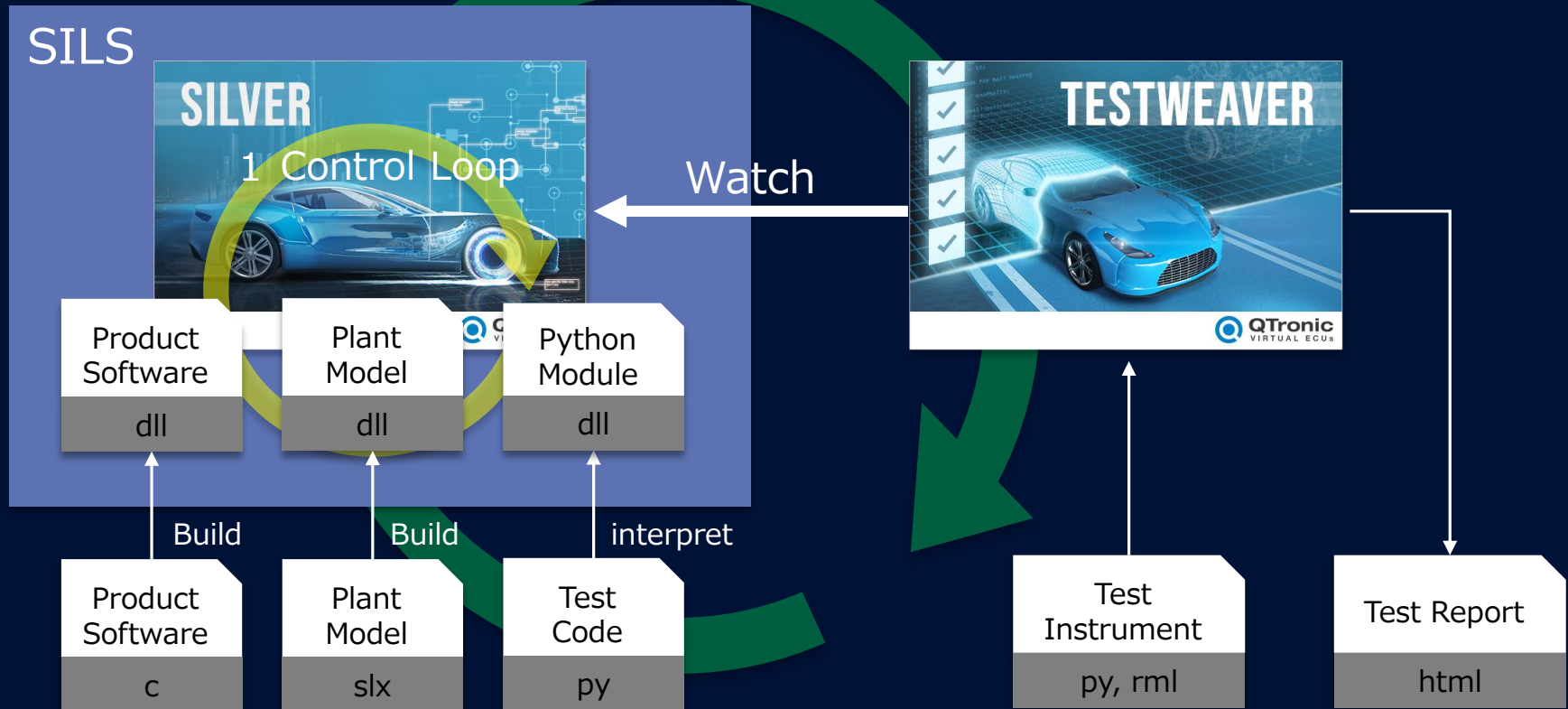


Software Development V-Model



1. Silver was introduced as a tool for simulation of product ECU, and SILS environment was constructed.
2. TestWeaver Light was introduced to automatically generate reports and automatically judge for simulation results.

2 Test Automation



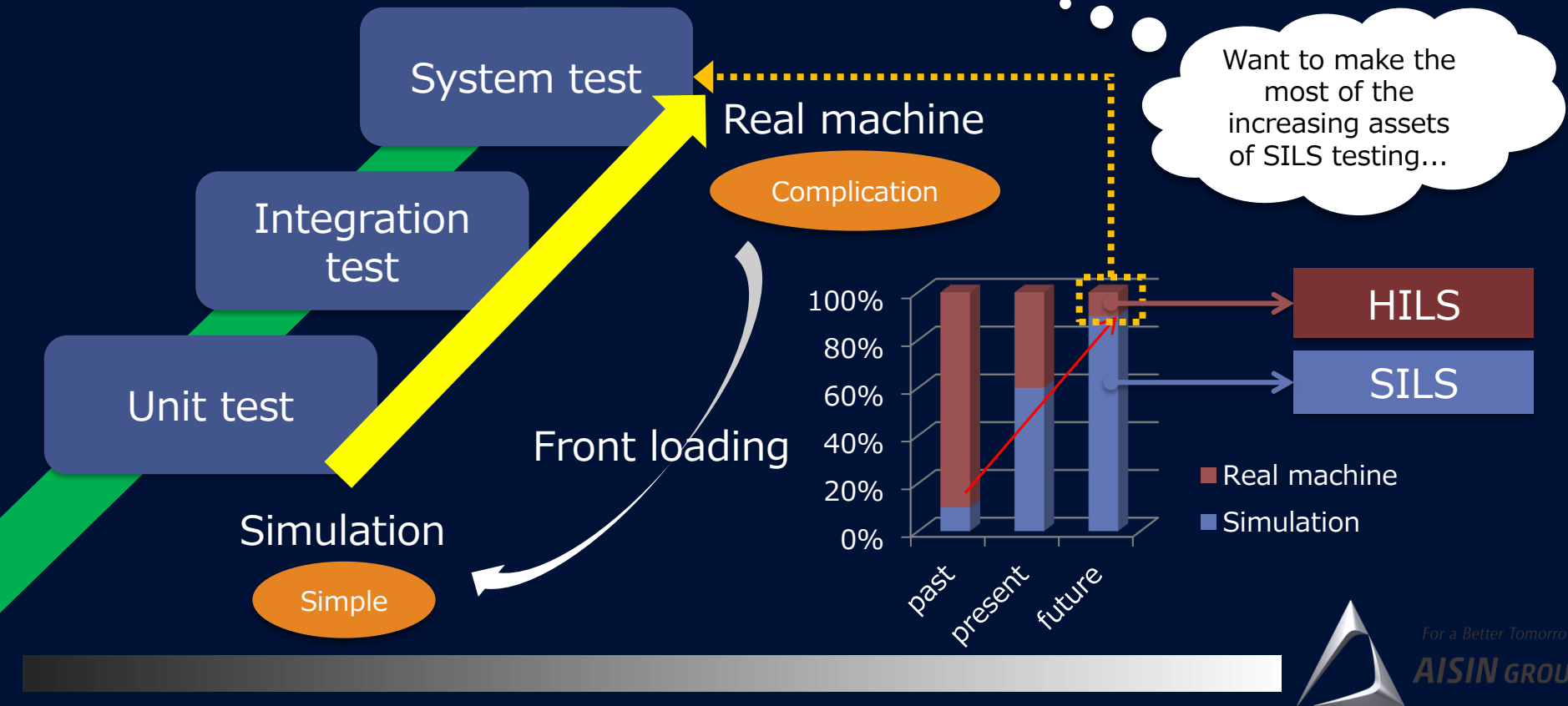
2-3. Issue in ECU Test

- SILS enables efficient testing on PCs
- Increasing the ratio of SILS testing can improve the efficiency of the entire test process.

BUT... In the development of embedded software, we can not eliminate tests requiring verification of actual machines.

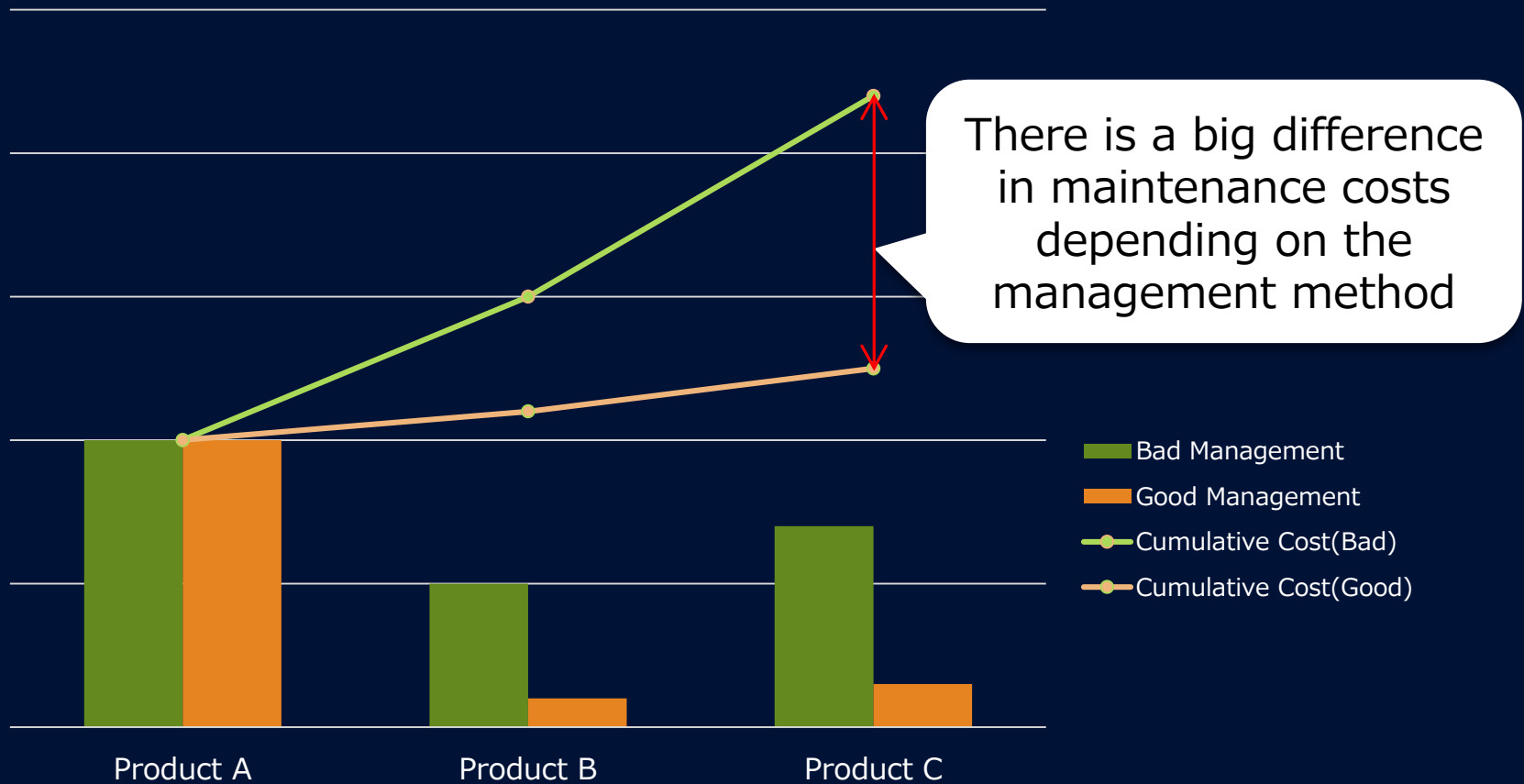
ex.) Interrupt/Timing/Processing Load

-> HILS capable of efficient automatic testing in actual machines is required



Good management practices for test assets are needed to keep testing costs lower.

Management Cost



1. Introduction

2. Background

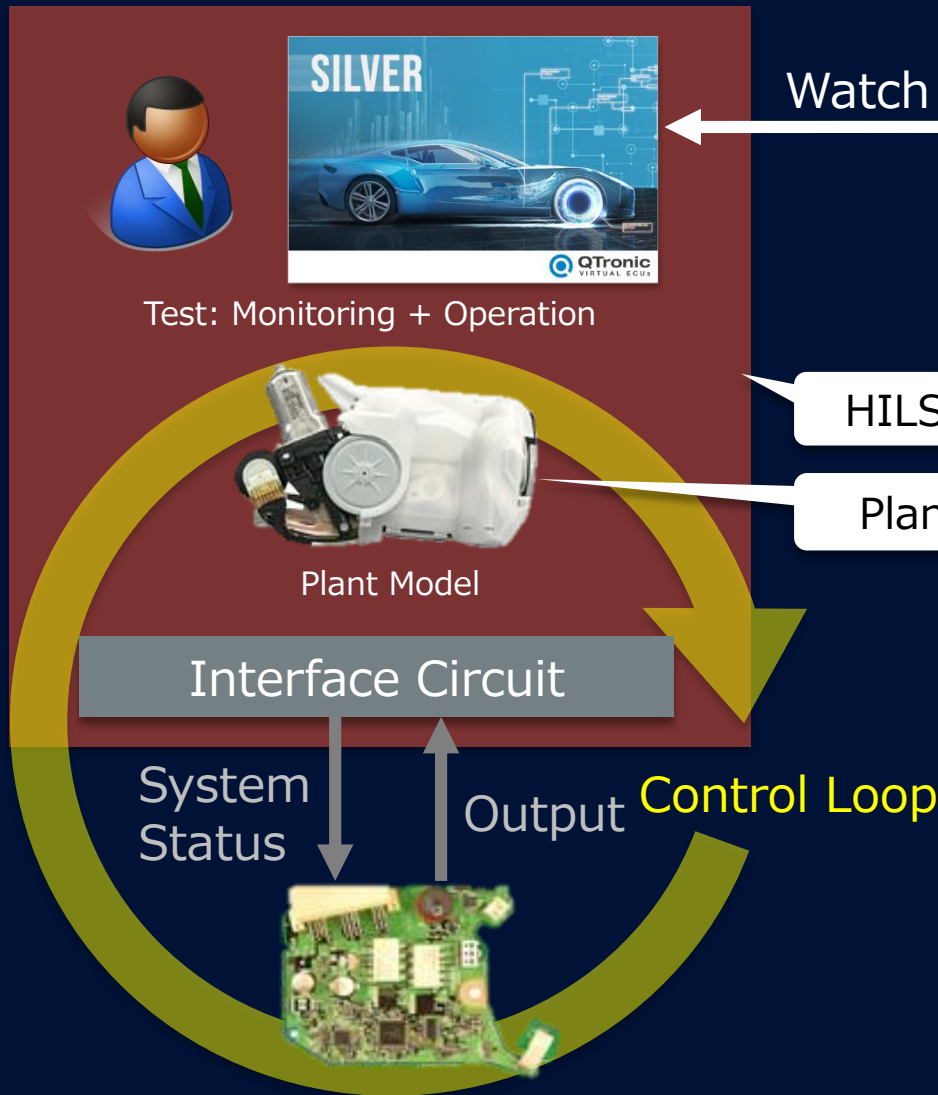
3. HILS by Silver

4. Python Scripts Management

5. Summary



HILS

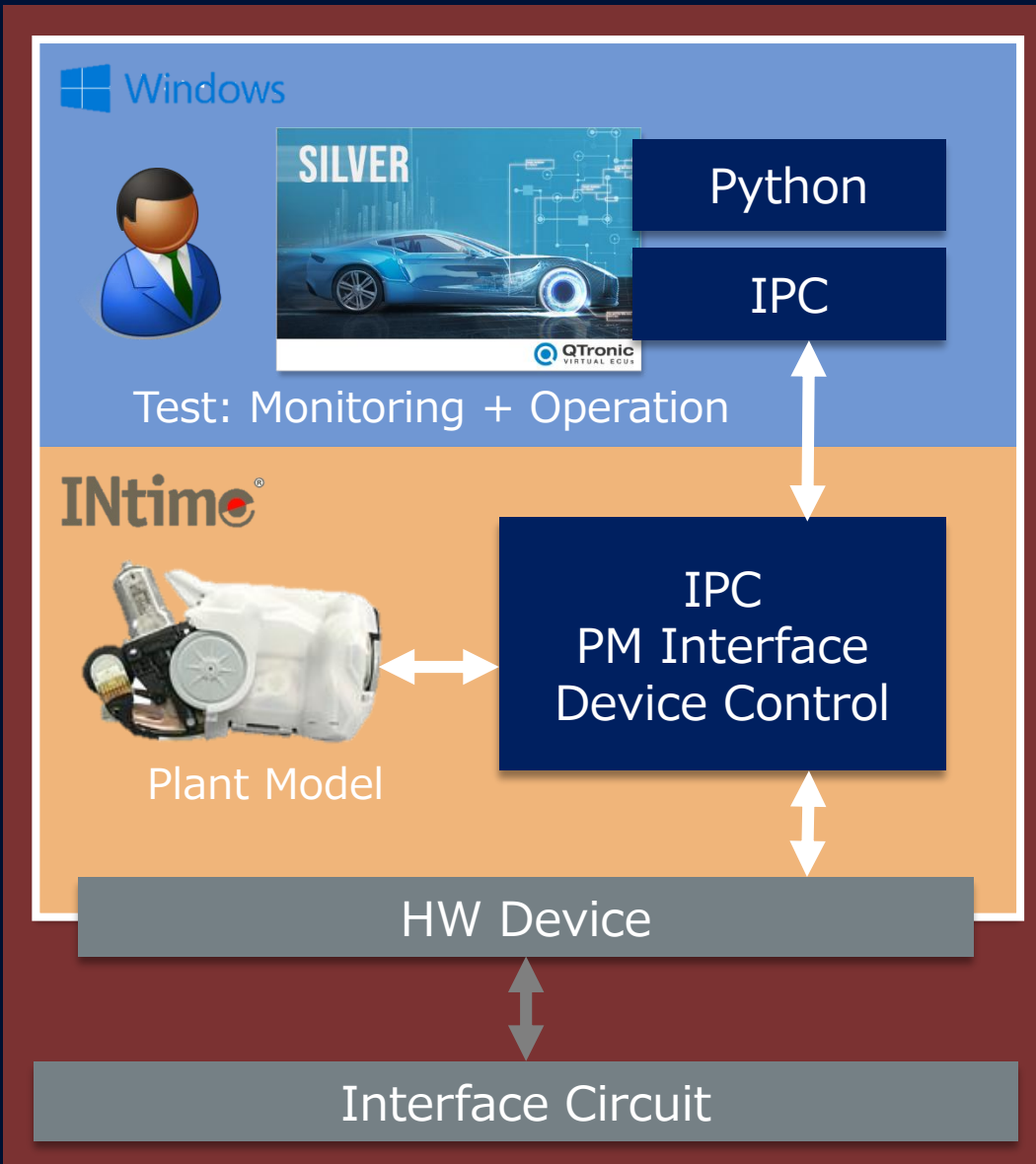


HILS can do automated testing with scripts

Plant models can be easily fail-safe tested

In-house HILS already exists, but there are issues.

- Must use low-level in-house language for test scenario creation.
- Language specifications are not maintained and operated.



Benefits

- Common test interface with SILS
- Easily run Simulink models in real time
- Reduce in-house software components
- Can use Python as a scripting language

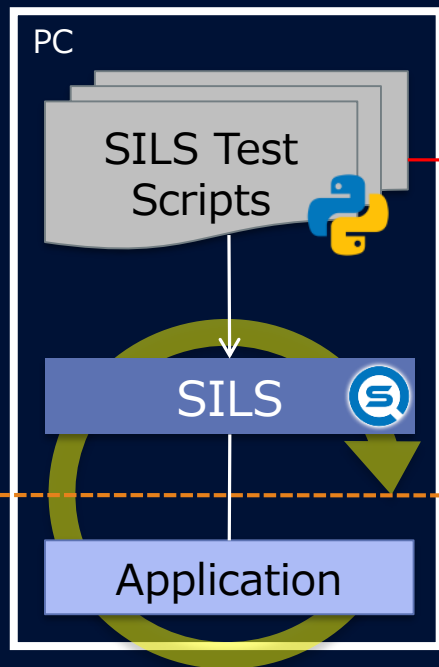
Implementation Function

- Digital In/Out
- Analog Out
- CAN Communication
- LIN Communication

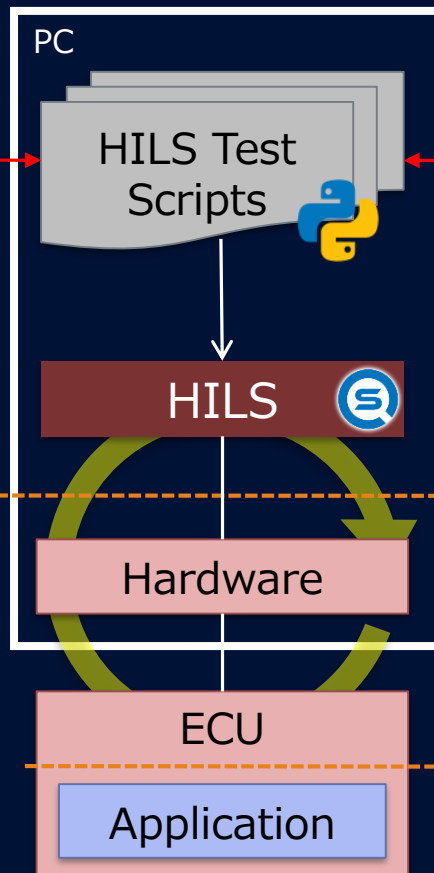
Common Test IF with SILS = Aim to reduce costs through reuse

- Keep the environment (Tools, scripts, and project-related files) used in the SILS tests

Simulation test

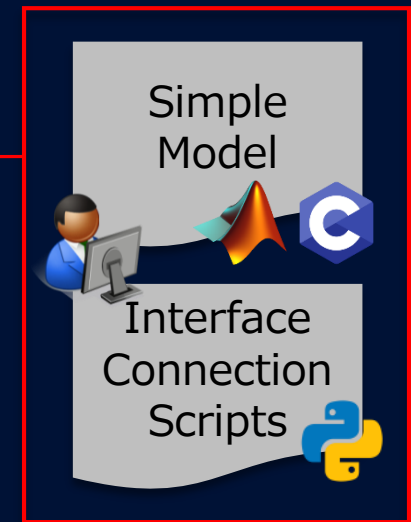


Real machine test



Reuse

Add



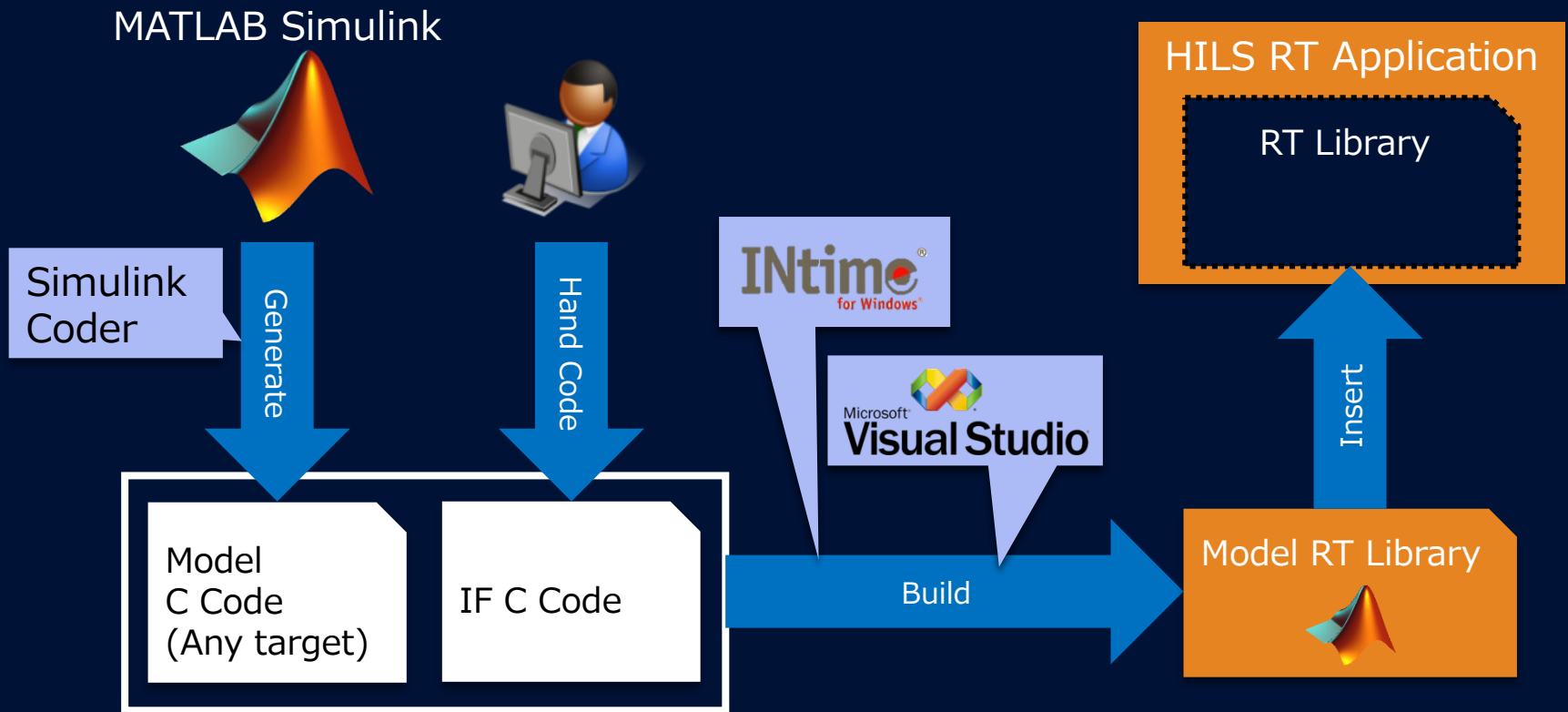
Interface with target

Interface difference

Regulate

3-4. Easily run Simulink models in RT

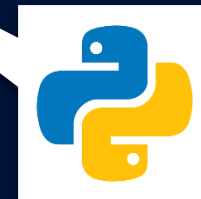
Flexible support for MBD by designing the HILS Plant Model Interface so that it can be imported from Simulink Model. Model development is performed externally using Simulink, and by providing the model, real time operation can be easily performed.



<Reduce in-house software components>

- Cultivate successors through in-house software refactoring.
- Smaller in-house software components

Component	Conventional	Development
1.PC	Outside	Outside
2.Test Scripts	In-house	In-house
3.GUI	In-house	Outside
4.Interpreter	In-house	Outside
5.Plant Model	In-house	Internal
6.Controller	In-house	In-house(new)
7.Device	Outside	Outside(used)
8.I/F Box	In-house	In-house(used)
9.Relay Board	In-house	In-house
10.Target ECU	In-house	In-house



<Can use Python as a scripting language>

- Accelerate script coding by moving from the original low-level to the widely used high-level languages
- Advantages in learning costs, external resources, information, and serviceability (by means of structure)



1. Introduction

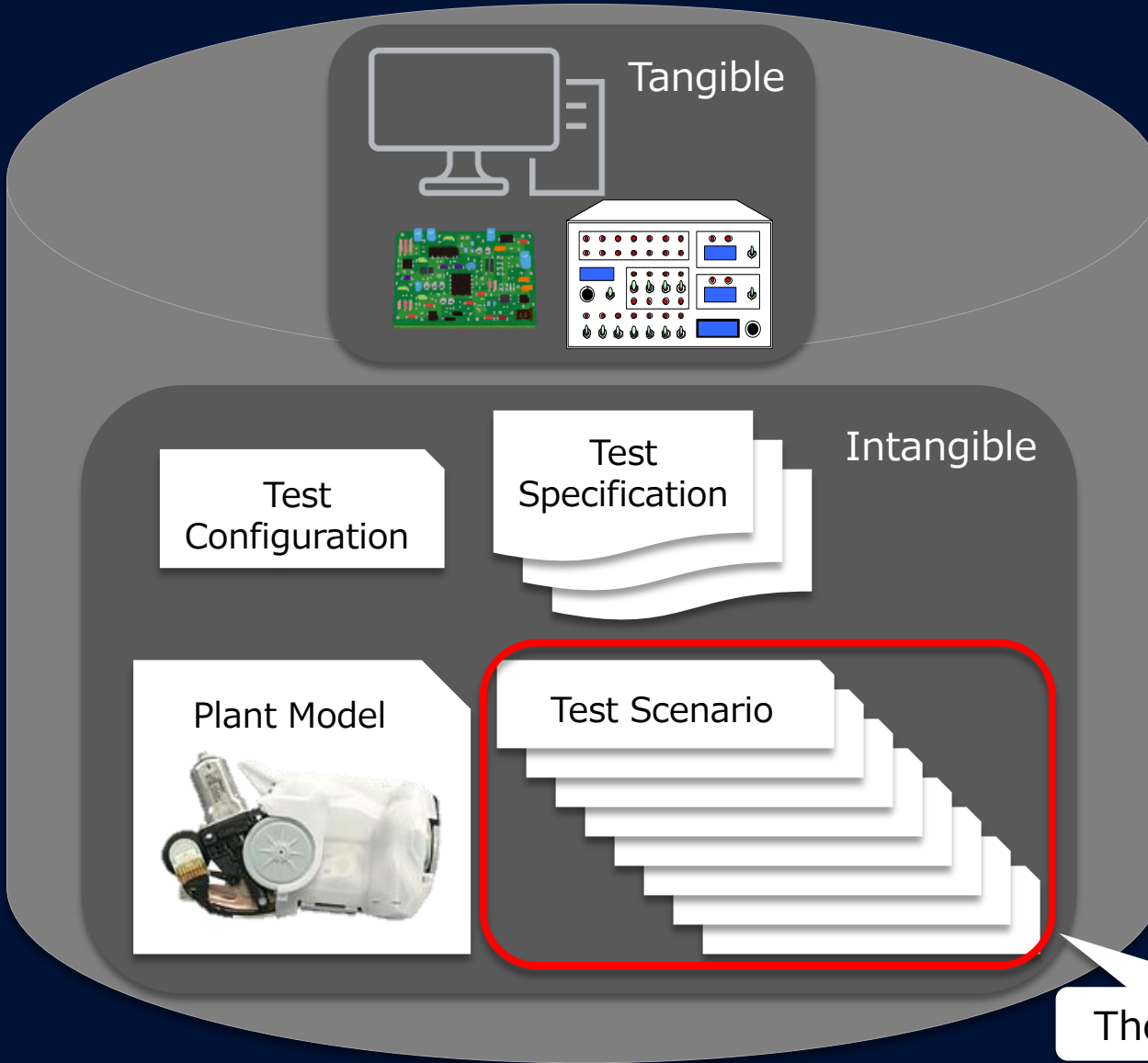
2. Background

3. HILS by Silver

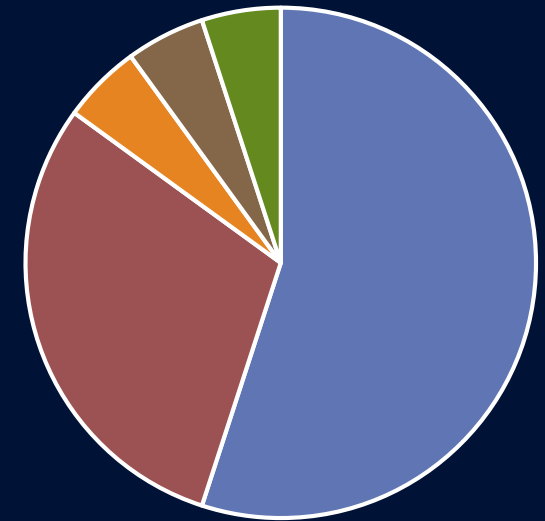
4. Python Scripts Management

5. Summary





Maintenance Cost (intangible)



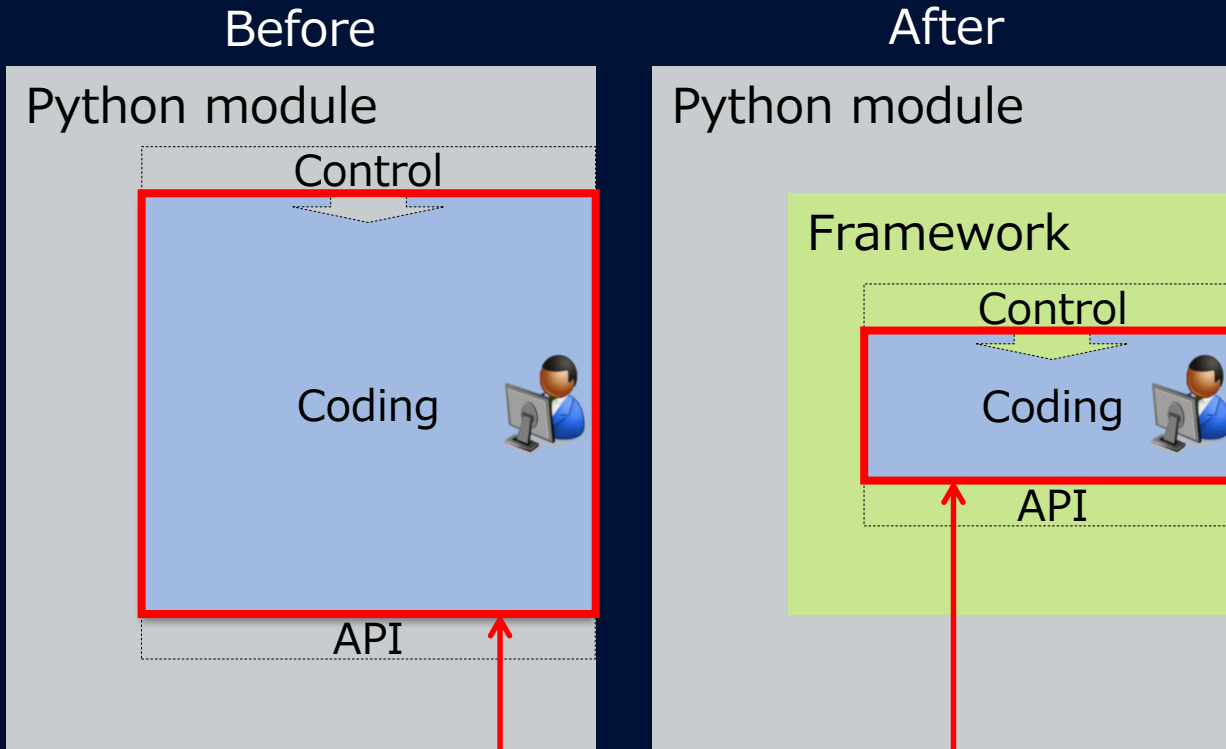
- Test Scenario
- Test Specification
- Test Configuration
- Plant Model
- Other

Thousands more!!

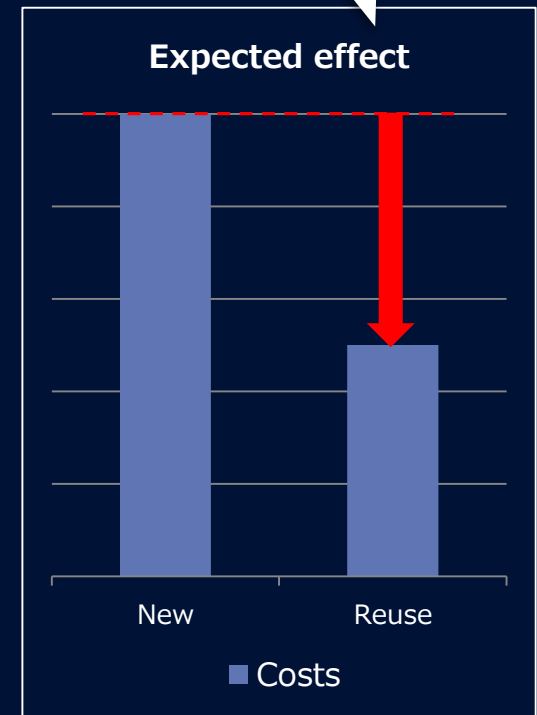
4-2. Test Framework

Advantages of the framework

- Reduce code volume
- Prevent unintended variations
- Fewer bugs



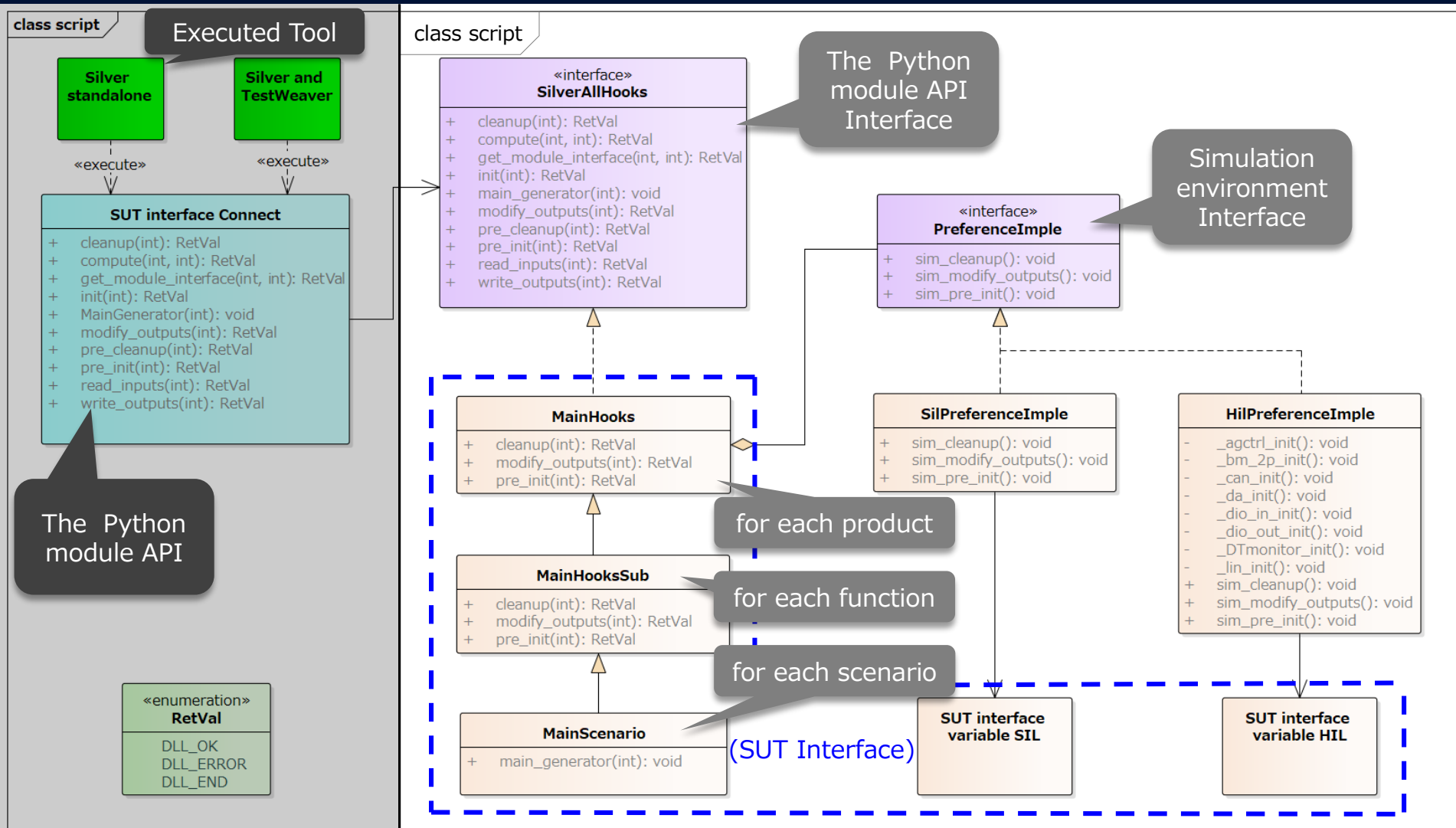
50% more efficient



Maintenance Target

5-1. Test Script Architecture Design

Design Test Script Architecture by Python for



1. Introduction

2. Background

3. HILS by Silver

4. Python Scripts Management

5. Summary



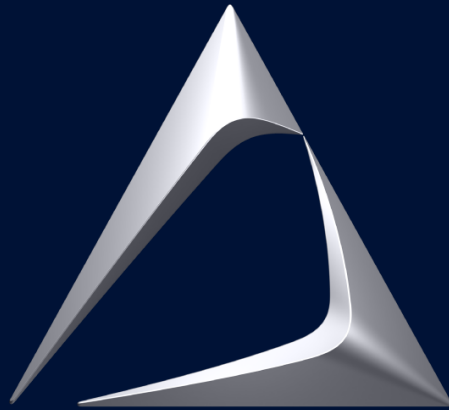
- Develop HILS System by Silver

Tests using real ECU are now possible in terms of test such as timing that cannot be confirmed by SILS

- Framework design for Python module of Silver

Can create and maintain test scenarios with minimal cost, even with a large number of tests





For a Better Tomorrow
AISIN GROUP

Thank you for your attention!!



For a Better Tomorrow
AISIN GROUP