팝콘사 Adaptive AUTOSAR
(R20-11) 개발 플랫폼 소개

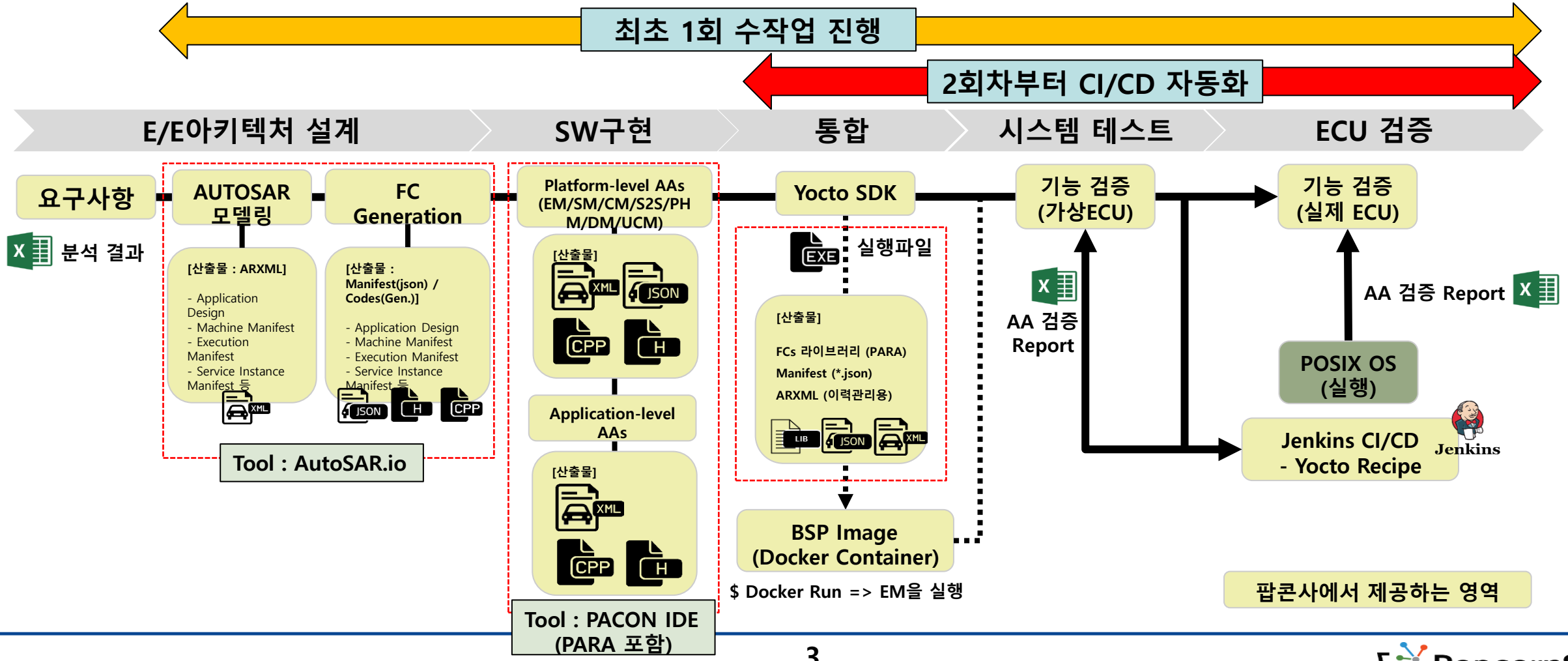
2022.03

PopcornSAR

목차

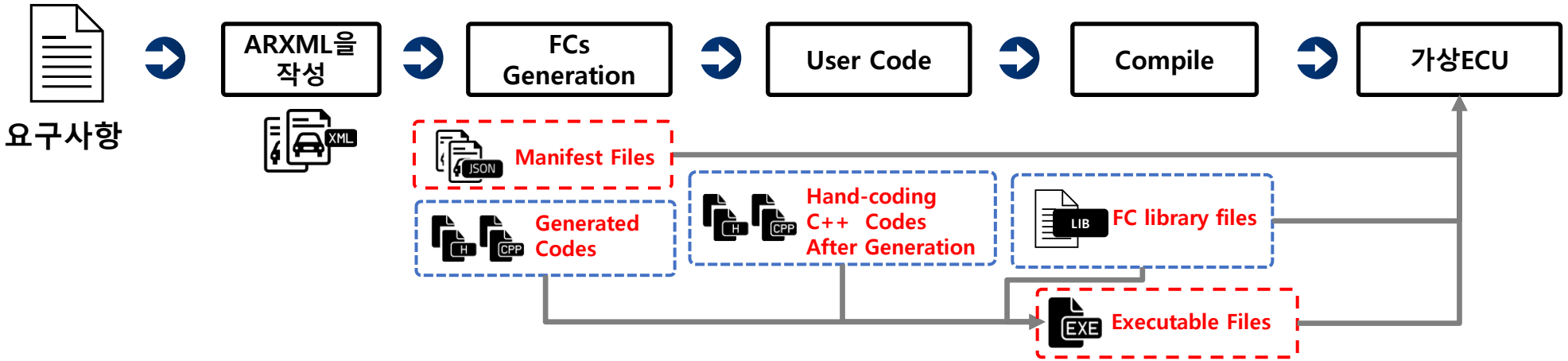
1. Adaptive AUTOSAR (R20-11) 개발 Flow

1. 팝콘사에서는 End-to-End로 Adaptive AUTOSAR Toolchain을 제공
   - OEM/Tier1의 Adaptive Application(AA)/ECU 개발 전체 프로세스를 지원
2. 전체 개발 플로우를 최초 1회 수작업 진행하여 Image를 생성한 이후 CI/CD로 자동화 시스템 구축
3. Yocto SDK는 ECU를 양산한 이후 AA 개발을 지원하기 위해 사용

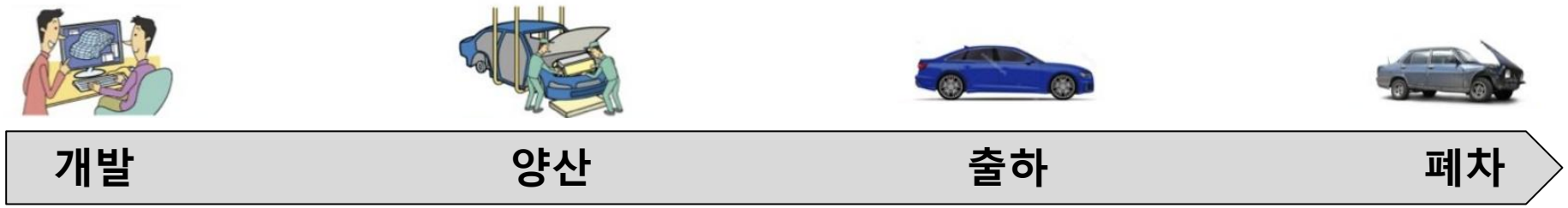


PopcornSAR

1. Adaptive AUTOSAR (R20-11) 개발 Flow



- 가상 ECU에서 테스트 완료 후, 실제 ECU에서 시스템 테스트를 실시함.



AP version (R20-11)

OTA로 AP version을 업그레이드 (R21-11)

- 양산 후에 기능 개선, 사이버 시큐리티를 대응하여 OTA로 업데이트 실시함

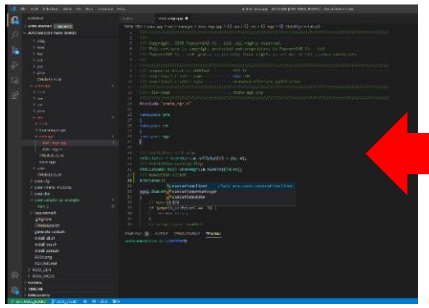
2. 팝콘사 Adaptive AUTOSAR (R20-11) 개발 플랫폼 개요

- PC 인스톨 버전의 Tool과 연동하는 개발 플랫폼
  - ① AP의 FC SDK 또는 Vehicle API의 SDK(Middleware 벤더 의존성 없음)
  - ② 가상 ECU를 제공(시뮬레이션과 테스트), C++ 이외의 개발언어 지원
  - ③ Coding Rule 체크, API 자동완성, 그 외

AutoSAR.io

ARXML 설계 도구
(AP, CP 모두 지원)

vscode



사용자가 직접 설치

개발 서버

PACON IDE (User N)

vscode-server | Vehicle SDK

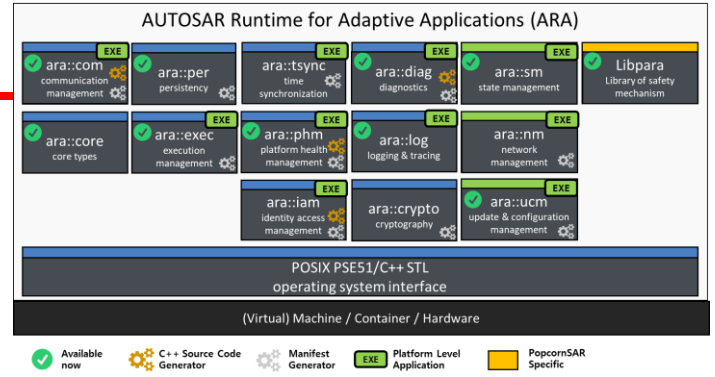
Compiler | other tools

Virtual ECU Docker Container N

System Test Docker Container N

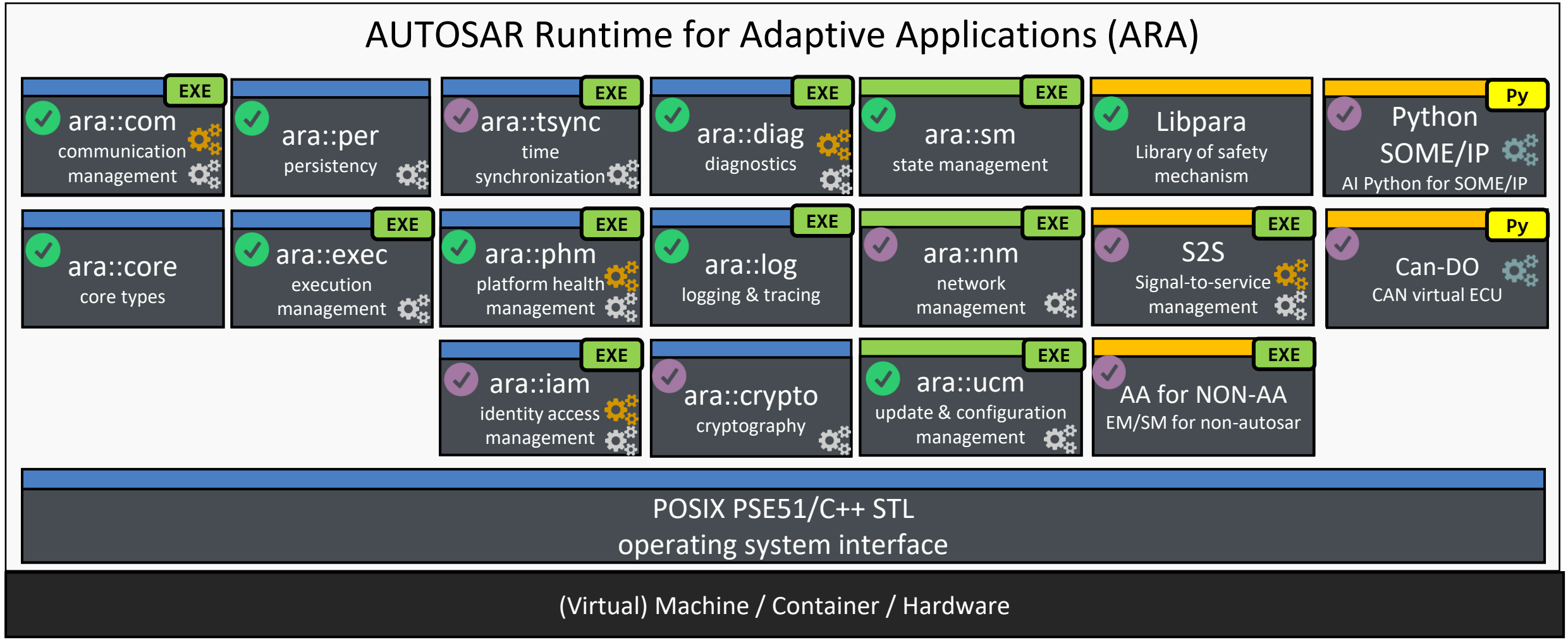
Ubuntu(HOST OS)

PARA



AP용 Functional Cluster와
Platform Level의 AA을 제공

3. 제품 소개 : PARA - FCs list (R20-11) To-be (before 2023)



Available now

Available before 2023

C++ Source Code Generator

Manifest Generator

Specific Generator

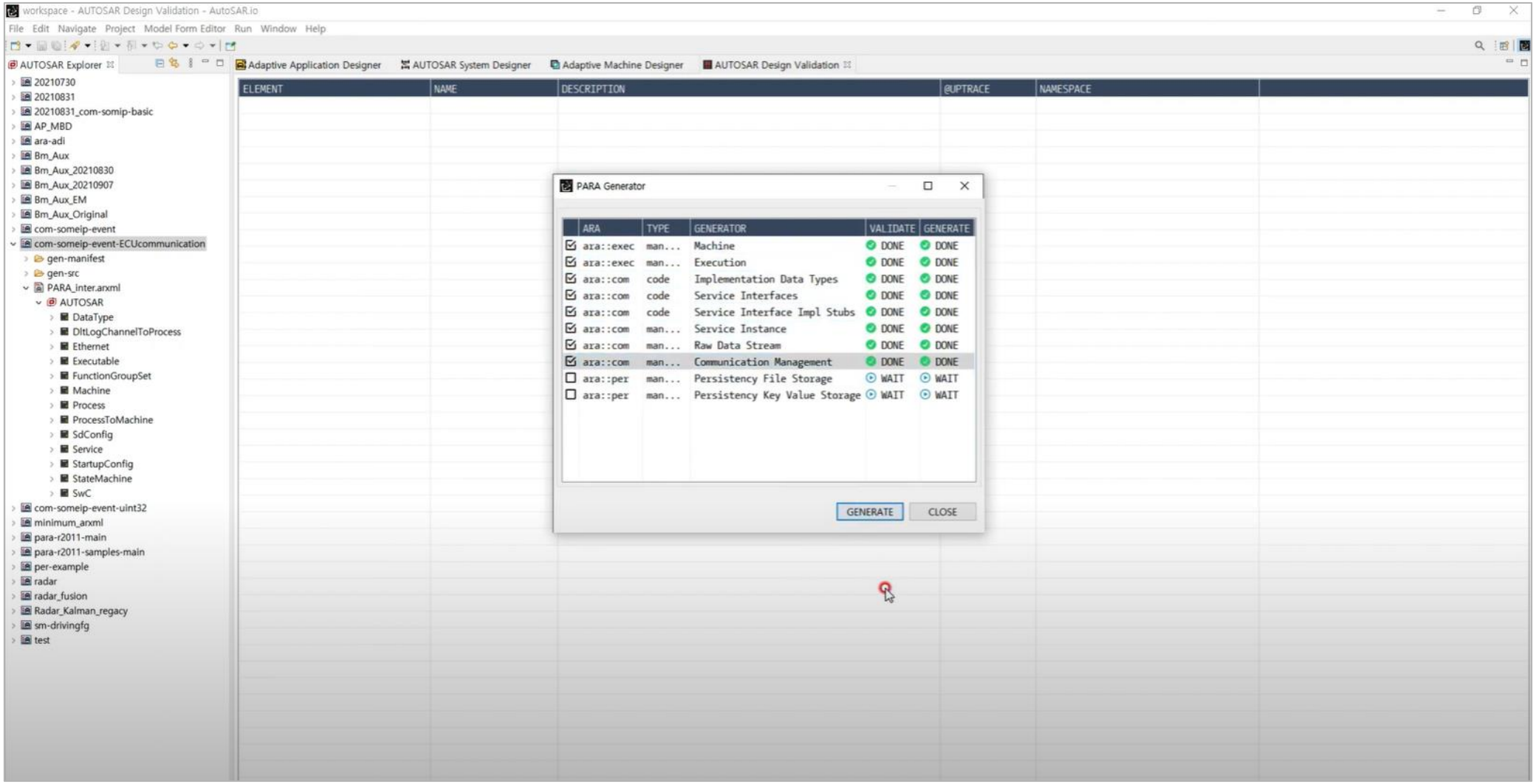
EXE Platform Level Application

PopcornSAR Specific

3. 제품 소개 : PARA - 특징

1. **팝콘사의 PARA는 EM 없이 AA(Adaptive Application) 테스트가 가능**
   - AUTOSAR 표준에서는 EM이 필수이지만, 팝콘사의 PARA는 개발 편의성을 높이기 위하여 EM이 없더라도 AA개발 및 테스트를 수행할 수 있음.
2. **팝콘사의 FC의 Generator는 파라미터 입력 없이, ARXML의 Validation이 OK가 된 경우 자동으로 Generator 실행함.**
   - AutoSAR.io에 FC의 Generator가 포함되어 있기 때문에, 자동으로 AA과 Machine을 해석하여, AA에 관련된 소스코드와 Manifest파일을 자동으로 생성함.
3. **팝콘사는 Matlab/Simulink 2021a(R19 - 11)와 AP MBD, Legacy MBD의 결과물(C++)를 AA로 Migration하는 검증을 마쳤으며, 이에 대한 가이드라인 자료를 제공함.**
   - 고객이 제어용AA를 개발할 때, Matlab/Simulink를 사용하는 것이 필수 사항임.
4. **팝콘사의 PARA는 AI개발을 위한 Python용 SOME/IP 통신을 지원함.**

3. 제품 소개 : PARA - 특징



<AutoSAR.io에서 PARA의 Generation을 평균적으로 1분 이내에 완료>

PopcornSAR

3. 제품 소개 : PARA – ISO 26262를 위한 독자 Safety Mechanism

- Adaptive AUTOSAR의 비표준에 영역으로 팝콘사 독자 기능(libpara)으로 Adaptive Application에서 개발자의 실수로 ara API를 잘 못 사용한 소스코드 구현이나 POSIX OS에서 발생할 있는 Human error를 Adaptive Application에 대한 execution level fault detection 기능 (232개 이상 기능 탑재)

```
2021/11/19 08:08:04.216822 3612010536 101 ECU1 unde DFLT log verbose V 1 [[RCVR] REQ(OFFER) sess:1,sid:201,iid:2,maj:1,min:4294967295,p
ath:/example/var/ara-channel/com/4600_BMS_AA_RootSwc_RequiredPort.service <= /example/var/ara-channel/com/4600_BMS_AA_RootSwc_RequiredP
ort.tmp]
2021/11/19 08:08:04.216950 3612010537 102 ECU1 unde DFLT log error V 1 [CmRouter::ProcessOfferServiceReq:: not exist service fail <key:
pport:201:2:1:4294967295>]
2021/11/19 08:08:04.217106 3612010539 103 ECU1 unde DFLT log verbose V 1 [[SNDR] RES(OFFER) sess:1,res:not_supported => /example/var/ar
a-channel/com/4600_BMS_AA_RootSwc_RequiredPort.tmp]
```

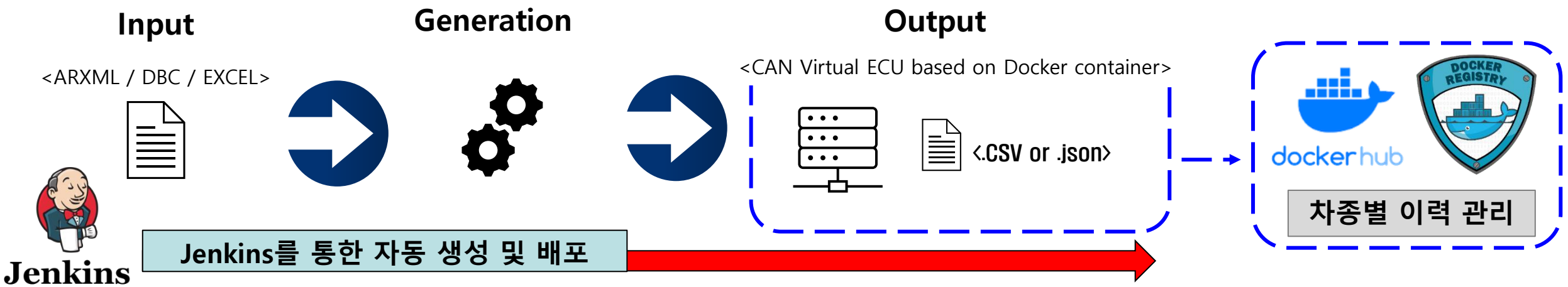
<ex1 ara API의 Human error, AA 실행시 Someip Error fault detection : someip OfferService API>

```
2022/02/09 03:39:18.645278 102307244 001 ECU1 CM-- DFLT log error V 1 [InitByManifest:: ManifestParser::InitByManifest::
HasParserError fail <0> (/home/popcornsar/para-r2011-main/para-api/com/internal/database/database.cpp #63)] 5terminate
called after throwing an instance of 'std::runtime_error' 6 what(): InitByManifest:: ManifestParser::InitByManifest::
HasParserError fail <0> (/home/popcornsar/para-r2011-main/para-api/com/internal/database/database.cpp #63
```

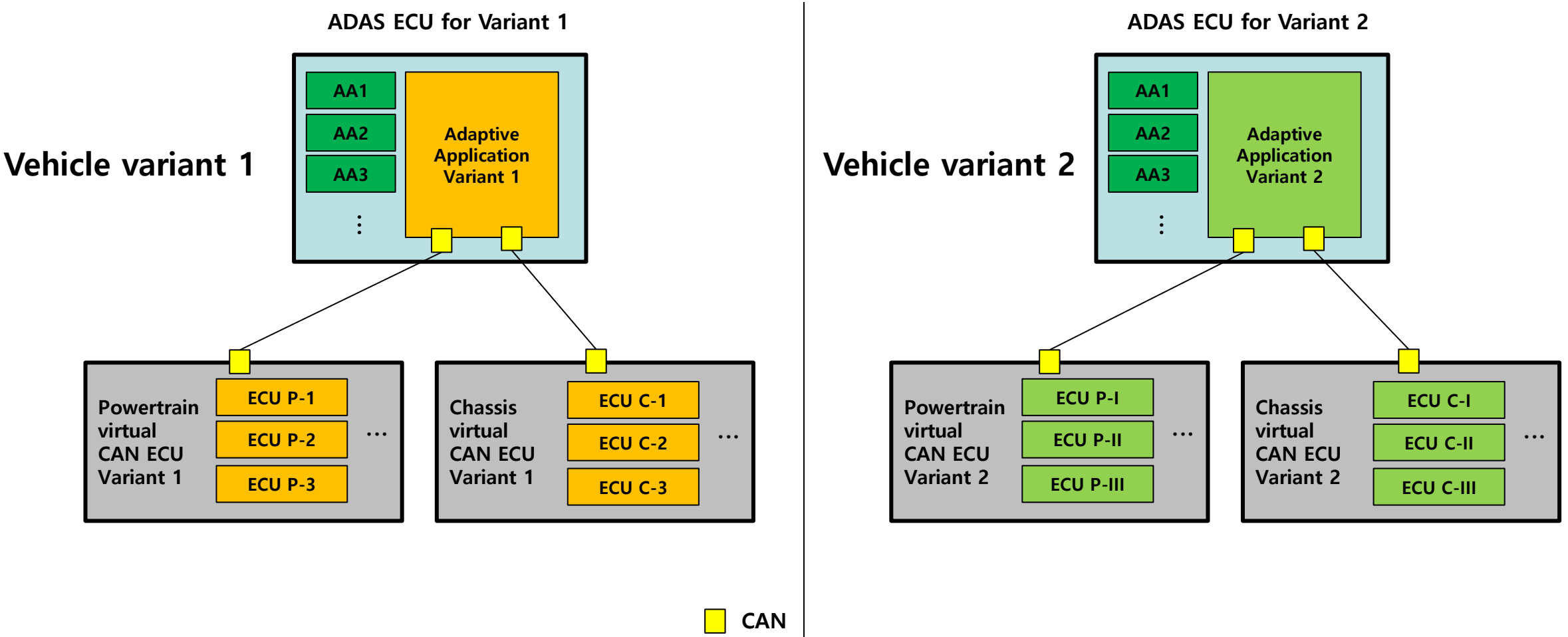
<ex2 POSIX OS의 Human error, AA 실행시 보드에서 POSIX OS의 vi editor사용에 대한 dummy 파일 생성 이슈>

3. 제품 소개 : PARA – CAN-DO(based on Docker container)

- CAN-DO는 팝콘사에서 제공하는 generator로, ARXML/DBC/EXCEL을 입력하면 자동으로 docker container 기반 CAN 가상 ECU를 생성합니다.
- 과도한 수작업 없이, 빠른 시간 안에 여러 개의 CAN 가상 ECU를 자동 생성하여 ECU 테스트를 진행할 수 있습니다.
- Dockerhub를 통해 CAN 가상 ECU 이력관리가 가능하며, Jenkins를 통해 CAN 가상 ECU 생성작업을 자동화할 수 있습니다.
- AUTOSAR 적용 여부/AUTOSAR 버전에 구애 받지 않습니다.

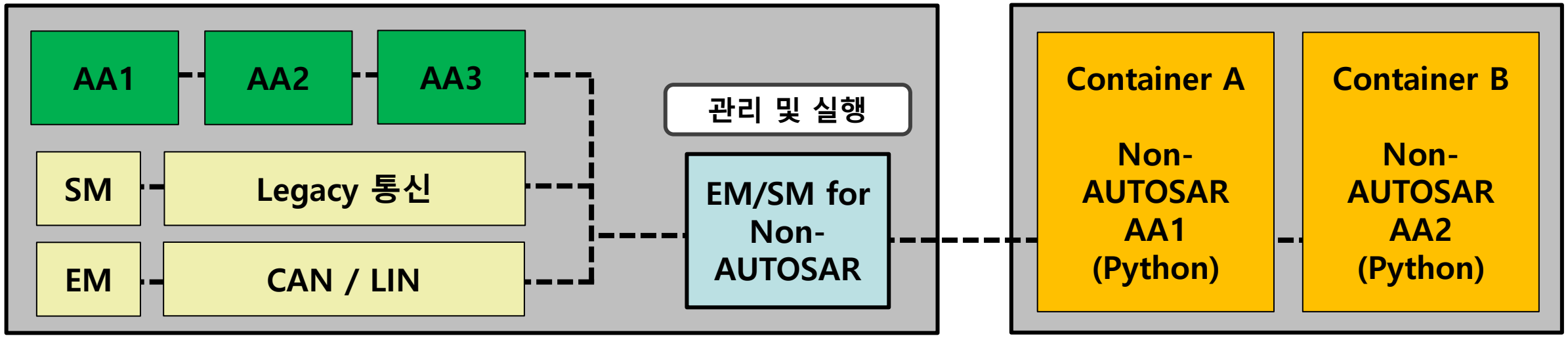


3. 제품 소개 : PARA– Example use case of CAN-DO



- CAN-DO는 Virtual CAN ECU를 자동 생성합니다.
- 각 ECU와 설정 사항은 Dockerhub/Docker registry를 통해 저장/관리될 수 있습니다.

3. 제품 소개 : PARA – Non-AUTOSAR Application 실행을 위한 특수 AA



Host OS AA
(Adaptive AUTOSAR)

Docker Container
(Non-AUTOSAR)

- EM 적용시, Non-AUTOSAR App (docker, intercomm, Python 등)을 인식하지 못하는 경우가 발생함.
- 해결방안으로 EM/SM의 역할을 수행하는 목적인 AA를 생성하여, Non-AUTOSAR App을 실행할 수 있음.

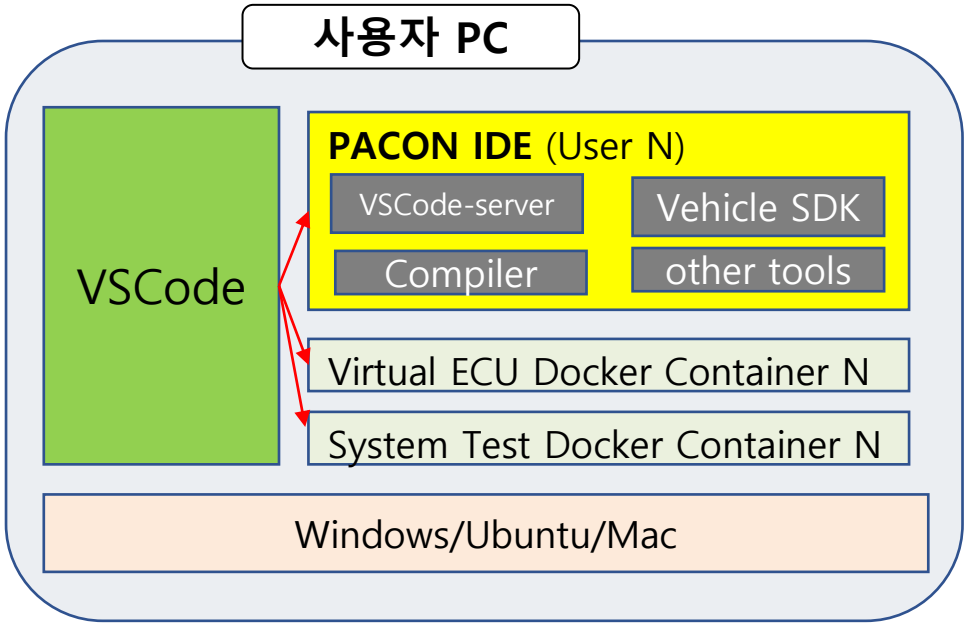
4. 제품 소개 : PACON IDE - 특징

1. **PACON IDE를 고객 브랜드 IDE로 커스터마이징 제공 가능**
   - 대상 차종별로 IDE를 별도 관리하여 사용할 수 있음.
   - Jenkins CI/CD를 이용한 고객 IDE 재배포 및 버전 관리 제공
2. **PACON IDE는 Adaptive Platform Vendor에 대한 의존성이 없음.**
3. **VSCcode와 연동하여 PACON IDE(=Docker Container)를 사용할 수 있음.**
   - Docker Container에는 PARA와 VSCode에서 사용하는 필수적인 Extension이 내장
   - 외부 네트워크 연결 없이 사용 가능함.
4. **사용자의 PC에 Wireshark가 없더라도 PACON IDE내에서 Wireshark를 제공함.**
5. **Jenkins CI/CD에서 PACON IDE를 배포할 수 있음.**
6. **개발 편의성을 높이는 부가 기능 제공**
   - ARA API와 오픈소스 API의 자동완성 기능을 제공함.
   - 실시간으로 Code Rule을 체크할 수 있음.
   - Debug 및 다른 개발 언어(Python 등)을 지원함.

4. 제품 소개 : PACON IDE - 제공 형태

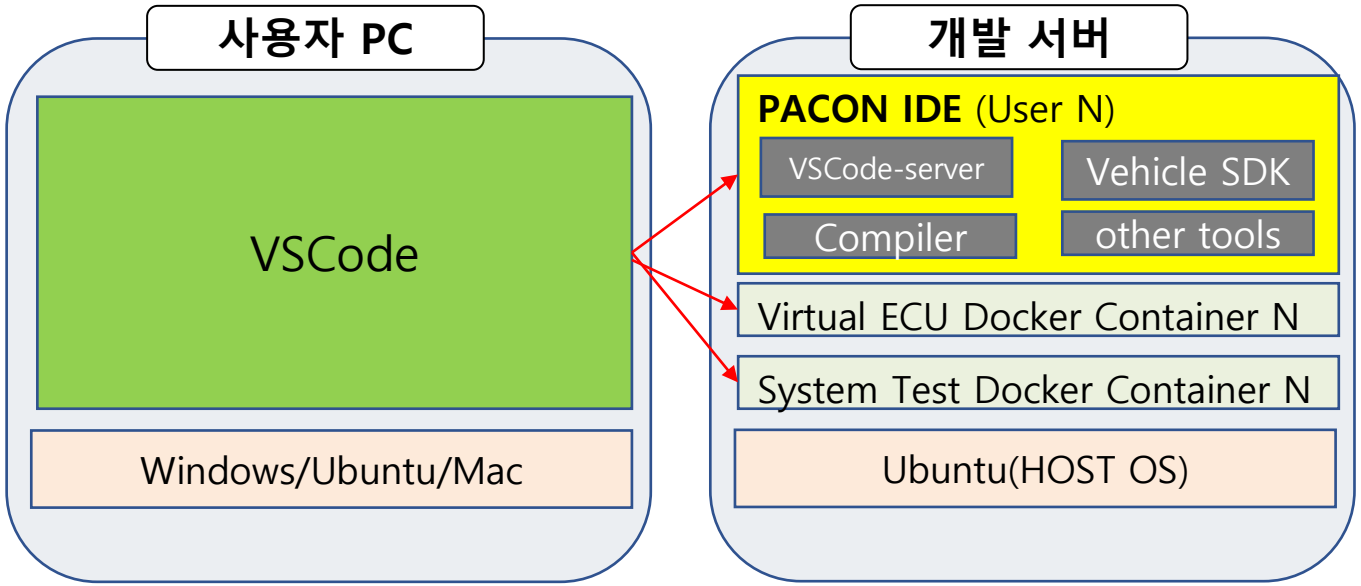
① 사용자 PC에서 개발하는 형태

(네트워크 없이 개발 가능)



② 사용자 PC와 개발서버를 연동하여 개발하는 형태

(사용자 간 원격 협업 시 편의성 및 개발효율성 증대)



docker container : dockerhub에서 배포함

4. 제품 소개 : PACON IDE – 고객 브랜드 IDE로 커스터마이징 제공

차종	CPU	OS	
Variant 1	NVIDIA	Linux (5.0)	→ **Variant1:1.0.0**
	NVIDIA	Linux (5.15)	→ **Variant1:1.1.0**
Variant 2	Intel	QNX	→ **Variant2_intel:1.0.0**
	NXP	QNX	→ **Variant2_nxp:1.0.0**
Variant 3	R-car	Android	→ **Variant3:1.0.0**

- 대상 차종별로 IDE를 별도 관리하여 사용할 수 있음.
- Jenkins CI/CD를 이용한 고객 IDE 재배포 및 버전 관리 제공
- OS 업데이트에 따라 IDE의 재배포 필요

Jenkins

Jenkins를 통한 자동 배포

4. 제품 소개 : PACON IDE - IDE 및 가상ECU 구성 프로세스

고객의 Target ECU 정보

CPU 종류
(NXP, RENESAS, NVIDIA 등)

사용할 POSIX OS
(Linux, QNX 등)

사용할 컴파일러

사용할 AP의 FC

Jenkins

Target ECU용 AP 개발환경
(=PACON IDE)
PACON IDE (User N)

vscode-server | Vehicle SDK

Compiler | other tools

Target ECU의 가상ECU

Virtual ECU Docker Container N

시스템 테스트용 Container

System Test Docker Container N

Jenkins

이력 관리

5. 제품 소개 : 가상ECU - 특징

1. **가상ECU를 고객 브랜드 가상ECU로 커스터마이징 제공 가능**
   - 대상 차종별로 가상ECU를 별도 관리하여 사용할 수 있음.
   - Jenkins CI/CD를 이용한 고객 가상ECU 재배포 및 버전 관리 제공
2. **가상ECU는 Adaptive Platform Vendor에 대한 의존성이 없음.**
3. **타사는 QEMU로 대응하고 있음. (무겁고 성능이 떨어지며, 호환성이 없음)**
   - 팝콘사는 고객이 사용하는 POSIX OS와 실제 TargetECU를 Docker Container로 제공함.
   - Docker Container에서 AA의 테스트가 끝나면 그대로 AA를 TargetECU로 납품함.
4. **다수의 ECU를 Docker Container로 개발하고, DockerHub 또는 Docker registry로 이력관리가 가능하기 때문에, 다른 개발자와 공유하여 작업할 수 있음.**
5. **Jenkins CI/CD에서 다수의 가상ECU를 자동으로 생성하여 AA 테스트가 가능함.**
6. **VSCode와 연동하는 PACON IDE를 이용하여 간단하게 가상ECU를 생성할 수 있음.**
7. **가상ECU와 연동하는 다른 시스템 테스트 전용 Docker Container를 제공함.**
   - 가상ECU 내부에는 시스템 테스트 SW를 넣는 것이 어려움.
     예 : tshark의 가상ECU 내부(ARM)에 넣는 것
   - 테스트 SW(python등)은 다른 시스템 테스트 전용 Docker Container의 내부에 넣음.

5. 제품 소개 : 가상ECU – 고객 브랜드 가상ECU로 커스터마이징 제공

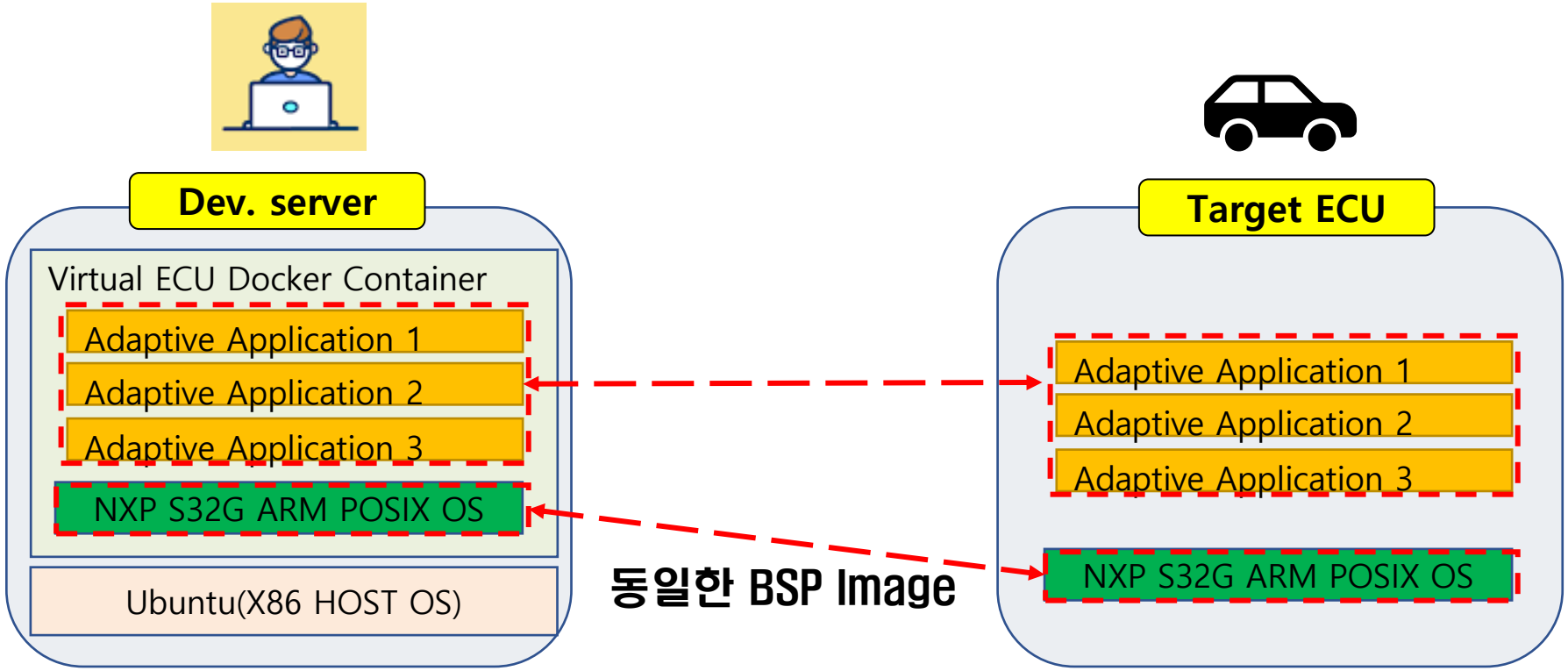
차종	CPU	OS	
Variant 1	NVIDIA	Linux (5.0)	→ vECU_Variant1:1.0.0
	NVIDIA	Linux (5.15)	→ vECU_Variant1:1.1.0
Variant 2	Intel	QNX	→ vECU_Variant2_intel:1.0.0
	NXP	QNX	→ vECU_Variant2_nxp:1.0.0
Variant 3	R-car	Android	→ vECU_Variant3:1.0.0

- 대상 차종별로 가상ECU를 별도 관리하여 사용할 수 있음.
- Jenkins CI/CD를 이용한 고객 가상ECU 재배포 및 버전 관리 제공
- 가상 제어기는 자사/타사 ECU도 고려해야 함

Jenkins

Jenkins를 통한
자동 배포

5. 제품 소개 : 가상ECU – 고객 브랜드 가상ECU로 커스터마이징 제공



단, QNX의 경우는 가상제어기는 QEMU로 실행 가능하며 docker container 사용하지 않음.

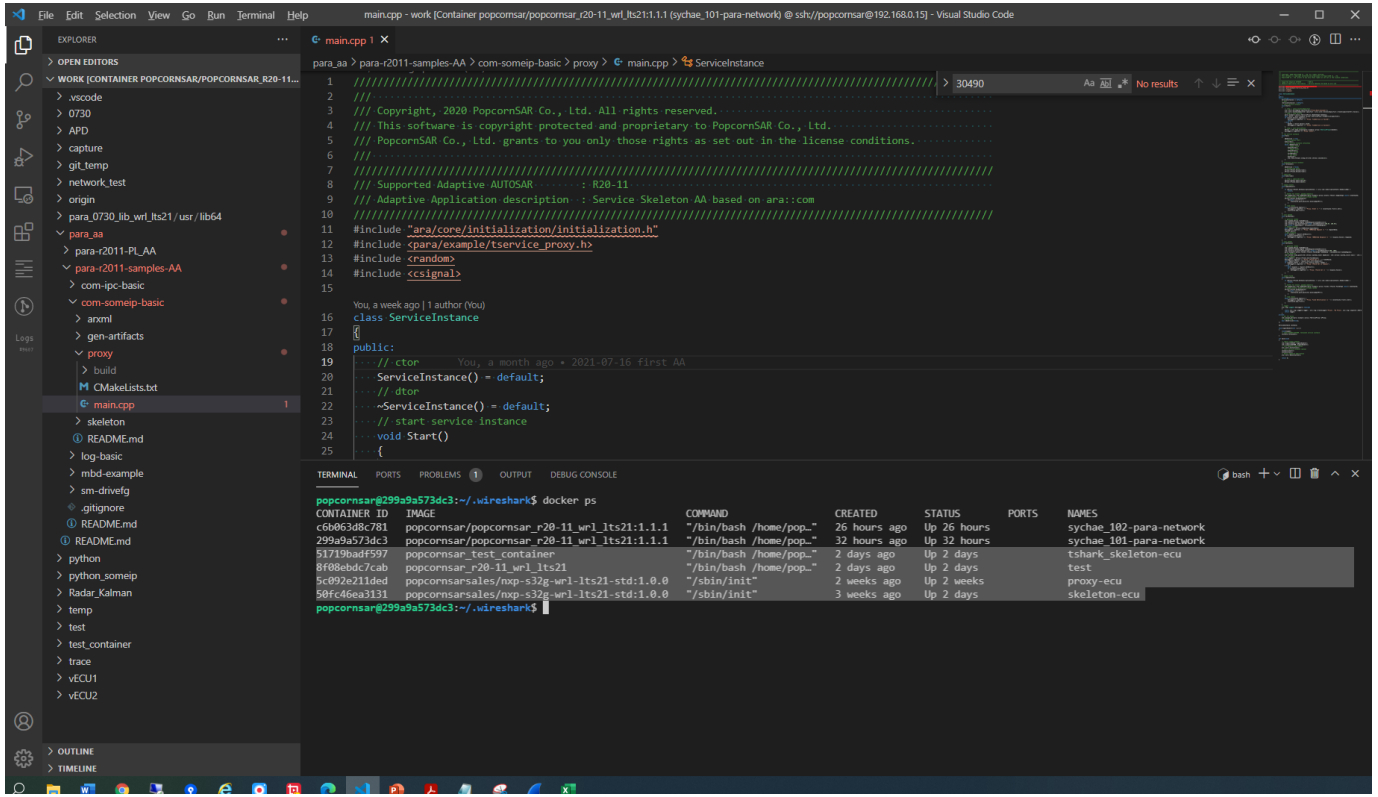
- Docker Container에서 AA의 테스트가 끝나면 재컴파일 필요 없이 그대로 AA를 Target ECU로 납품함.
- 재컴파일에 소모되는 시간이 없으므로, 개발 시간을 비약적으로 단축시킬 수 있음.

5. 제품 소개 : 가상ECU - QEMU vs Docker Container

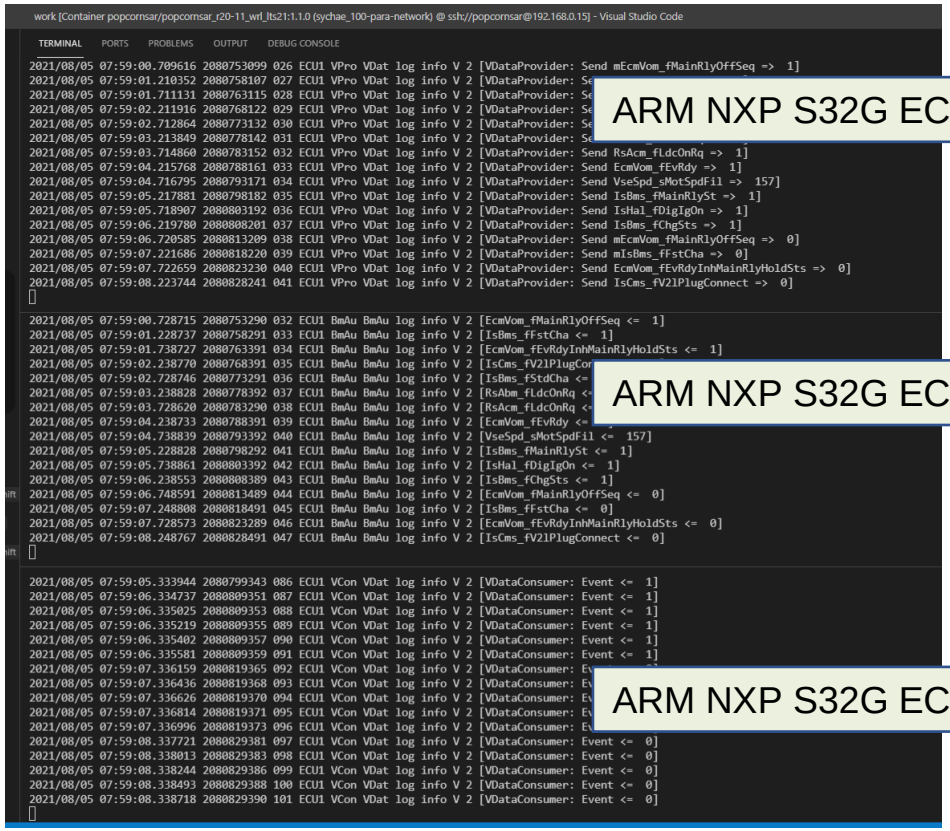
	QEMU (AS-IS)	DockerContainer(To-be)
Target board의 AA 실시간성 보장	X (ROM/RAM 사용량 많음)	O
가상ECU 이력 관리	X	O
타 개발자와의 공유	X	O
네트워크를 통한 배포	X	O
가상ECU에 AA 추가 후 배포	X	O
기타	-	System test용 docker container와 연동 가능

5. 제품 소개 : 가상ECU - 사용 예시

- 다수의 가상ECU를 생성하여 가상ECU간 통신 테스트 수행할 수 있음.



<PACON IDE 내의 가상ECU>

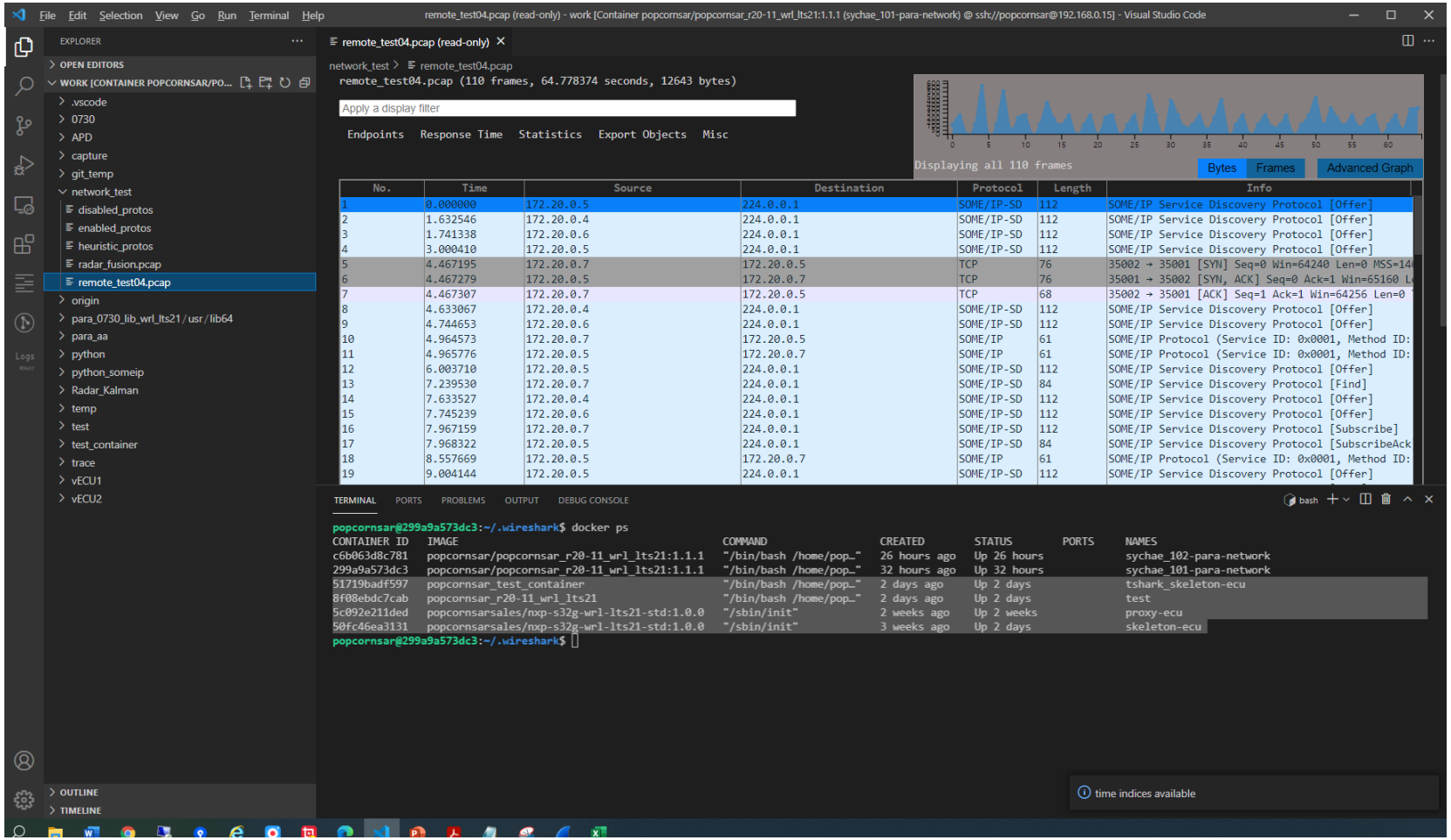


<PACON IDE 내의 Terminal>

PopcornSAR

5. 제품3: 가상 ECU - 사용 예시

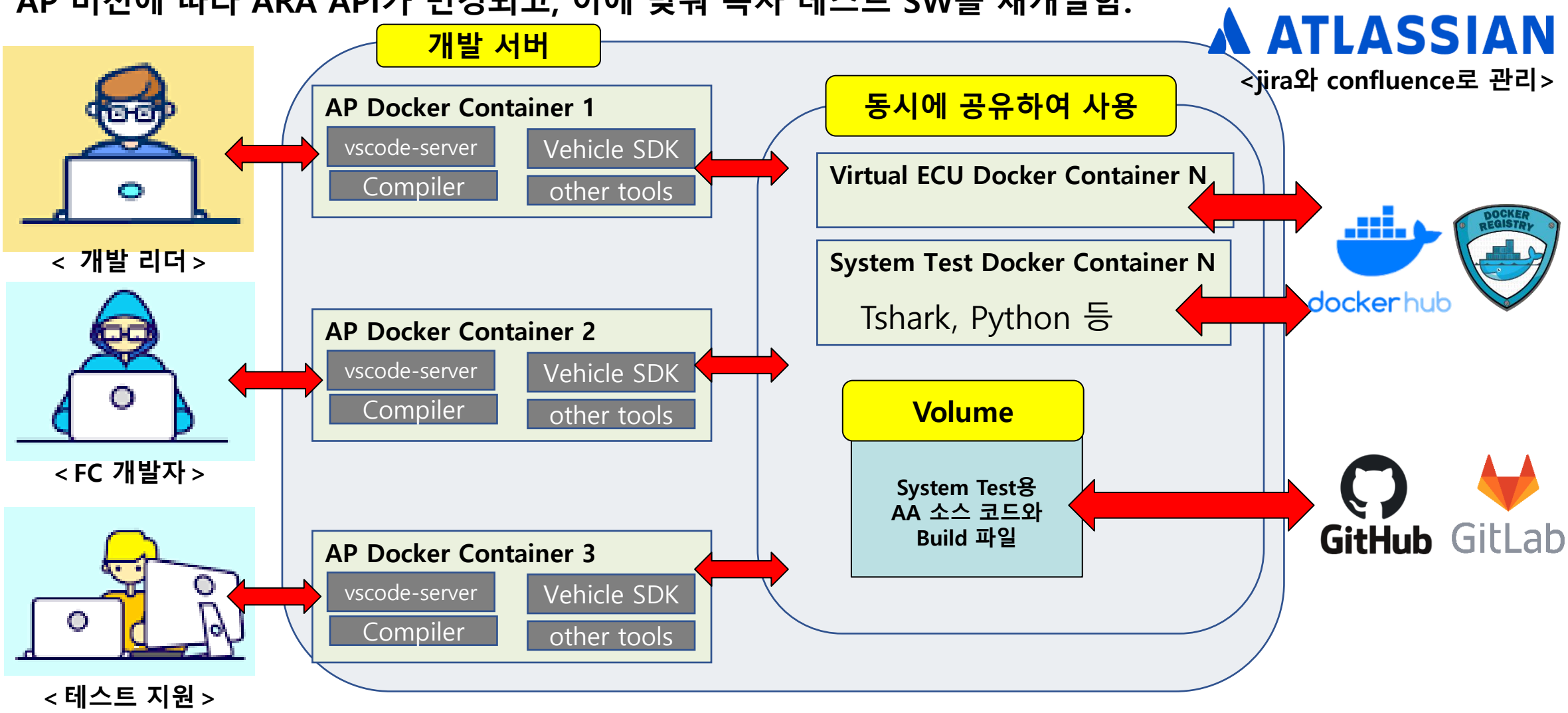
- System Test용 Docker Container에서 tshark를 사용하여 가상ECU(ARM등)의 모니터링 가능함.
- 사용자의 PC에 Wireshark가 설치 되어있지 않더라도 이미 PACON IDE에서 설치되어 있어 사용할 수 있음.



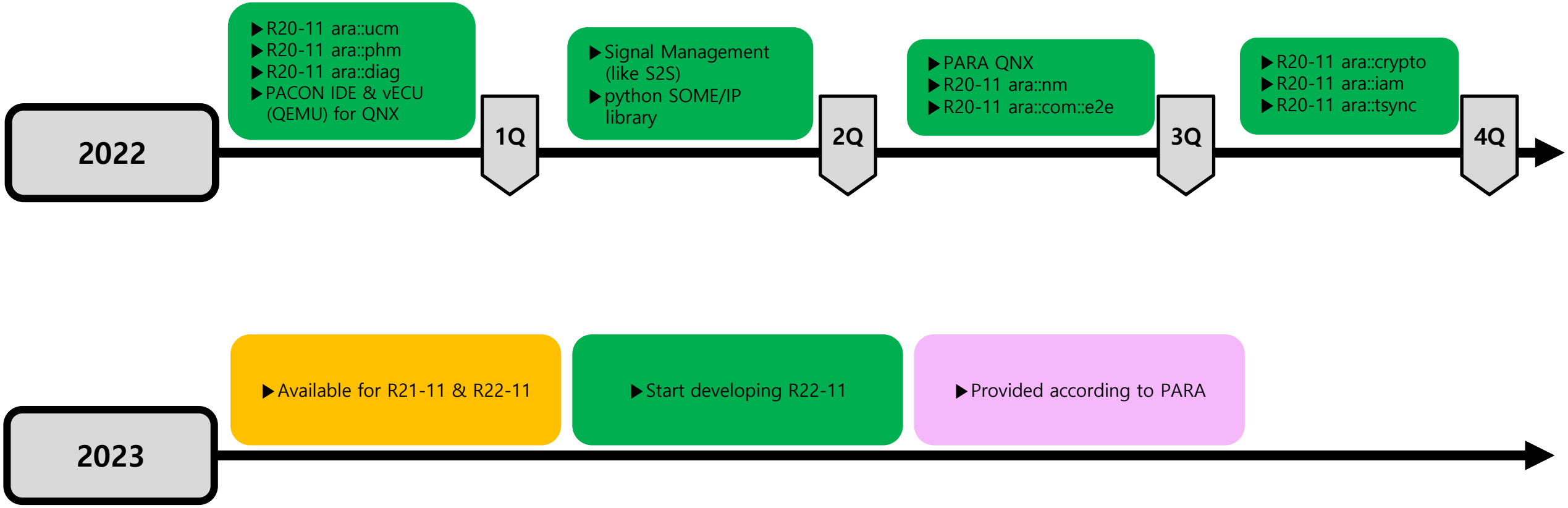
PopcornSAR

6. 팝콘사 AP 개발 환경 구축 예시

- 팝콘사는 QEMU와 Yocto를 사용하지 않음.
- AP 버전에 따라 ARA API가 변경되고, 이에 맞춰 독자 테스트 SW를 재개발함.



7. 팝콘사 AP 개발 로드맵 (~2023)



(참고자료1) 제품 시연 데모 영상

1. **R20-11 Autosar.io ARXML modeling :**
   - Link : https://youtu.be/3FYzR0bQ44s

2. **R20-11 generation & build for communication between 2 Adaptive Applications :**
   - Link : https://youtu.be/HYqNEMrYYAw

3. **R20-11 ARA::DIAG demo:**
   - Link1 : https://youtu.be/jvySoUdoAJs
   - Link2 : https://youtu.be/tm_Cr80d52w

(참고자료2) 팝콘사 모델링 도구(AutoSAR.io) ISO26262 인증서

KAPA

ARCS

Certificate of Achievement

This is to certify that

PopcornSAR Co., Ltd.
Development of an AUTOSAR Adaptive Platform Authoring Tool supporting Automotive OTA design and being qualified with Automotive Functional Safety

has successfully achieved

Functional safety process capability Level 1
for ASIL B base practices
Nov. 22, 2021

Standard

ISO/IEC 33002:2015 Information technology – Process assessment – Requirements for performing process assessment

Assessment Model

SS 7740:2018 Road vehicle – Functional safety process assessment model

SS 7740 supports to audit the processes of ISO 26262:2011 Road vehicles- Functional safety(all parts)

Gil Jo Kim
Principal Assessor
82216102

Details of Achievement

Process ID	Process Name	Capability Level 1
		ASIL B base practices
ENG.4.SE	Software requirements analysis	●
ENG.5.SE	Software design	●
ENG.6.SE	Software construction	●
ENG.7.SE	Software integration test	●
ENG.8.SE	Software testing	●