SAFE architecture for Robust distributed Application Integration in roLling Stock 2

CONtributing to Shift2Rail's NExt generation of high Capable and safe TCMS. Phase 2

Functional Distribution Framework and Application Profiles

Thomas Waschulzik, SIEMENS thomas.waschulzik@siemens.com

Xabier Artaetxebarria,CAF xartaetxebarria@caf.net

Martin Zauner, LIEBHERR martin.zauner@liebherr.com

Technical Seminar on Advanced Architectures and Components for Next-Generation TCMS

January 21st 2020, Brussels

The projects CONNECTA-2 and Safe4RAIL-2 have received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 826098 (CONNECTA-2) and No. 826073 (Safe4RAIL-2) respectively. The information and views set out in this document are those of the author(s) and do not necessarily reflect the official opinion of Shift2Rail Joint Undertaking. The JU does not guarantee the accuracy of the data included in this article. Neither the JU nor any person acting on the JU's behalf may be held responsible for the use which may be made of the information contained therein.

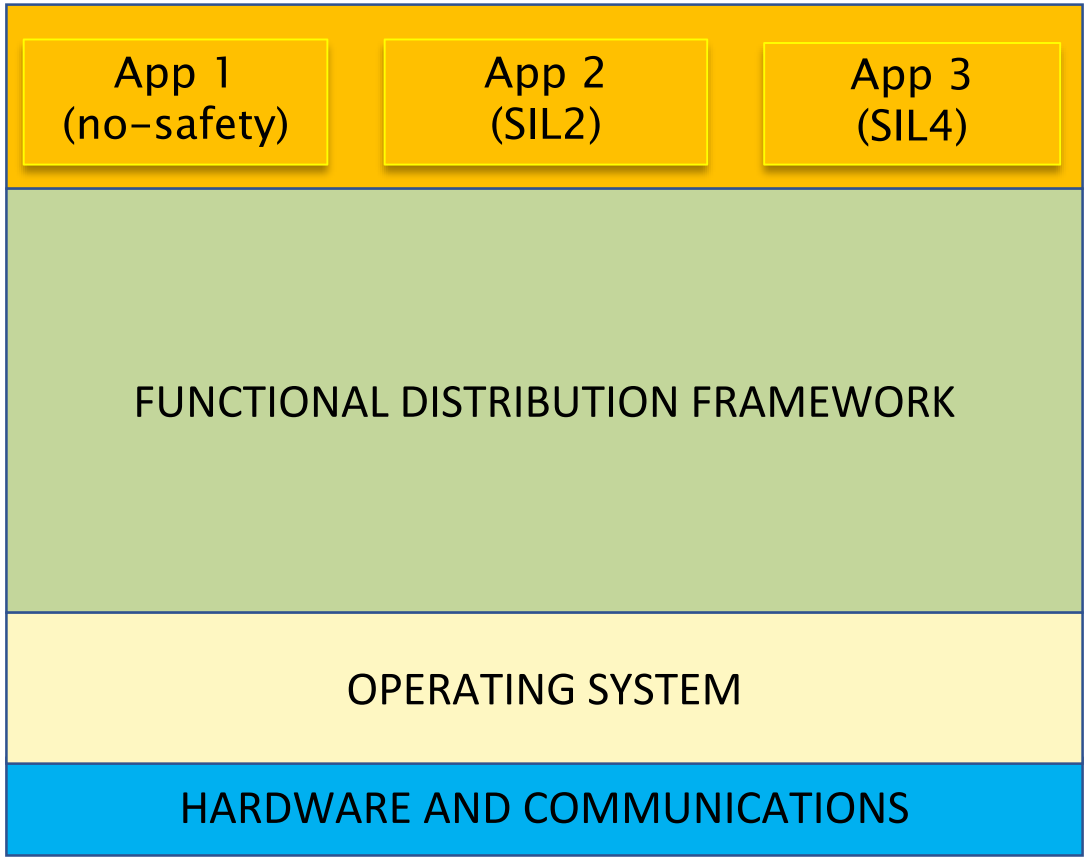
Outline

1. Introduction
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

Outline

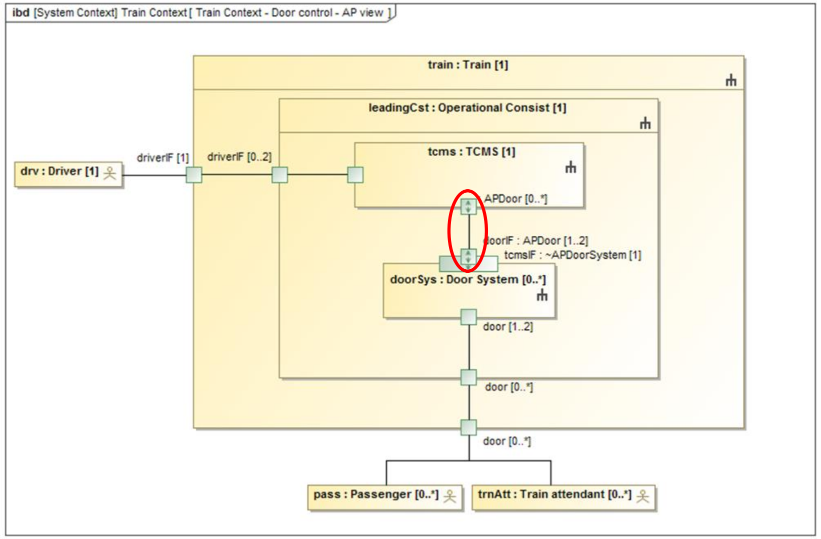
1. **Introduction**
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

FDF: Functional Distribution Framework


App 1
(no-safety)
App 2
(SIL2)
App 3
(SIL4)
FUNCTIONAL DISTRIBUTION FRAMEWORK
OPERATING SYSTEM
HARDWARE AND COMMUNICATIONS

- A framework with a standardized API for the development of distributed train functions (e.g. Air Condition, Doors Control, …)

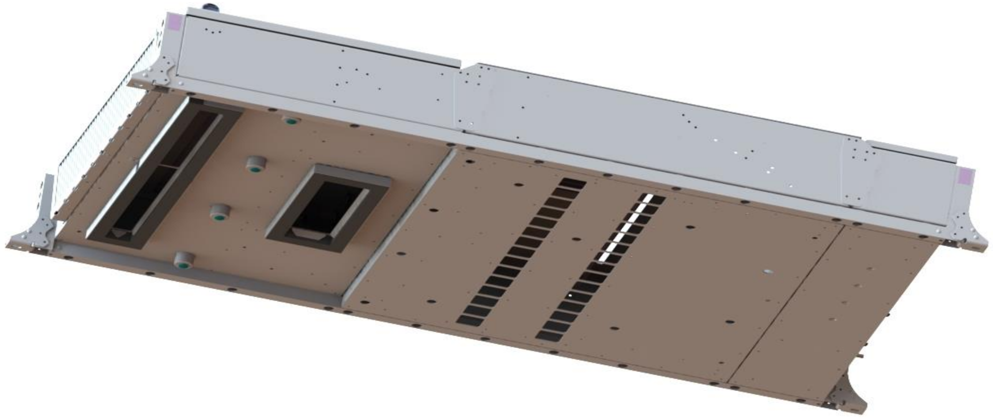
AP: Application Profile


ibd [System Context] Train Context [Train Context - Door control - AP view]
train : Train [1]
leadingCst : Operational Consist [1]
tcms : TCMS [1]
drv : Driver [1]
driverIF [1]
driverIF [0..2]
APDoor [0..*]
doorIF : APDoor [1..2]
tcmsIF : ~APDoorSystem [1]
doorSys : Door System [0..*]
door [1..2]
door [0..*]
door [0..*]
pass : Passenger [0..*]
trnAtt : Train attendant [0..*]

Train Context - Door control - AP view

- According to our project goal an Application Profile describes a **functional interface** between the Train Control and Monitoring System (TCMS) and a subsystem
- The interface definition is based on an analysis of which use cases have to be supported and defines the information (flow properties) that can be exchanged between the communication partners

HVAC: Heating, ventilation, and air conditioning



- System to provide thermal comfort and indoor air quality inside the train.

Outline

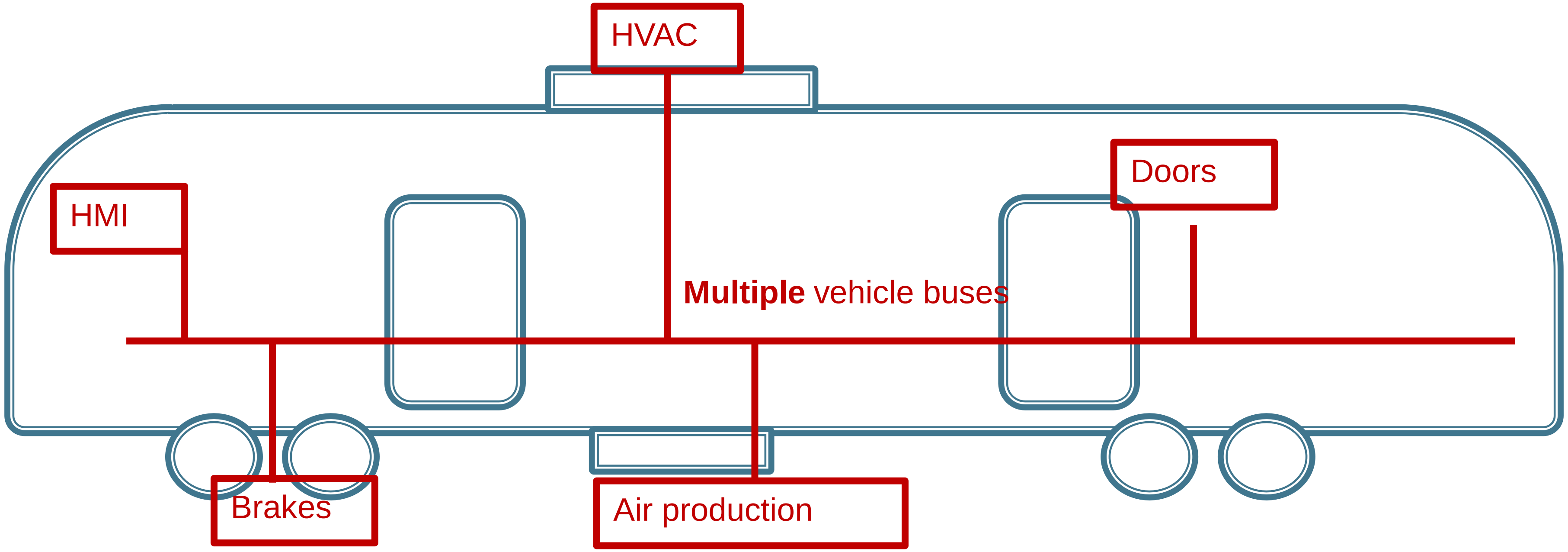
1. Introduction
2. **FDF**
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

Outline

1. Introduction
2. FDF
   - 2.1 **Motivation**
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

Classical Architecture

- Separate functions in separate subsystems, multiple communication buses
- Many different systems, controllers, wires, connectors
- Maintenance and (re)commissioning high complexity and lifecycle costs

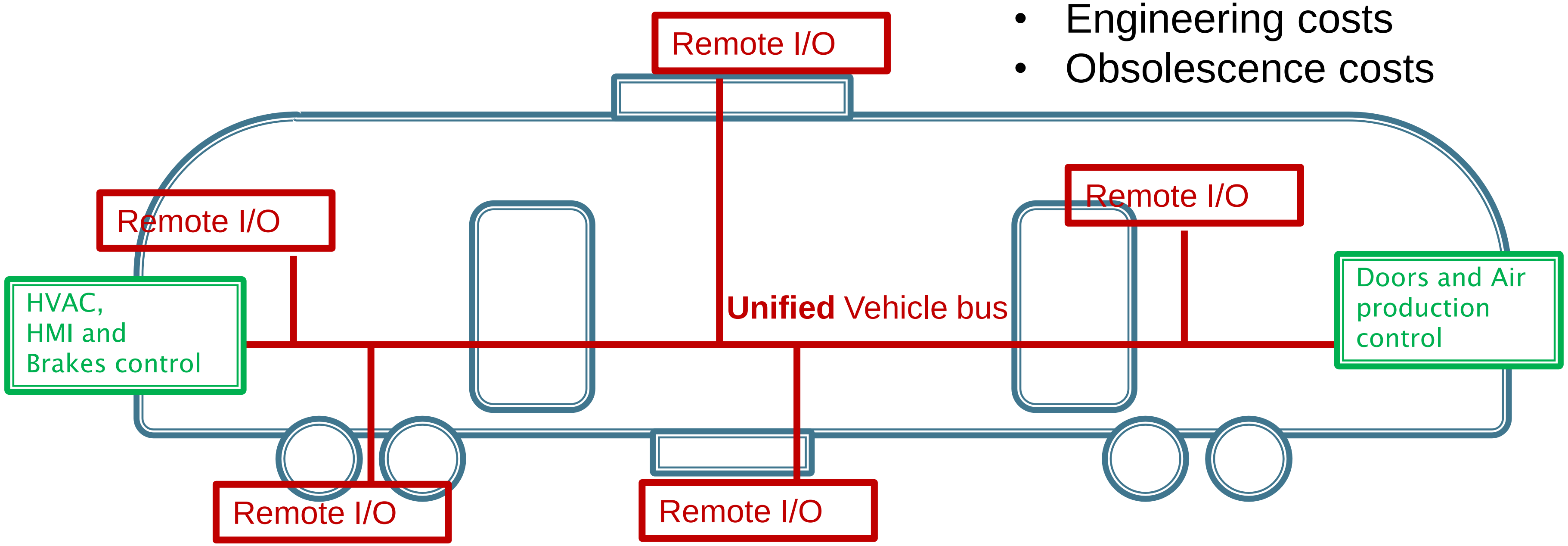


Future Architecture

- Advanced Integrated Modular Architecture for Train Control and Monitoring System (TCMS)

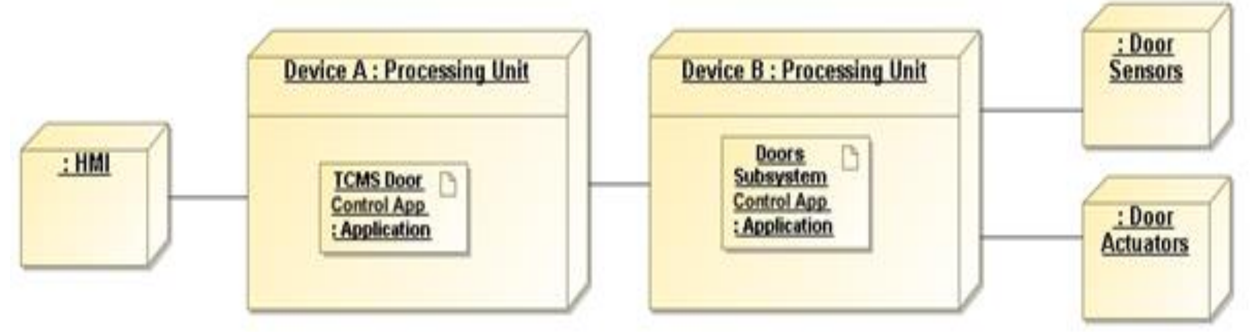
Reduces:

- Diversity and complexity of architecture
- Engineering costs
- Obsolescence costs

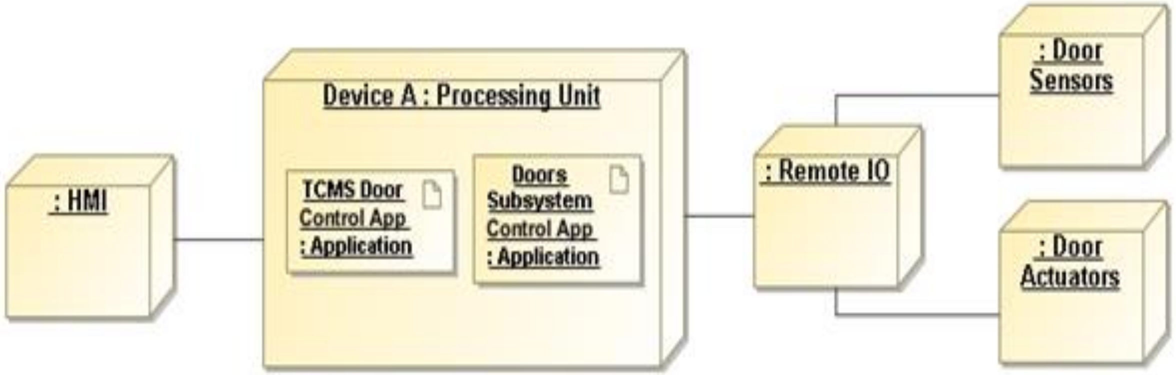


Use example: Door control

Without FDF


: HMI
Device A : Processing Unit
TCMS Door Control App : Application
Device B : Processing Unit
Doors Subsystem Control App : Application
: Door Sensors
: Door Actuators

With FDF


: HMI
Device A : Processing Unit
TCMS Door Control App : Application
Doors Subsystem Control App : Application
: Remote IO
: Door Sensors
: Door Actuators

Main Goal

- Provide the “Functional Distribution” architecture concept for a mixed criticality embedded platform, offering an execution environment:
  - for multiple Train Control and Monitoring System (TCMS) application functions inside the end-system
  - that ensures:
    - Common API and Services
    - Portable Applications between different FDF Implementations
    - Abstraction from the underlying network protocols and hardware

Outline

1. Introduction

2. FDF

   2.1 Motivation

   2.2 Architecture and API

   2.3 Implementations

3. Application Profile

   3.1 Motivation

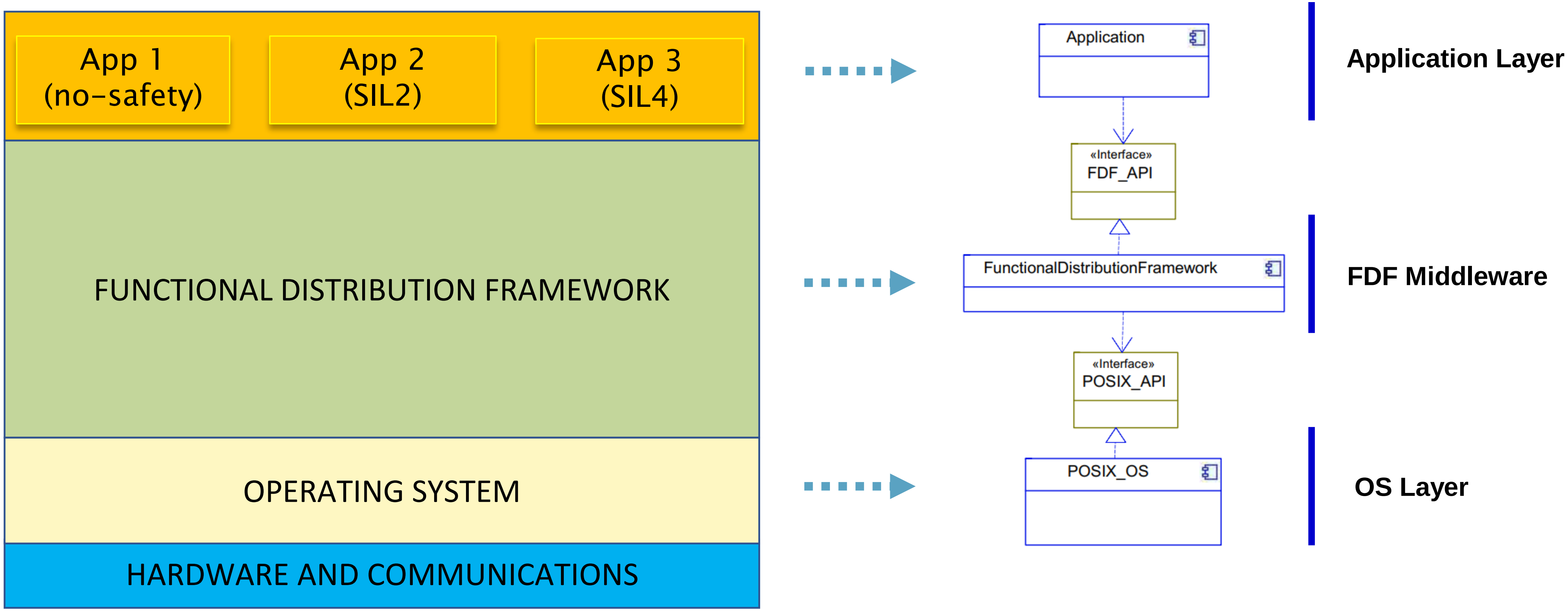
   3.2 Application Profile for HVAC

   3.3 Interface Generation

   3.4 Integration of HVAC Application on FDF

4. Summary

System View



App 1 (no-safety)

App 2 (SIL2)

App 3 (SIL4)

FUNCTIONAL DISTRIBUTION FRAMEWORK

OPERATING SYSTEM

HARDWARE AND COMMUNICATIONS

Application

«Interface» FDF_API

FunctionalDistributionFramework

«Interface» POSIX_API

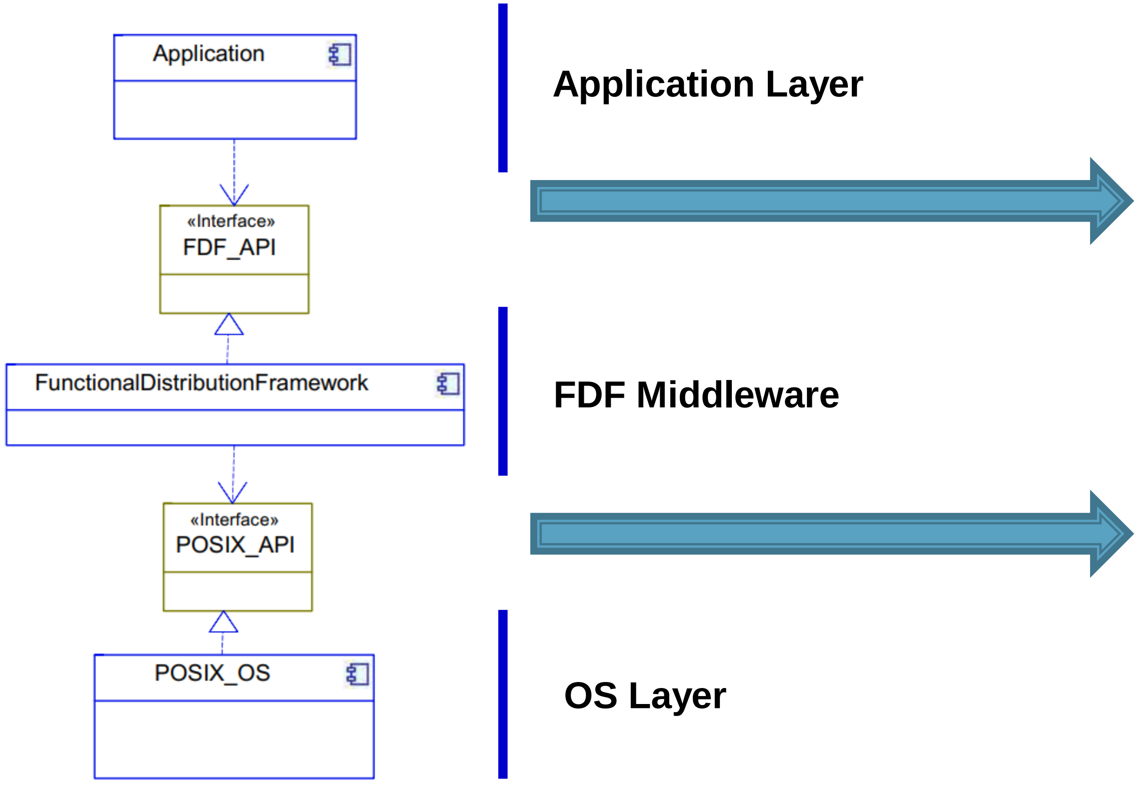
POSIX_OS

Application Layer

FDF Middleware

OS Layer

FDF API + AP



- FDF_API
  - Implement user logic
  - Access to Application Profile Variables
  - Access to FDF Services
  - Enables portability of applications
- POSIX OS (Not visible to applications)
  - OS services
  - Drivers functionality

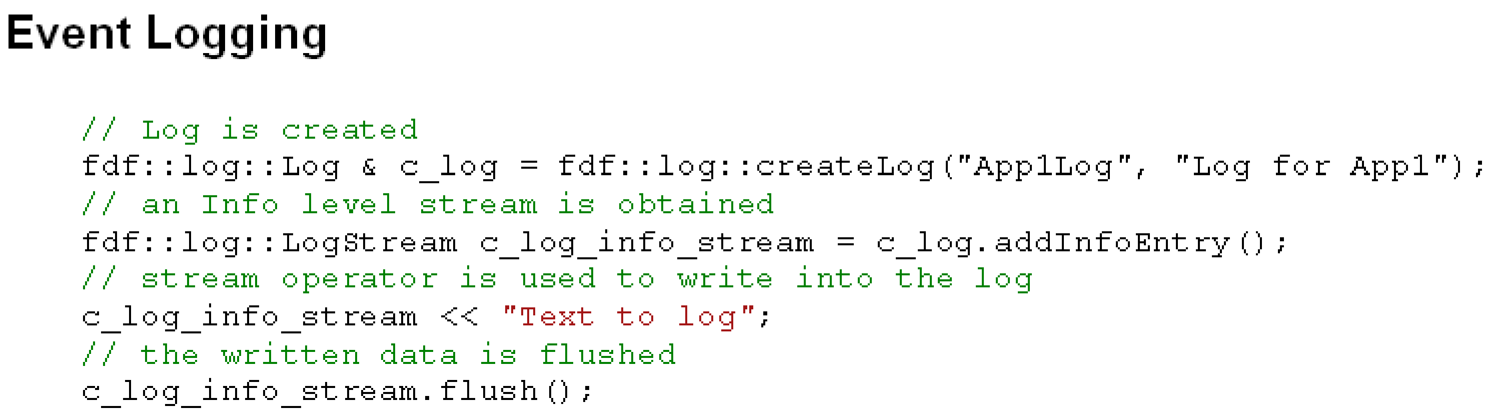
Why using a Service Oriented Architecture?

Strong decoupling provided

- ➔Train layer: In case of functional open coupling multiple suppliers and operators may contribute to a train
- ➔TCMS ⬄ Subsystem: Separation between system integrator and supplier of the component

FDF API

- C++14
- Available in CTA-2 Deliverable D1.2 (Public)
- Grouped in Functional Components



Event Logging

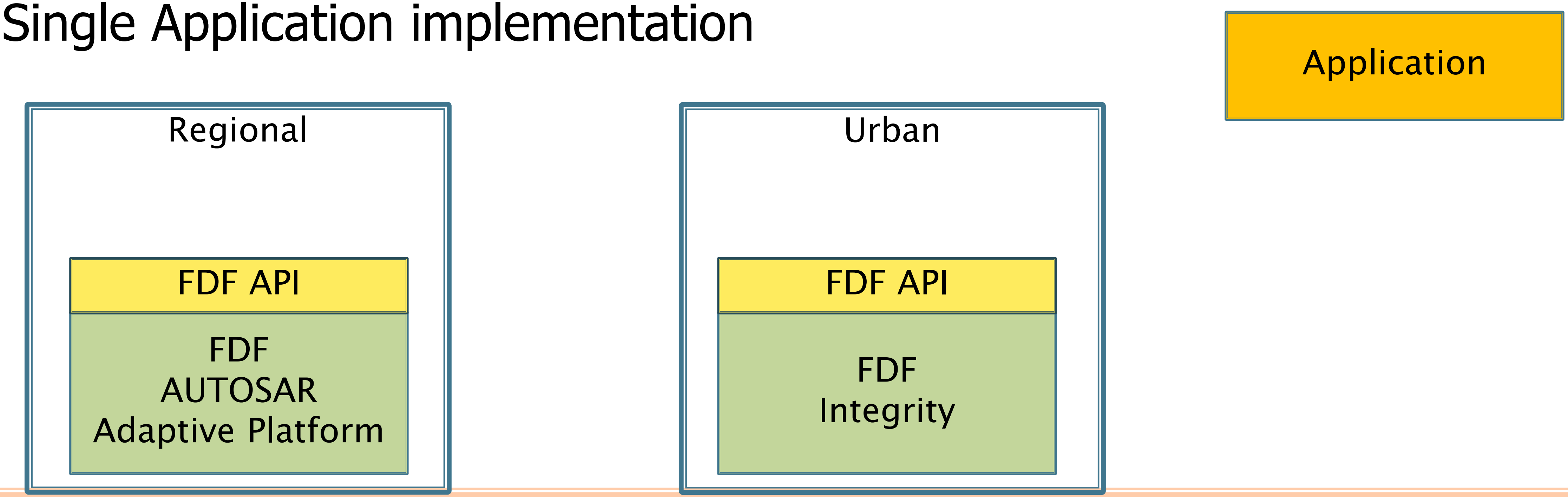
```
// Log is created
fdf::log::Log & c_log = fdf::log::createLog("Appl1Log", "Log for Appl");
// an Info level stream is obtained
fdf::log::LogStream c_log_info_stream = c_log.addInfoEntry();
// stream operator is used to write into the log
c_log_info_stream << "Text to log";
// the written data is flushed
c_log_info_stream.flush();
```

Outline

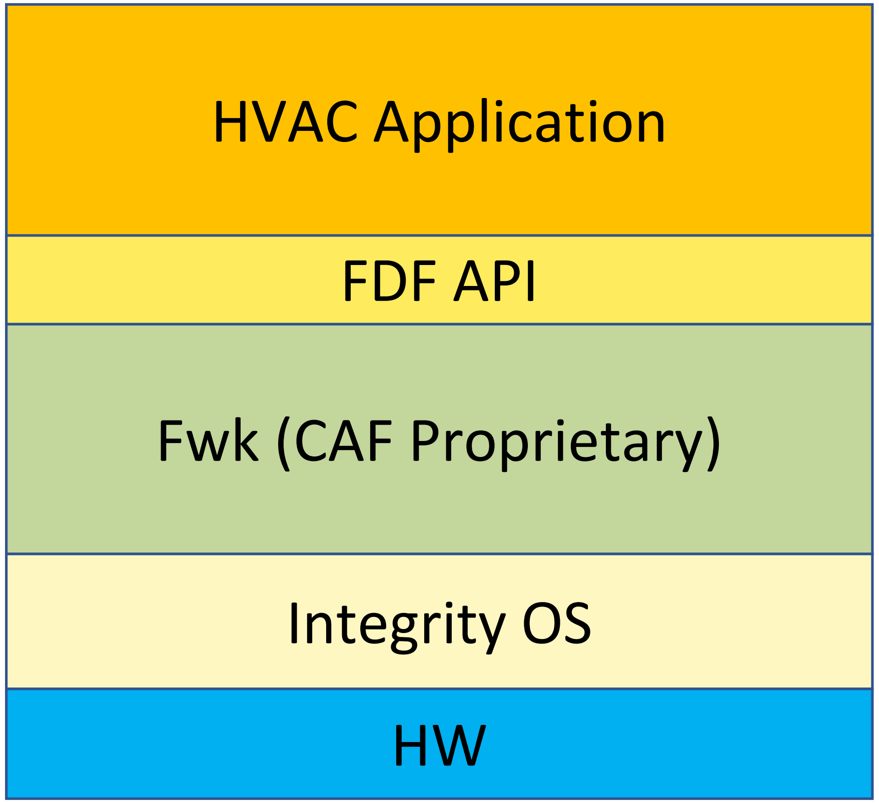
1. Introduction
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 **Implementations**
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

FDF Implementations

- 2 different implementations
  - Based on AUTOSAR Adaptive Platform → Regional Demonstrator
  - Based on Integrity RTOS → Urban Demonstrator
- Single Application implementation



Urban demonstrator: Integrity FDF



HVAC Application

FDF API

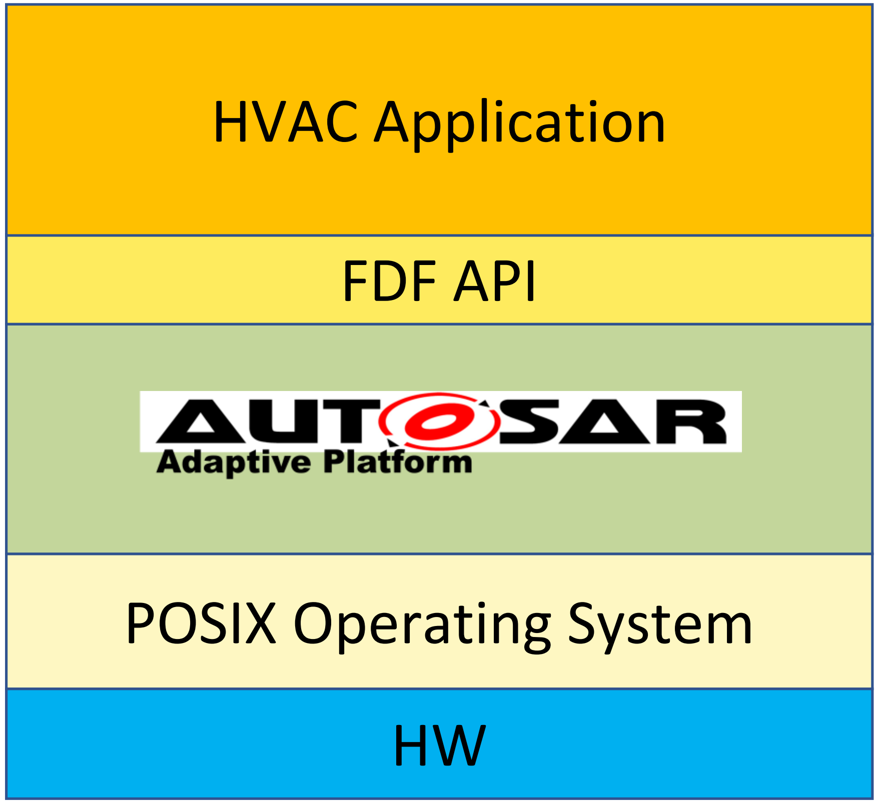
Fwk (CAF Proprietary)

Integrity OS

HW

- HVAC Application is independent of RTOS and of FDF implementation
- GreenHills Integrity is a certified Real Time Operating System (RTOS) for critical applications
- It facilitates implementation of core FDF requirements:
  - Limited execution time for each application
  - Limited hardware resources for each application

Regional demonstrator: AUTOSAR FDF



HVAC Application

FDF API

AUTOSAR Adaptive Platform

POSIX Operating System

HW

- HVAC Application is independent of the FDF implementation
- AUTOSAR Adaptive Platform highly compatible with FDF architecture and requirements
- Highly portable FDF to POSIX PSE51 Operating systems
- Suitable for high performance computing, high bandwidth communication and connectivity

AUTOSAR: AUTomotive Open System ARchitecture

Outline

1. Introduction
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. **Application Profile**
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

Outline

1. Introduction
2. FDF
   2.1 Motivation
   2.2 Architecture and API
   2.3 Implementations
3. Application Profile
   3.1 **Motivation**
   3.2 Application Profile for HVAC
   3.3 Interface Generation
   3.4 Integration of HVAC Application on FDF
4. Summary

Why did we define the Application Profiles?

It reduces:

- Engineering costs due to standardization of
  - Requirements for the subsystem (e. g. Use cases)
  - Interface between TCMS and the subsystem
  - Documentation
  - Tests
- Project duration due to less negotiations between subsystem supplier and integrator
- Problems during system introduction phase, due to less changed software and hardware components

Selected Application Profiles

1. HVAC (Heating, Ventilation and Air Conditioning)
2. Doors: Review together with PIVOT-2 (TD1.6)
3. BMS (Bogie Monitoring System)
4. ATO (Automatic Train Operation) over ETCS (European Train Control System) subset 139, together with X2RAIL-2
5. Lavatories

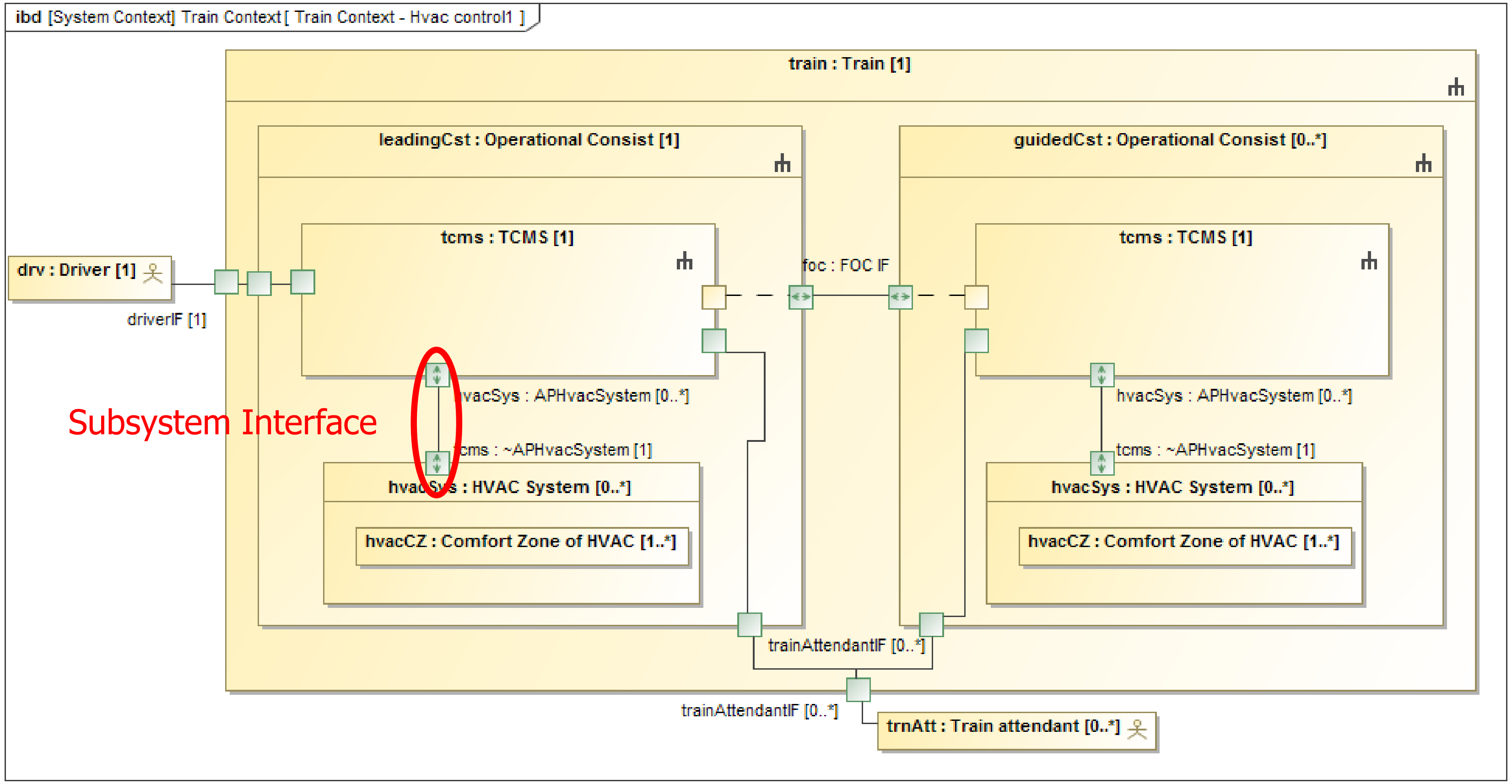
Defined by CONNECTA-1

Defined by CONNECTA-2

Outline

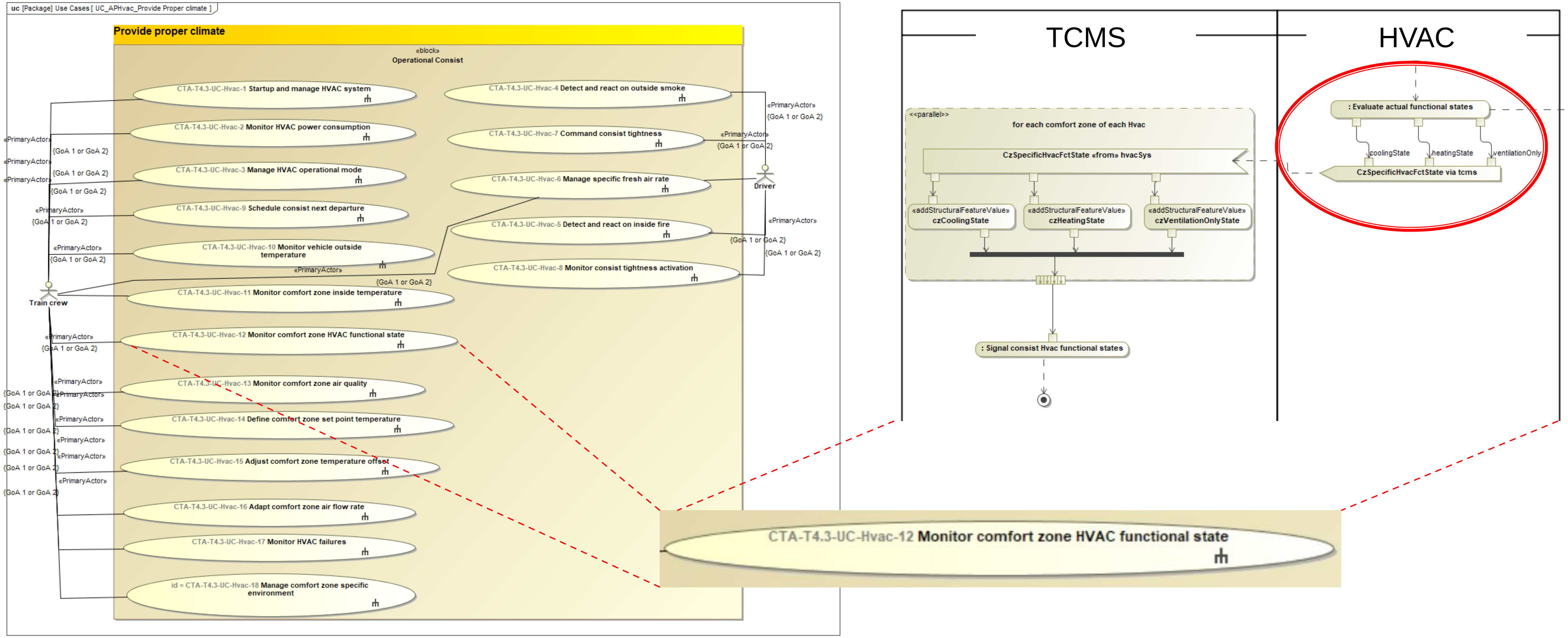
1. Introduction
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - **3.2 Application Profile for HVAC**
   - 3.3 Interface Generation
   - 3.4 Integration of HVAC Application on FDF
4. Summary

Application Profile for HVAC (Context Diagram)

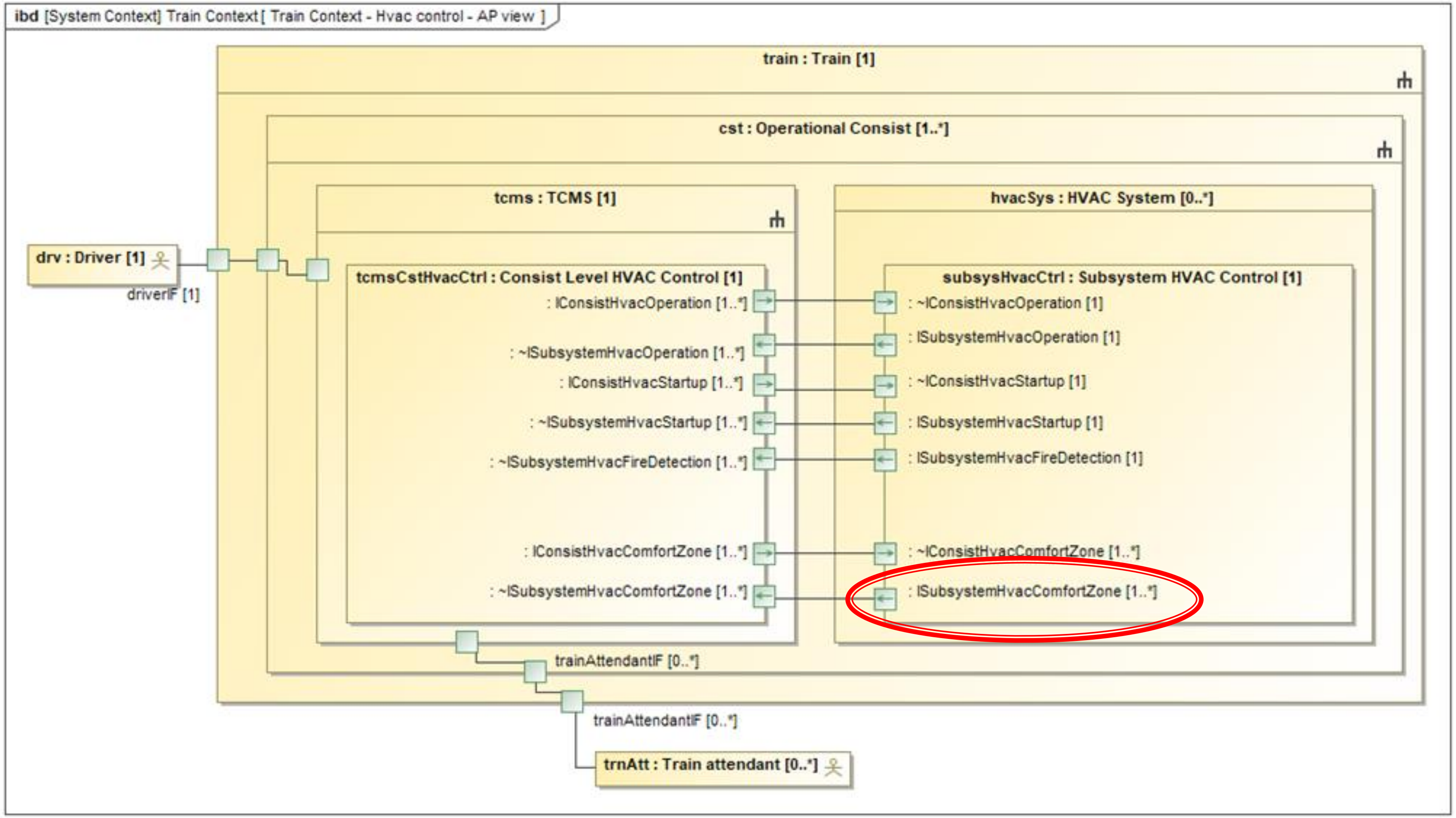

ibd [System Context] Train Context [Train Context - Hvac control1]
train : Train [1]
leadingCst : Operational Consist [1]
guidedCst : Operational Consist [0..*]
tcms : TCMS [1]
tcms : TCMS [1]
drv : Driver [1]
driverIF [1]
foc : FOC IF
hvacSys : APHvacSystem [0..*]
tcms : ~APHvacSystem [1]
hvacSys : HVAC System [0..*]
hvacCZ : Comfort Zone of HVAC [1..*]
hvacSys : APHvacSystem [0..*]
tcms : ~APHvacSystem [1]
hvacSys : HVAC System [0..*]
hvacCZ : Comfort Zone of HVAC [1..*]
trainAttendantIF [0..*]
trainAttendantIF [0..*]
trnAtt : Train attendant [0..*]

Subsystem Interface

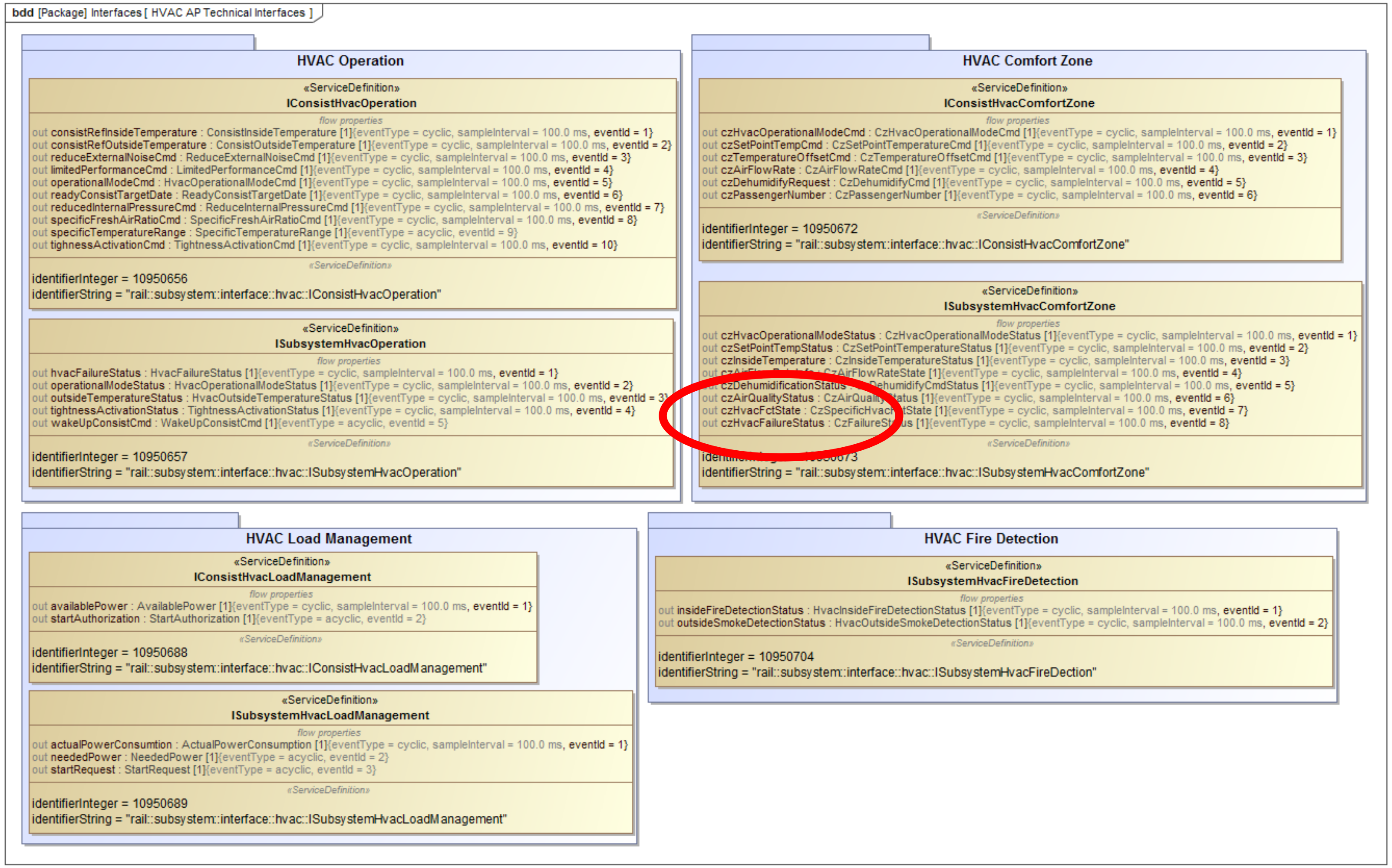
Application Profile for HVAC (Use Cases)


uc [Package] Use Cases [UC_APHvac_Provide Proper climate]
Provide proper climate
«block»
Operational Consist
CTA-T4.3-UC-Hvac-1 Startup and manage HVAC system
CTA-T4.3-UC-Hvac-2 Monitor HVAC power consumption
CTA-T4.3-UC-Hvac-3 Manage HVAC operational mode
CTA-T4.3-UC-Hvac-9 Schedule consist next departure
CTA-T4.3-UC-Hvac-10 Monitor vehicle outside temperature
CTA-T4.3-UC-Hvac-11 Monitor comfort zone inside temperature
CTA-T4.3-UC-Hvac-12 Monitor comfort zone HVAC functional state
CTA-T4.3-UC-Hvac-13 Monitor comfort zone air quality
CTA-T4.3-UC-Hvac-14 Define comfort zone set point temperature
CTA-T4.3-UC-Hvac-15 Adjust comfort zone temperature offset
CTA-T4.3-UC-Hvac-16 Adapt comfort zone air flow rate
CTA-T4.3-UC-Hvac-17 Monitor HVAC failures
id = CTA-T4.3-UC-Hvac-18 Manage comfort zone specific environment
CTA-T4.3-UC-Hvac-4 Detect and react on outside smoke
CTA-T4.3-UC-Hvac-7 Command consist tightness
CTA-T4.3-UC-Hvac-6 Manage specific fresh air rate
CTA-T4.3-UC-Hvac-5 Detect and react on inside fire
CTA-T4.3-UC-Hvac-8 Monitor consist tightness activation
«PrimaryActor»
{GoA 1 or GoA 2}
Train crew
Driver
TCMS
HVAC
: Evaluate actual functional states
coolingState
heatingState
ventilationOnly
CzSpecificHvacFctState via tcms
<<parallel>>
for each comfort zone of each Hvac
CzSpecificHvacFctState «from» hvacSys
«addStructuralFeatureValue»
czCoolingState
«addStructuralFeatureValue»
czHeatingState
«addStructuralFeatureValue»
czVentilationOnlyState
: Signal consist Hvac functional states
CTA-T4.3-UC-Hvac-12 Monitor comfort zone HVAC functional state

Application Profile for HVAC (Technical Interface)


ibd [System Context] Train Context [Train Context - Hvac control - AP view]
train : Train [1]
cst : Operational Consist [1..*]
tcms : TCMS [1]
hvacSys : HVAC System [0..*]
drv : Driver [1]
driverIF [1]
tcmsCstHvacCtrl : Consist Level HVAC Control [1]
: IConsistHvacOperation [1..*]
: ~ISubsystemHvacOperation [1..*]
: IConsistHvacStartup [1..*]
: ~ISubsystemHvacStartup [1..*]
: ~ISubsystemHvacFireDetection [1..*]
: IConsistHvacComfortZone [1..*]
: ~ISubsystemHvacComfortZone [1..*]
subsysHvacCtrl : Subsystem HVAC Control [1]
: ~IConsistHvacOperation [1]
: ISubsystemHvacOperation [1]
: ~IConsistHvacStartup [1]
: ISubsystemHvacStartup [1]
: ISubsystemHvacFireDetection [1]
: ~IConsistHvacComfortZone [1..*]
: ISubsystemHvacComfortZone [1..*]
trainAttendantIF [0..*]
trainAttendantIF [0..*]
trnAtt : Train attendant [0..*]

Application Profile for HVAC (Technical Interface)



bdd [Package] Interfaces [HVAC AP Technical Interfaces]

HVAC Operation

«ServiceDefinition»
IConsistHvacOperation

flow properties

out consistRefInsideTemperature : ConsistInsideTemperature [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out consistRefOutsideTemperature : ConsistOutsideTemperature [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out reduceExternalNoiseCmd : ReduceExternalNoiseCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out limitedPerformanceCmd : LimitedPerformanceCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out operationalModeCmd : HvacOperationalModeCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 5}
out readyConsistTargetDate : ReadyConsistTargetDate [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 6}
out reducedInternalPressureCmd : ReduceInternalPressureCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 7}
out specificFreshAirRatioCmd : SpecificFreshAirRatioCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 8}
out specificTemperatureRange : SpecificTemperatureRange [1]{eventType = acyclic, eventId = 9}
out tighnessActivationCmd : TightnessActivationCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 10}

«ServiceDefinition»

identifierInteger = 10950656
identifierString = "rail::subsystem::interface::hvac::IConsistHvacOperation"

«ServiceDefinition»
ISubsystemHvacOperation

flow properties

out hvacFailureStatus : HvacFailureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out operationalModeStatus : HvacOperationalModeStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out outsideTemperatureStatus : HvacOutsideTemperatureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out tightnessActivationStatus : TightnessActivationStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out wakeUpConsistCmd : WakeUpConsistCmd [1]{eventType = acyclic, eventId = 5}

«ServiceDefinition»

identifierInteger = 10950657
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacOperation"

HVAC Comfort Zone

«ServiceDefinition»
IConsistHvacComfortZone

flow properties

out czHvacOperationalModeCmd : CzHvacOperationalModeCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out czSetPointTempCmd : CzSetPointTemperatureCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out czTemperatureOffsetCmd : CzTemperatureOffsetCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out czAirFlowRate : CzAirFlowRateCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out czDehumidifyRequest : CzDehumidifyCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 5}
out czPassengerNumber : CzPassengerNumber [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 6}

«ServiceDefinition»

identifierInteger = 10950672
identifierString = "rail::subsystem::interface::hvac::IConsistHvacComfortZone"

«ServiceDefinition»
ISubsystemHvacComfortZone

flow properties

out czHvacOperationalModeStatus : CzHvacOperationalModeStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out czSetPointTempStatus : CzSetPointTemperatureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out czInsideTemperature : CzInsideTemperatureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out czAirFlowRateInfo : CzAirFlowRateState [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out czDehumidificationStatus : CzDehumidifyCmdStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 5}
out czAirQualityStatus : CzAirQualityStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 6}
out czHvacFctState : CzSpecificHvacFctState [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 7}
out czHvacFailureStatus : CzFailureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 8}

«ServiceDefinition»

identifierInteger = 10950673
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacComfortZone"

HVAC Load Management

«ServiceDefinition»
IConsistHvacLoadManagement

flow properties

out availablePower : AvailablePower [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out startAuthorization : StartAuthorization [1]{eventType = acyclic, eventId = 2}

«ServiceDefinition»

identifierInteger = 10950688
identifierString = "rail::subsystem::interface::hvac::IConsistHvacLoadManagement"

«ServiceDefinition»
ISubsystemHvacLoadManagement

flow properties

out actualPowerConsumtion : ActualPowerConsumption [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out neededPower : NeededPower [1]{eventType = acyclic, eventId = 2}
out startRequest : StartRequest [1]{eventType = acyclic, eventId = 3}

«ServiceDefinition»

identifierInteger = 10950689
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacLoadManagement"

HVAC Fire Detection

«ServiceDefinition»
ISubsystemHvacFireDetection

flow properties

out insideFireDetectionStatus : HvacInsideFireDetectionStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out outsideSmokeDetectionStatus : HvacOutsideSmokeDetectionStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}

«ServiceDefinition»

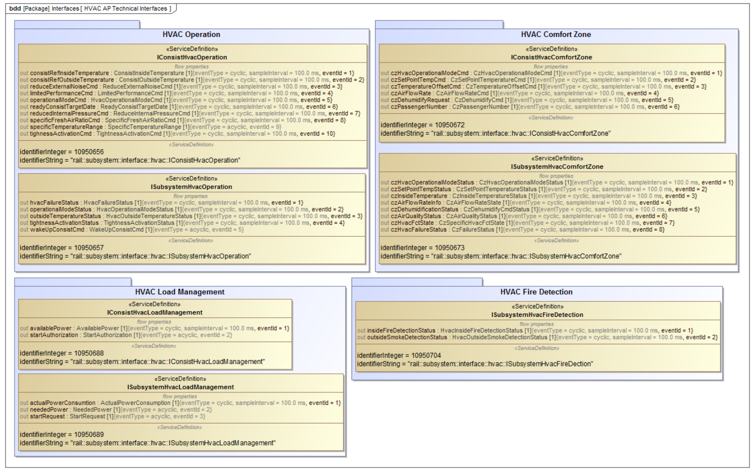
identifierInteger = 10950704
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacFireDection"

Outline

1. Introduction
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
4. **Development and Integration**
   - 4.1 **Interface Generation**
   - 4.2 Integration of HVAC Application on FDF
5. Summary

Technical Interface

- From SysML model to Interface Implementation



bdd [Package] Interfaces [HVAC AP Technical Interfaces]

HVAC Operation

«ServiceDefinition»
IConsistHvacOperation
flow properties
out consistRefInsideTemperature : ConsistInsideTemperature [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out consistRefOutsideTemperature : ConsistOutsideTemperature [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out reduceExternalNoiseCmd : ReduceExternalNoiseCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out limitedPerformanceCmd : LimitedPerformanceCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out operationalModeCmd : HvacOperationalModeCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 5}
out readyConsistTargetDate : ReadyConsistTargetDate [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 6}
out reducedInternalPressureCmd : ReduceInternalPressureCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 7}
out specificFreshAirRatioCmd : SpecificFreshAirRatioCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 8}
out specificTemperatureRange : SpecificTemperatureRange [1]{eventType = acyclic, eventId = 9}
out tighnessActivationCmd : TightnessActivationCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 10}
«ServiceDefinition»
identifierInteger = 10950656
identifierString = "rail::subsystem::interface::hvac::IConsistHvacOperation"

«ServiceDefinition»
ISubsystemHvacOperation
flow properties
out hvacFailureStatus : HvacFailureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out operationalModeStatus : HvacOperationalModeStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out outsideTemperatureStatus : HvacOutsideTemperatureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out tightnessActivationStatus : TightnessActivationStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out wakeUpConsistCmd : WakeUpConsistCmd [1]{eventType = acyclic, eventId = 5}
«ServiceDefinition»
identifierInteger = 10950657
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacOperation"

HVAC Comfort Zone

«ServiceDefinition»
IConsistHvacComfortZone
flow properties
out czHvacOperationalModeCmd : CzHvacOperationalModeCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out czSetPointTempCmd : CzSetPointTemperatureCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out czTemperatureOffsetCmd : CzTemperatureOffsetCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out czAirFlowRate : CzAirFlowRateCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out czDehumidifyRequest : CzDehumidifyCmd [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 5}
out czPassengerNumber : CzPassengerNumber [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 6}
«ServiceDefinition»
identifierInteger = 10950672
identifierString = "rail::subsystem::interface::hvac::IConsistHvacComfortZone"

«ServiceDefinition»
ISubsystemHvacComfortZone
flow properties
out czHvacOperationalModeStatus : CzHvacOperationalModeStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out czSetPointTempStatus : CzSetPointTemperatureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
out czInsideTemperature : CzInsideTemperatureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 3}
out czAirFlowRateInfo : CzAirFlowRateState [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 4}
out czDehumidificationStatus : CzDehumidifyCmdStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 5}
out czAirQualityStatus : CzAirQualityStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 6}
out czHvacFctState : CzSpecificHvacFctState [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 7}
out czHvacFailureStatus : CzFailureStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 8}
«ServiceDefinition»
identifierInteger = 10950673
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacComfortZone"

HVAC Load Management

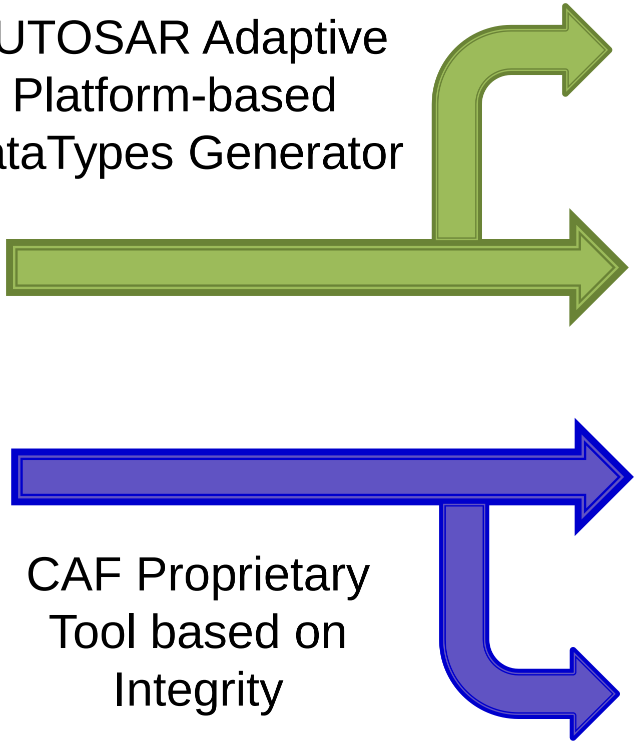
«ServiceDefinition»
IConsistHvacLoadManagement
flow properties
out availablePower : AvailablePower [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out startAuthorization : StartAuthorization [1]{eventType = acyclic, eventId = 2}
«ServiceDefinition»
identifierInteger = 10950688
identifierString = "rail::subsystem::interface::hvac::IConsistHvacLoadManagement"

«ServiceDefinition»
ISubsystemHvacLoadManagement
flow properties
out actualPowerConsumtion : ActualPowerConsumption [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out neededPower : NeededPower [1]{eventType = acyclic, eventId = 2}
out startRequest : StartRequest [1]{eventType = acyclic, eventId = 3}
«ServiceDefinition»
identifierInteger = 10950689
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacLoadManagement"

HVAC Fire Detection

«ServiceDefinition»
ISubsystemHvacFireDetection
flow properties
out insideFireDetectionStatus : HvacInsideFireDetectionStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 1}
out outsideSmokeDetectionStatus : HvacOutsideSmokeDetectionStatus [1]{eventType = cyclic, sampleInterval = 100.0 ms, eventId = 2}
«ServiceDefinition»
identifierInteger = 10950704
identifierString = "rail::subsystem::interface::hvac::ISubsystemHvacFireDection"

Application Profile for HVAC

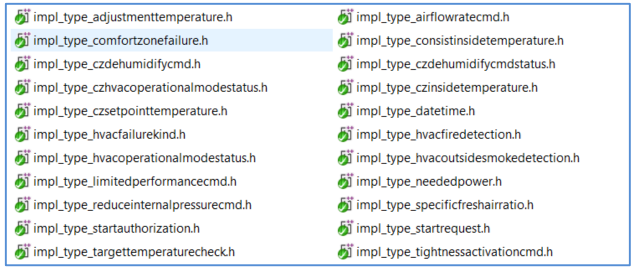


AUTOSAR Adaptive Platform-based DataTypes Generator

CAF Proprietary Tool based on Integrity

Implementations for AUTOSAR based FDF - APIs: Skeletons & Proxies

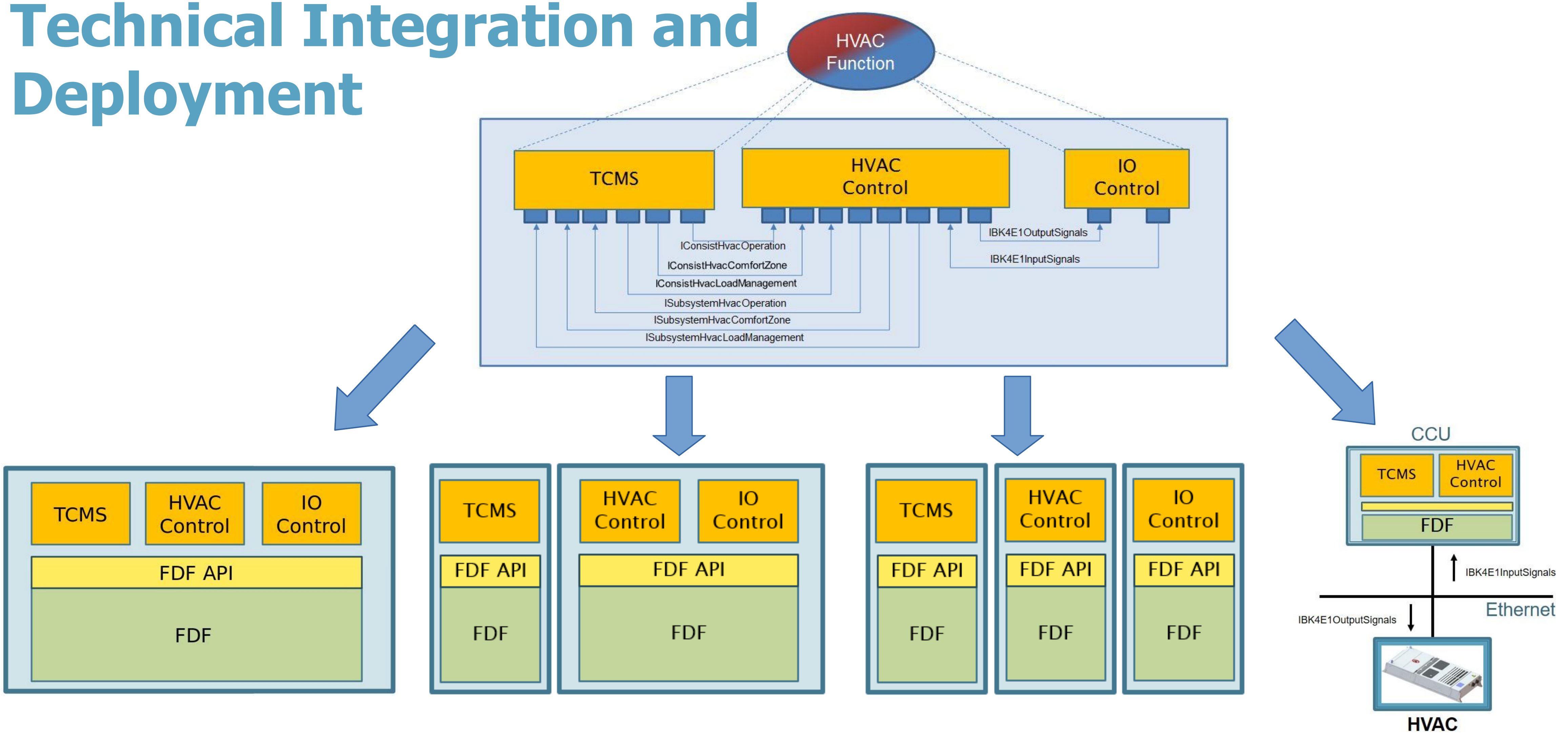
Data Structures for Application Profile DataTypes



impl_type_adjustmenttemperature.h
impl_type_comfortzonefailure.h
impl_type_czdehumidifycmd.h
impl_type_czhvacoperationalmodestatus.h
impl_type_czsetpointtemperature.h
impl_type_hvacfailurekind.h
impl_type_hvacoperationalmodestatus.h
impl_type_limitedperformancecmd.h
impl_type_reduceinternalpressurecmd.h
impl_type_startauthorization.h
impl_type_targettemperaturecheck.h
impl_type_airflowratecmd.h
impl_type_consistinsidetemperature.h
impl_type_czdehumidifycmdstatus.h
impl_type_czinsidetemperature.h
impl_type_datetime.h
impl_type_hvacfiredetection.h
impl_type_hvacoutsidesmokedetection.h
impl_type_neededpower.h
impl_type_specificfreshairratio.h
impl_type_startrequest.h
impl_type_tightnessactivationcmd.h

Implementation of data structures required by Integrity based FDF

Technical Integration and Deployment



Outline

1. Introduction
2. FDF
   - 2.1 Motivation
   - 2.2 Architecture and API
   - 2.3 Implementations
3. Application Profile
   - 3.1 Motivation
   - 3.2 Application Profile for HVAC
4. **Development and Integration**
   - 4.1 Interface Generation
   - 4.2 Integration of HVAC Application on FDF
5. **Summary**

Summary

- The FDF aims to have isolated but integrated applications instead of dedicated equipment (HW, SW, I/O) for each train function
- Takeaway:
  - Reduce number, diversity and complexity of equipment
  - Reduce **safety and certification** tasks and complexity
  - Reduce complexity of deployment for subsystem providers
  - Drastic reduction of obsolescence costs
  - Hardware and communication abstraction
  - Same application can run on different FDF implementations

Coordinator:

CAF, Igor López
✉ igor.lopez@caf.net
☎ +34 943 189 241

Coordinator:

IKERLAN, Aitor Arriola
✉ aarriola@ikerlan.es
☎ +34 943 712 400

The projects CONNECTA-2 and Safe4RAIL-2 have received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 826098 (CONNECTA-2) and No. 826073 (Safe4RAIL-2) respectively. The information and views set out in this document are those of the author(s) and do not necessarily reflect the official opinion of Shift2Rail Joint Undertaking. The JU does not guarantee the accuracy of the data included in this article. Neither the JU nor any person acting on the JU's behalf may be held responsible for the use which may be made of the information contained therein.