PMSF
IT Consulting

Open Simulation Interface (OSI/OSMP)

A Short Introduction to OSI/OSMP

ASAM OSI Crash Course
Webinar 2020-07-29

Who am I?

Pierre R. Mai, PMSF IT Consulting
pmai@pmsf.de

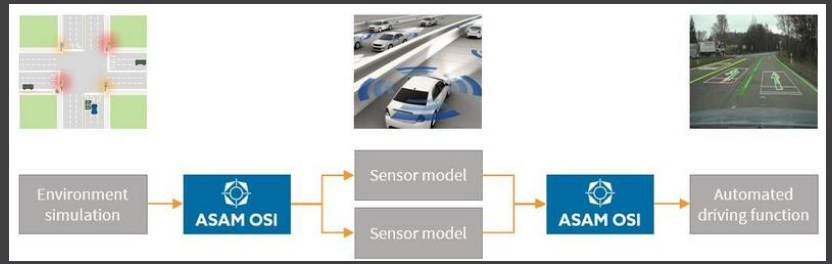
- MAP FMI Advisory Board Member
- MAP SSP Founding Member
- Pre-ASAM OSI CCB Member
- ASAM OSI CCB Member
  WG Lead Packaging & Performance

What is the Open Simulation Interface?

PMSF IT Consulting

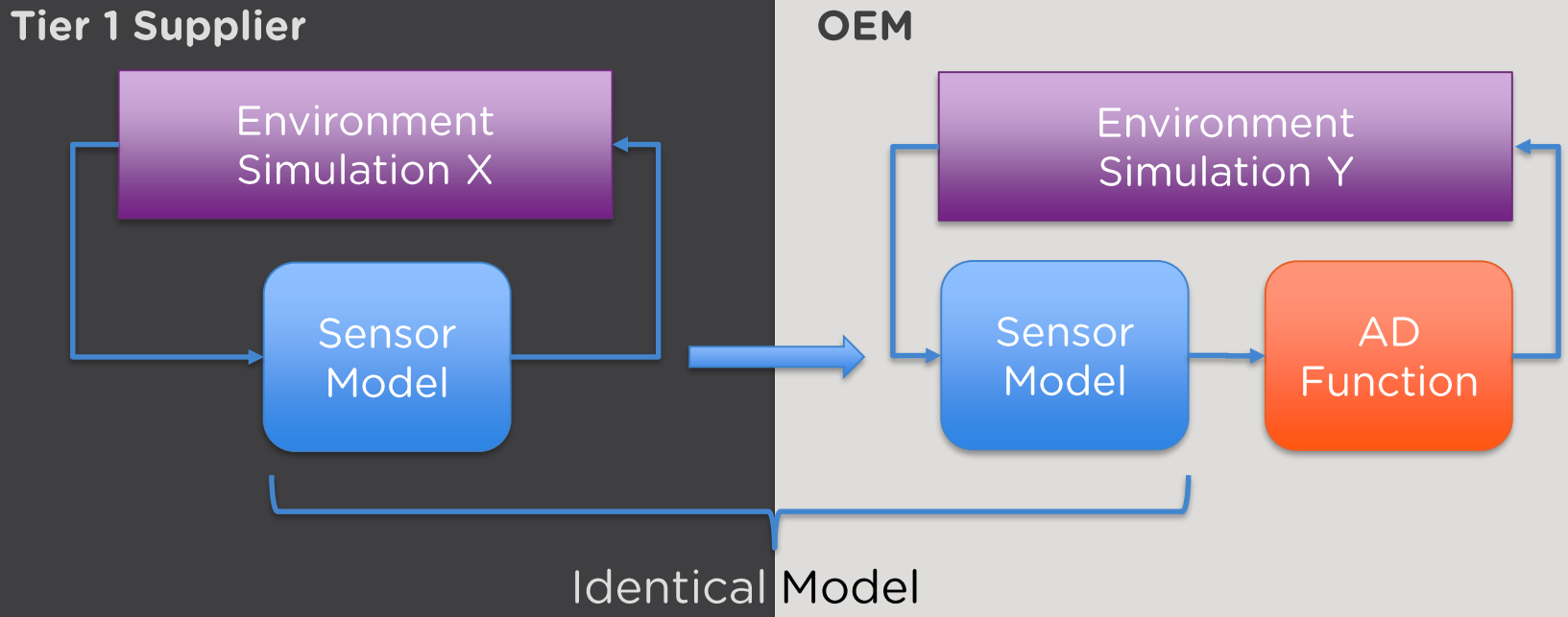
*"The Open Simulation Interface (OSI) defines a **generic interface** between **automated driving functions**, **driving simulation frameworks** and **sensor models**. Its long-term goal is to provide users the ability to connect any automated driving function to any driving simulator or sensor."*

Source: ASAM e.V. Press Release



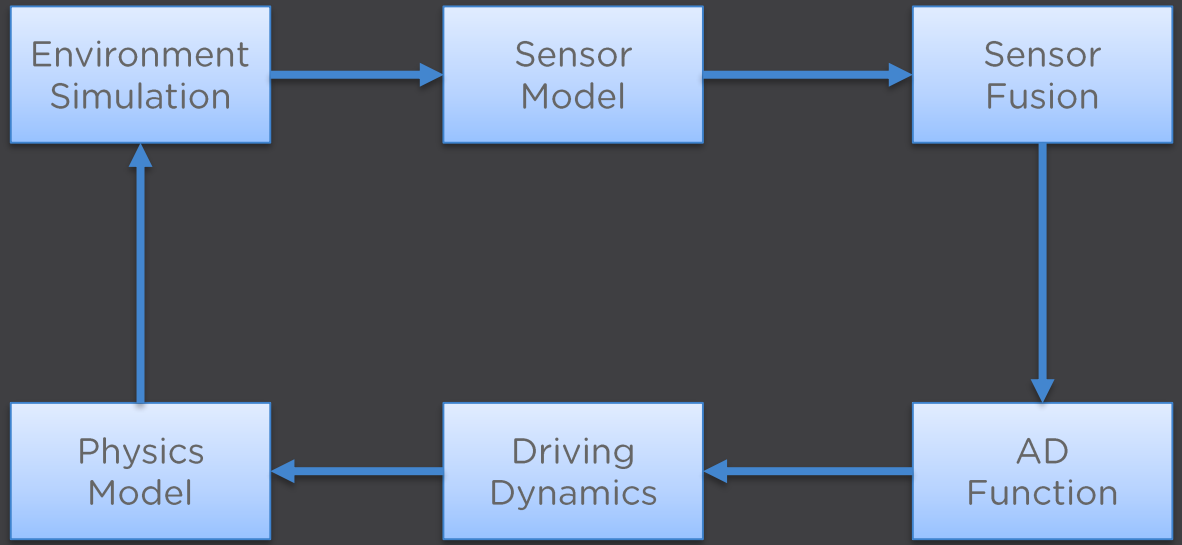
Original Motivation – One of

PMSF
IT Consulting



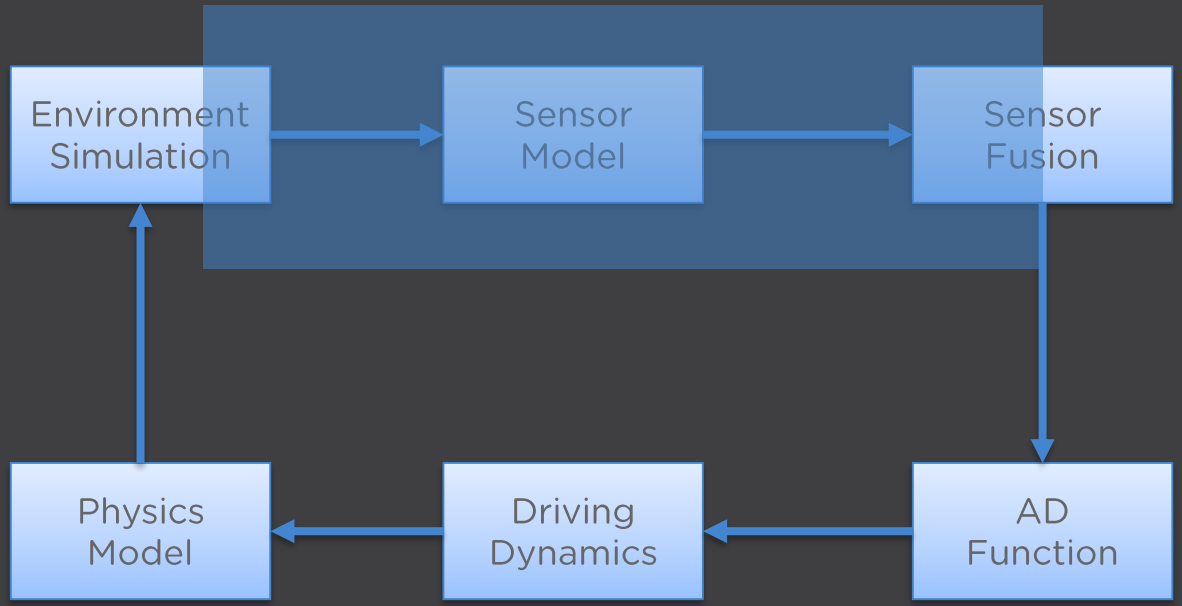
OSI Scope – Then and Now

PMSF
IT Consulting



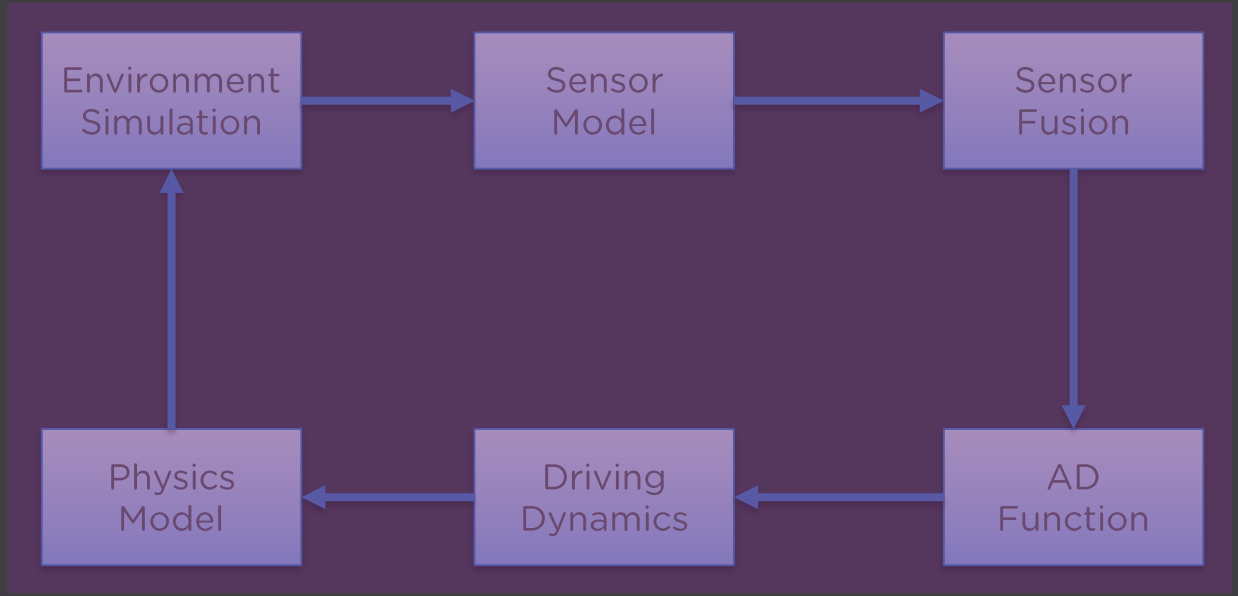
OSI Scope – Then and Now

PMSF
IT Consulting



OSI Scope – Then and Now

PMSF
IT Consulting



OSI Background

PMSF IT Consulting

- **Location:** https://github.com/OpenSimulationInterface/
- **Documentation:** https://opensimulationinterface.github.io/osi-documentation/
- **Open-source** (MPL 2.0)

- Initial Contributors: BMW (OSI v2), PMSF (OSMP)
- Developed further in conjunction with BMWi-supported **PEGASUS Project** (http://www.pegasus-projekt.info/) and follow-on projects:
  Main Contributors: AVL, Bosch, BMW, Continental ADC, Daimler, dSPACE, FZD, IPG, PMSF, Vires

- Used in various research and series code development projects

- **Moved to ASAM**, ASAM OSI Project Kick-Off Workshop on 2020-02-21
- Participation by all interested parties welcomed, open development process

OSI Introduction

PMSF
IT Consulting

- **Core rationale:**
  - **Get contents right**, then worry about technical implementation details
  - Seperate **interface definitions** from **middle-ware** as much as possible
  - **Flexible** simulation **architectures**, minimal invasiveness
  - **Ease of extension**, flexible forward/backward compatibility

- **Currently five sub-repositories:**
  - **open-simulation-interface:** OSI protocol buffer definitions (Data Layer)
  - **osi-sensor-model-packaging:** Packaging of models using OSI as FMUs
  - **osi-validation:** Scripts and tools to validate OSI messages
  - **osi-documentation:** Central documentation site
  - **osi-visualizer:** Open source visualization of OSI data

OSI Layering

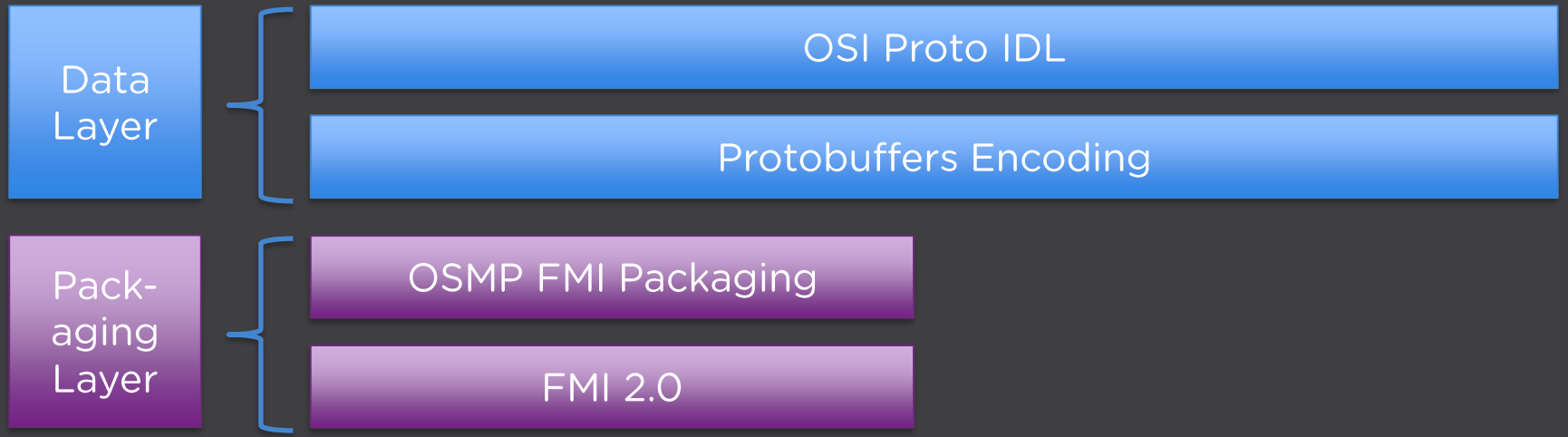
PMSF
IT Consulting

OSI Layering

PMSF
IT Consulting

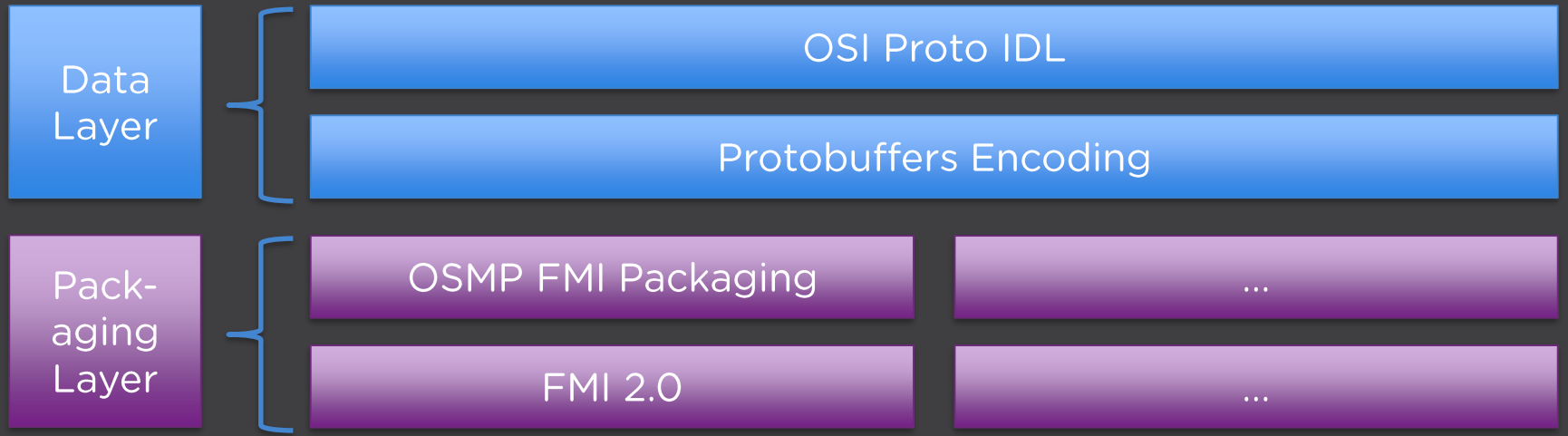
OSI Layering

PMSF
IT Consulting



OSI Layering

PMSF IT Consulting

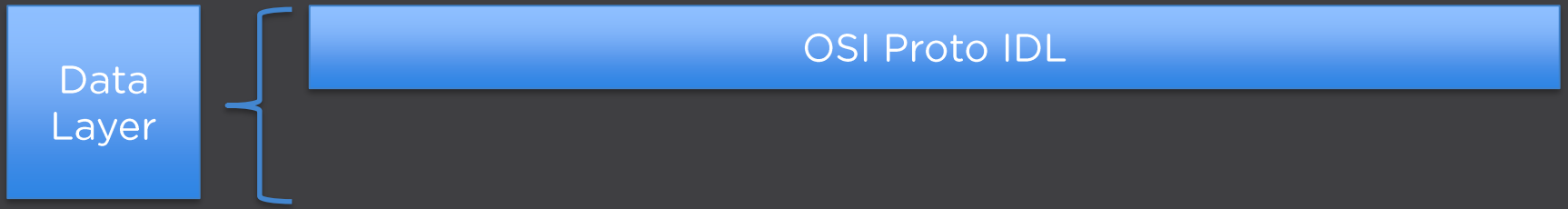


OSI Layering – Future Potential Directions

PMSF IT Consulting

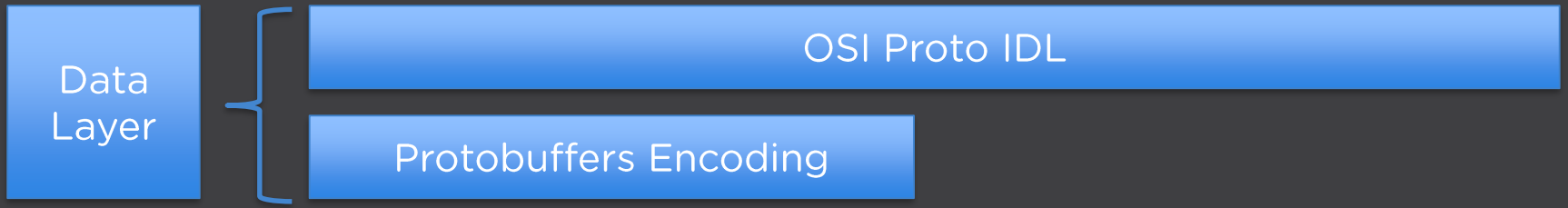
OSI Layering – Future Potential Directions

PMSF IT Consulting



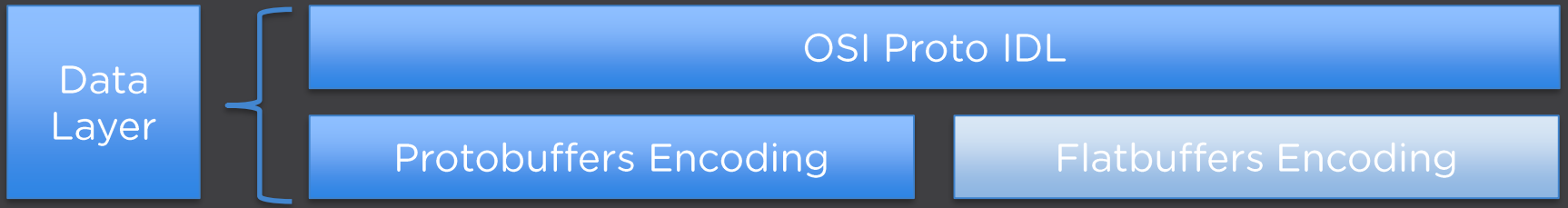
OSI Layering – Future Potential Directions

PMSF IT Consulting



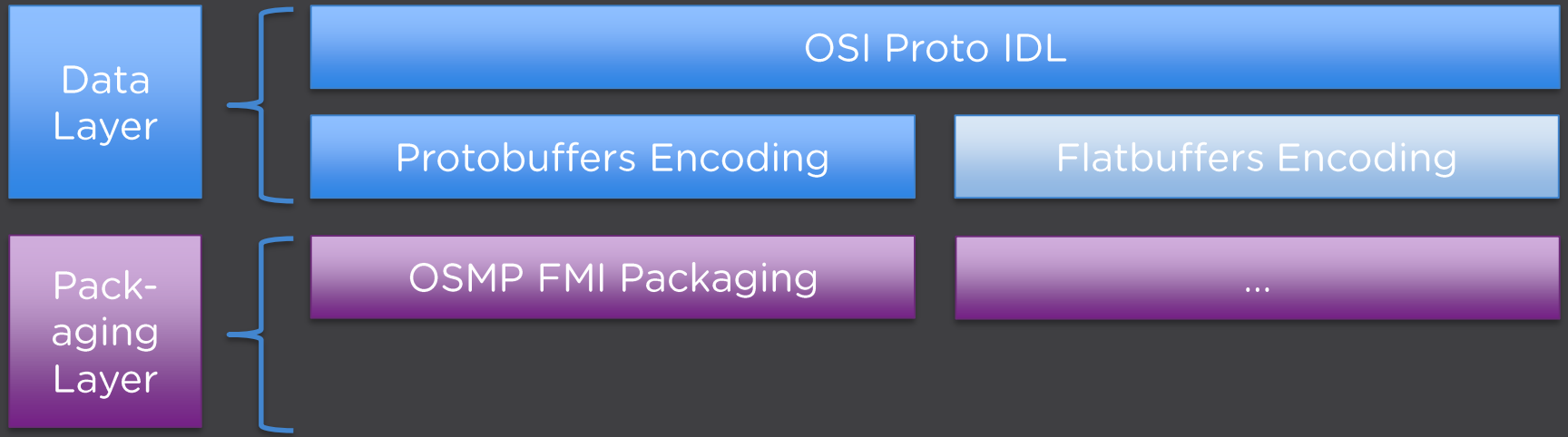
OSI Layering – Future Potential Directions

PMSF IT Consulting



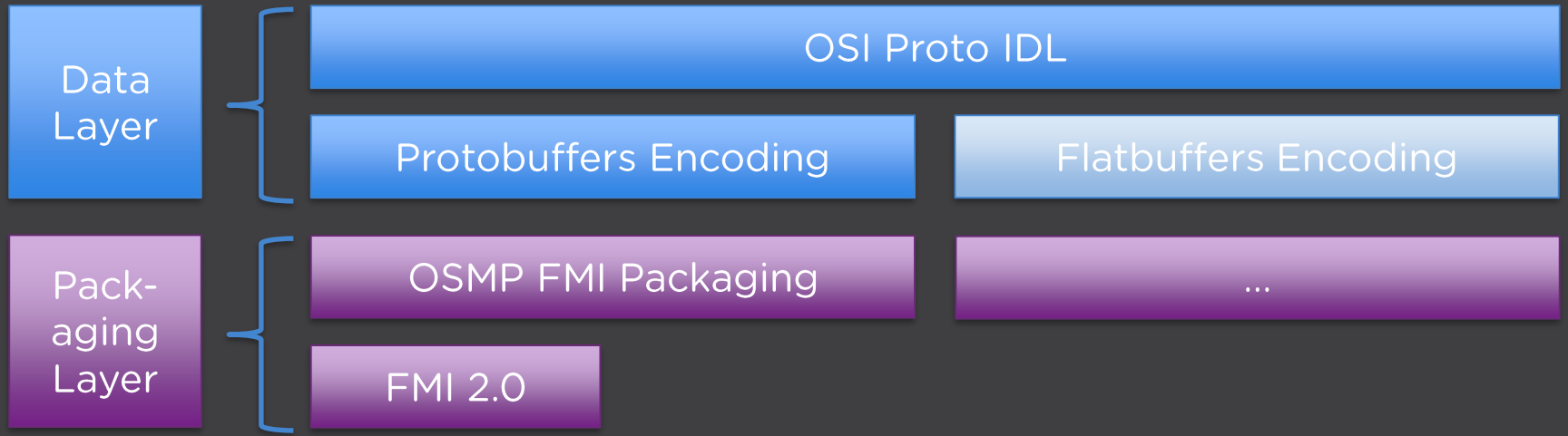
OSI Layering – Future Potential Directions

PMSF IT Consulting



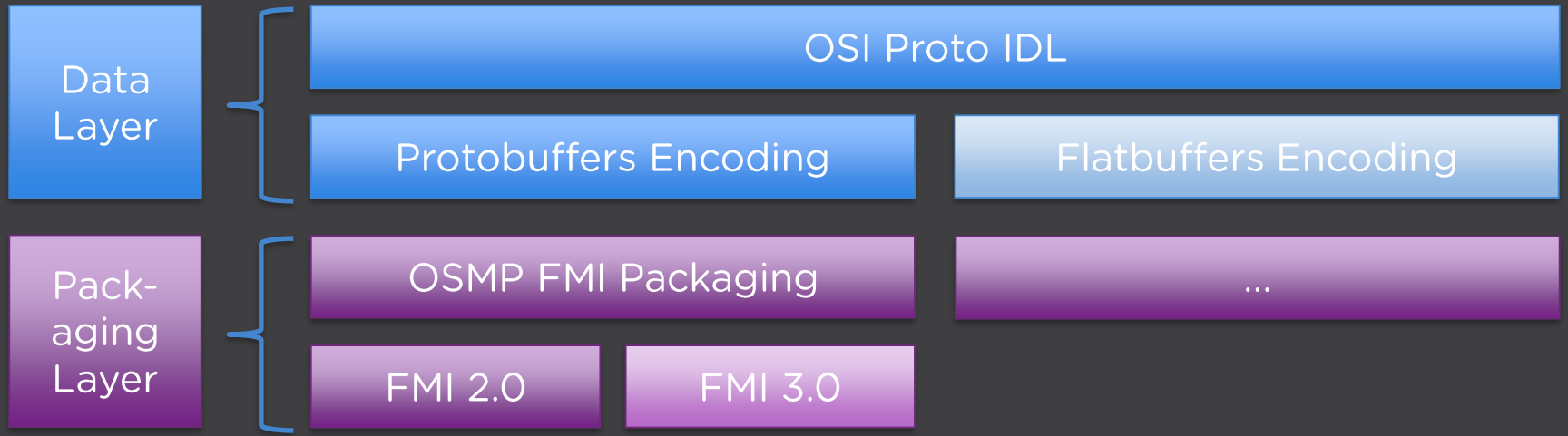
OSI Layering – Future Potential Directions

PMSF IT Consulting



OSI Layering – Future Potential Directions

PMSF IT Consulting



OSI Layering – Future Potential Directions

PMSF IT Consulting

OSI Data Layer

PMSF IT Consulting

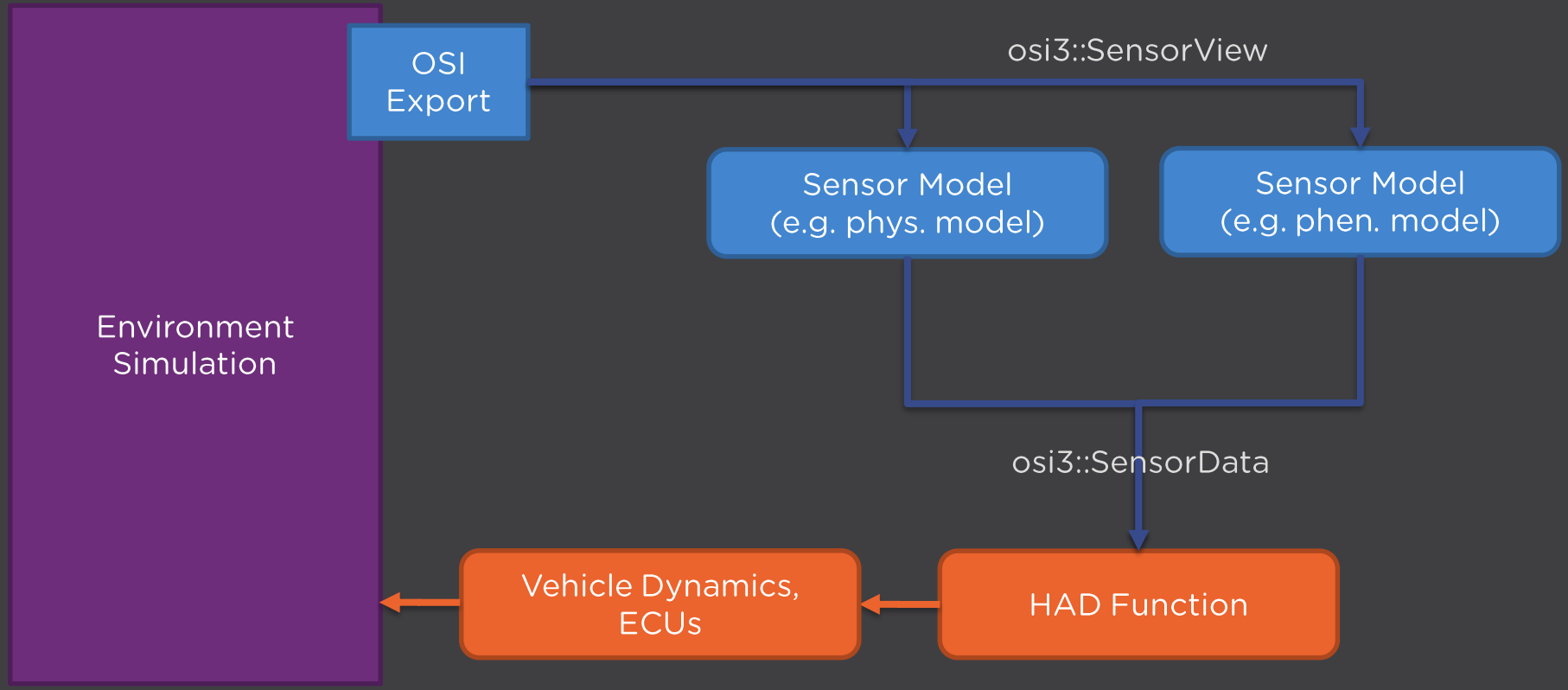
- Definitions for **ground truth**, **sensor input** and **sensor output**
- **In the future:** Additional non-sensor-related data, e.g. HMI, vehicle data, …
- Location: https://github.com/OpenSimulationInterface/open-simulation-interface

- Uses **Google protocol buffers** for data description and encoding

- **osi3::SensorView**:
  - Ground truth object lists
  - Optional sensor-specific data (e.g. camera images, LIDAR ray tracing data)
- **osi3::SensorData**:
  - Detected object lists
  - Feature data: Reflection lists, Point clouds
  - Optional copy of osi3::SensorView for traceability

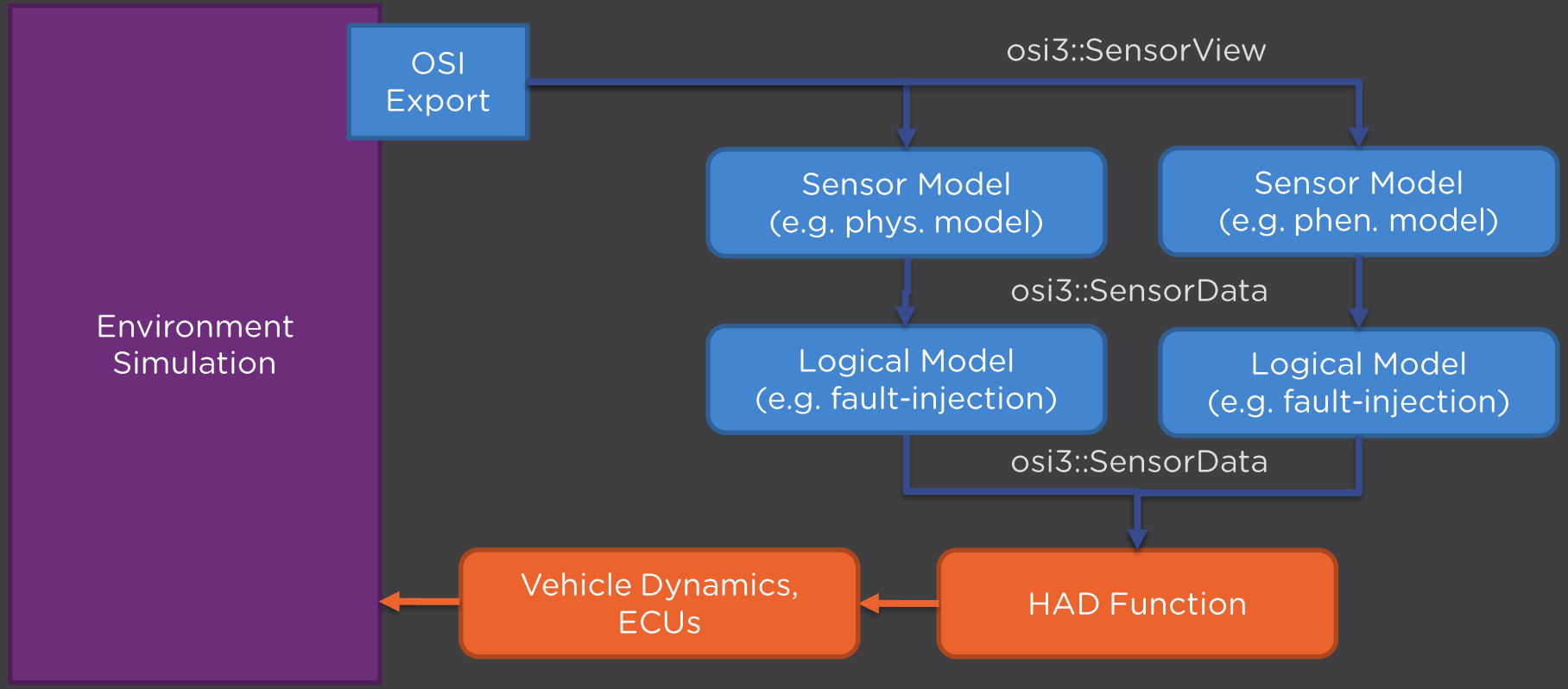
OSI Logical Data Flow

PMSF IT Consulting



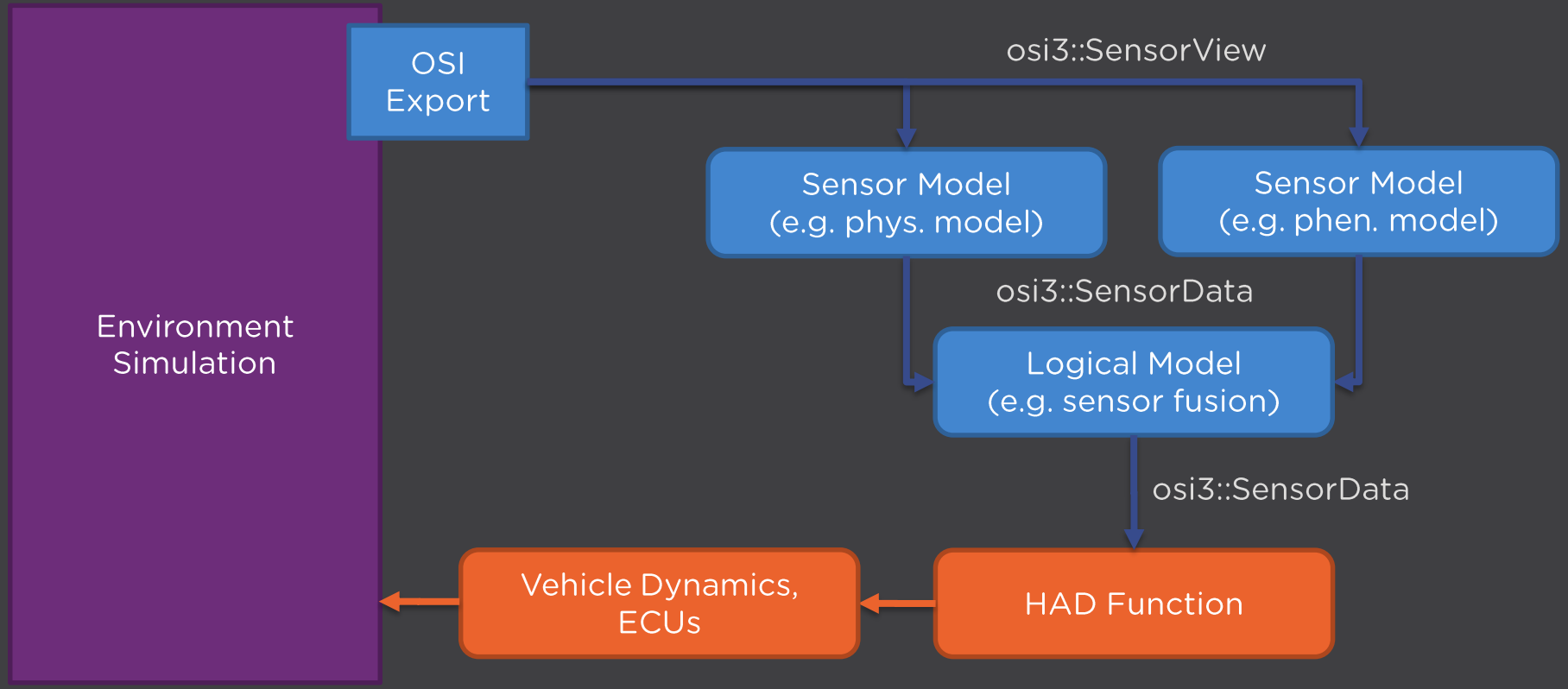
OSI Logical Data Flow (Fault-Injection)

PMSF IT Consulting



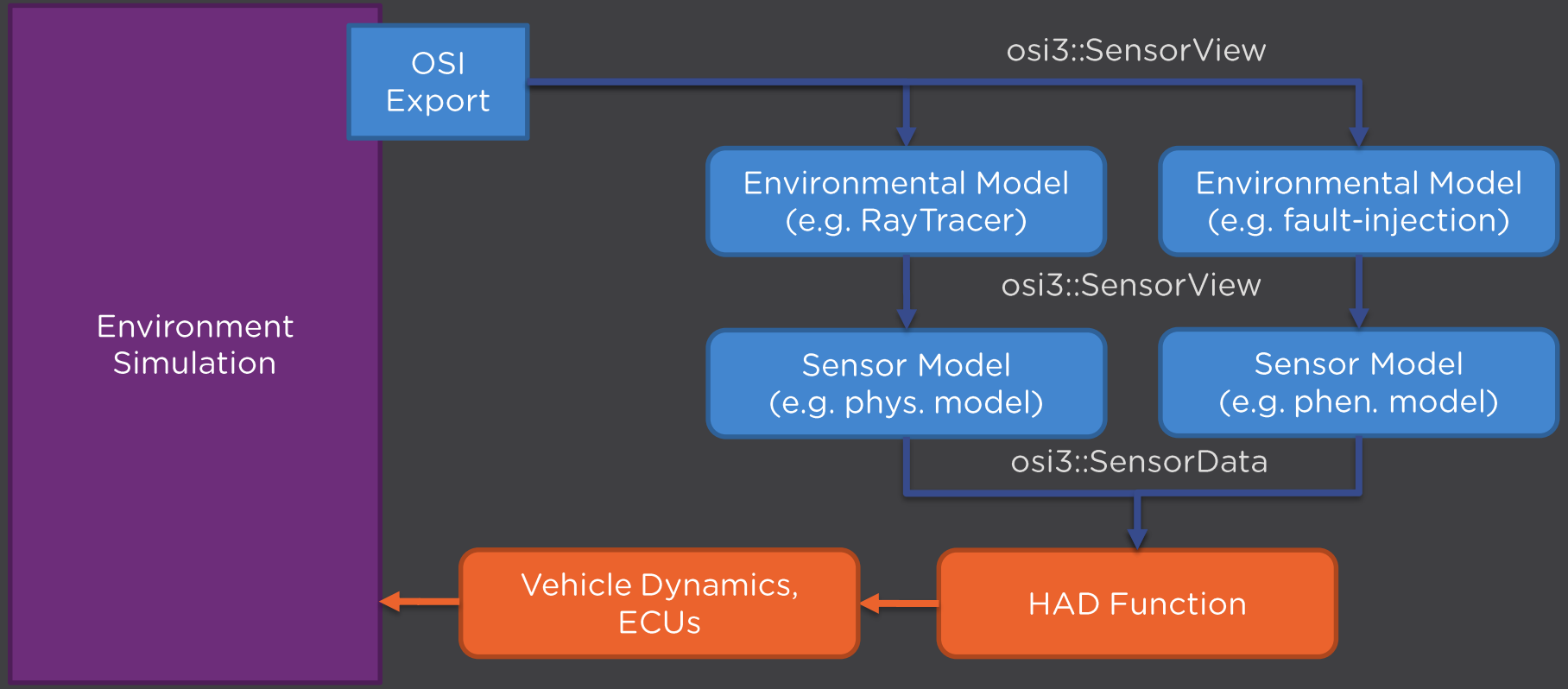
OSI Logical Data Flow (Sensor Fusion)

PMSF IT Consulting



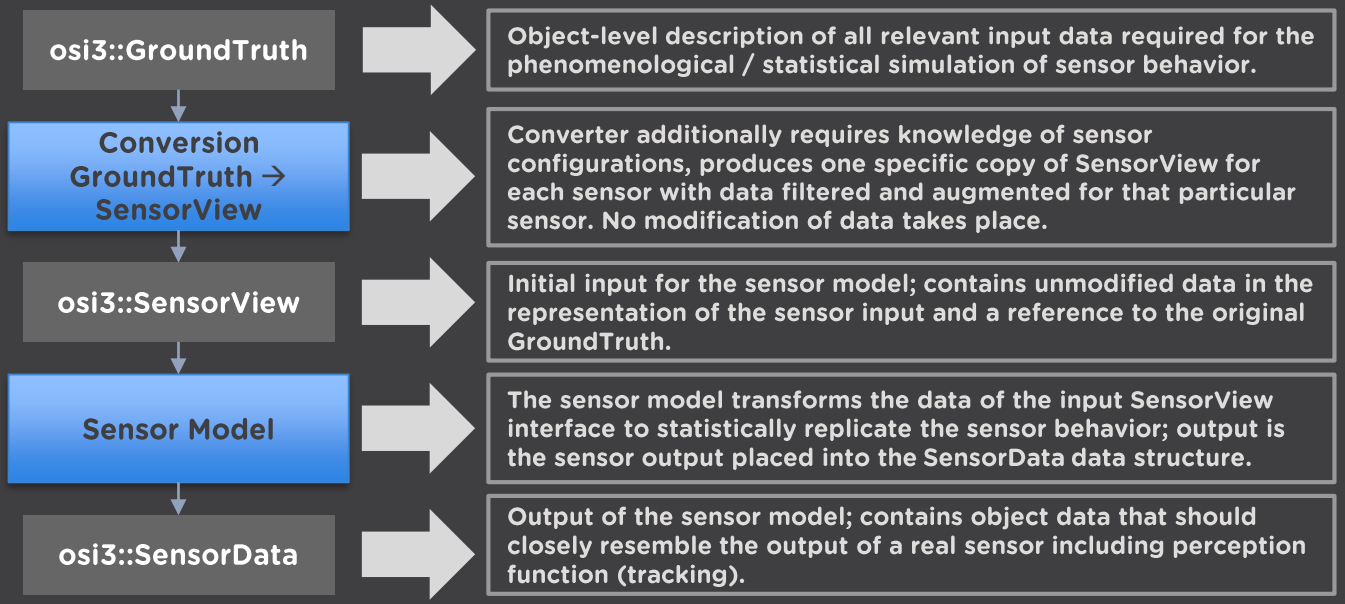
OSI Logical Data Flow (Augmented SensorView)

PMSF IT Consulting



OSI Logical Processing Flow

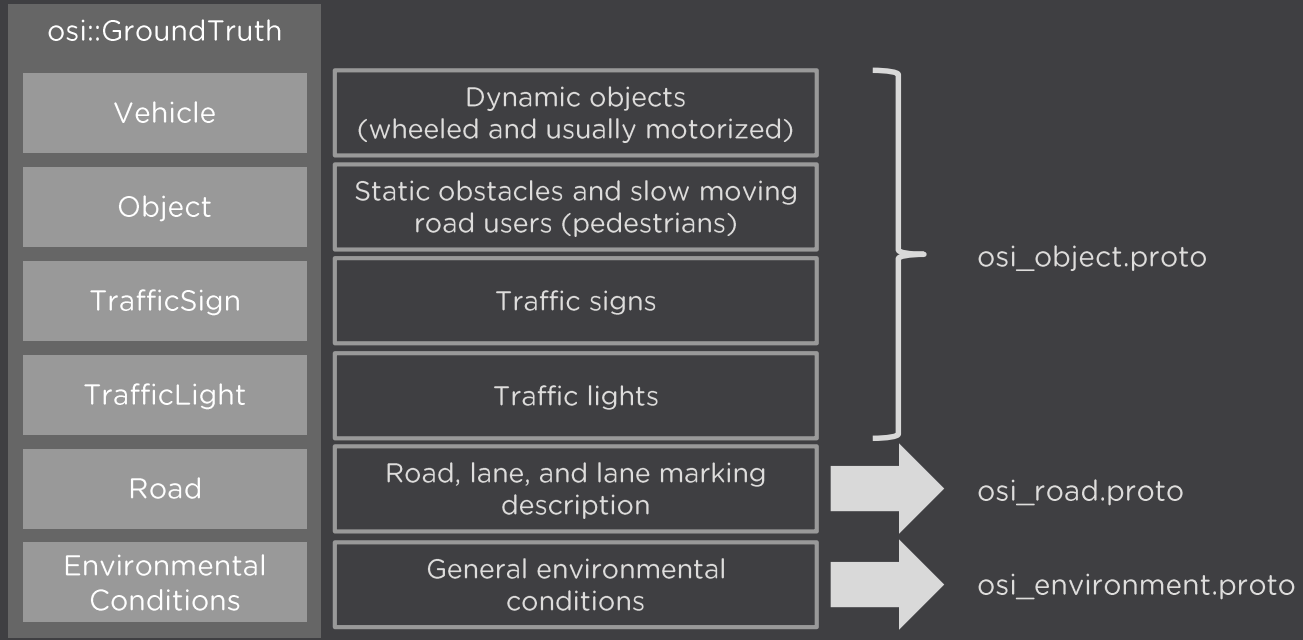
PMSF IT Consulting



osi3::GroundTruth Description

PMSF IT Consulting

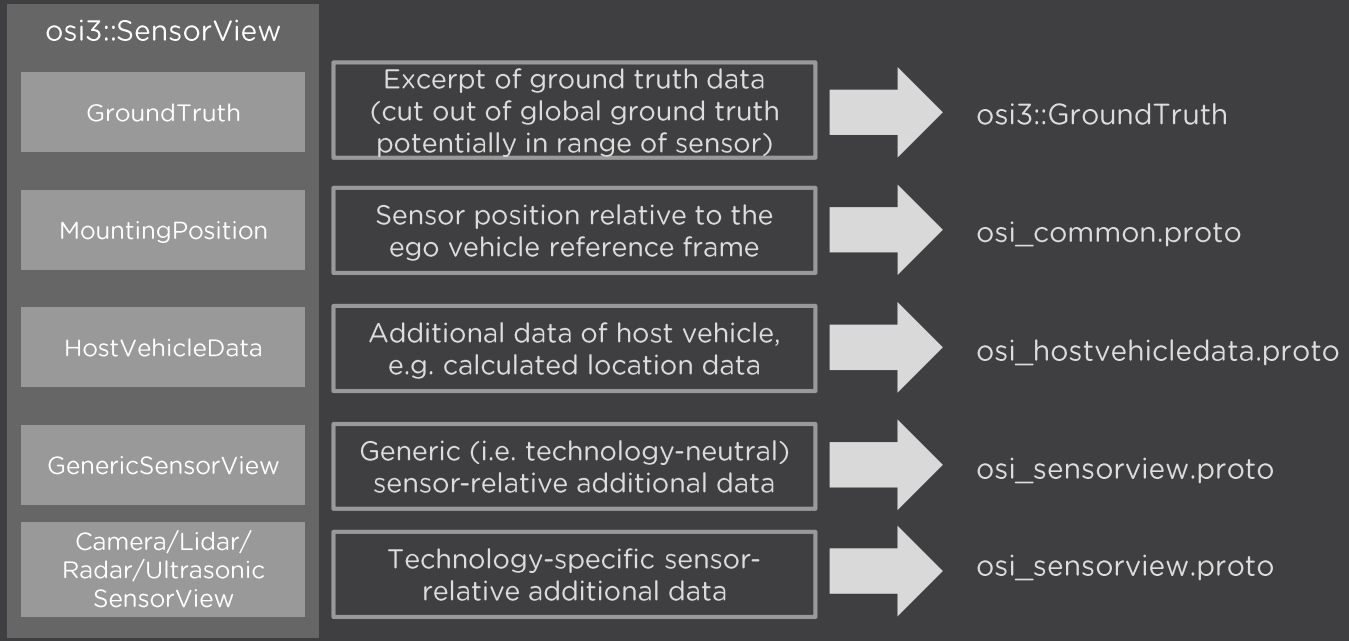
- **GroundTruth** contains unmodified object data describing the environment of the ego vehicle as required for phenomenological / statistical sensor models in world coordinates.



osi3::SensorView Description

PMSF IT Consulting

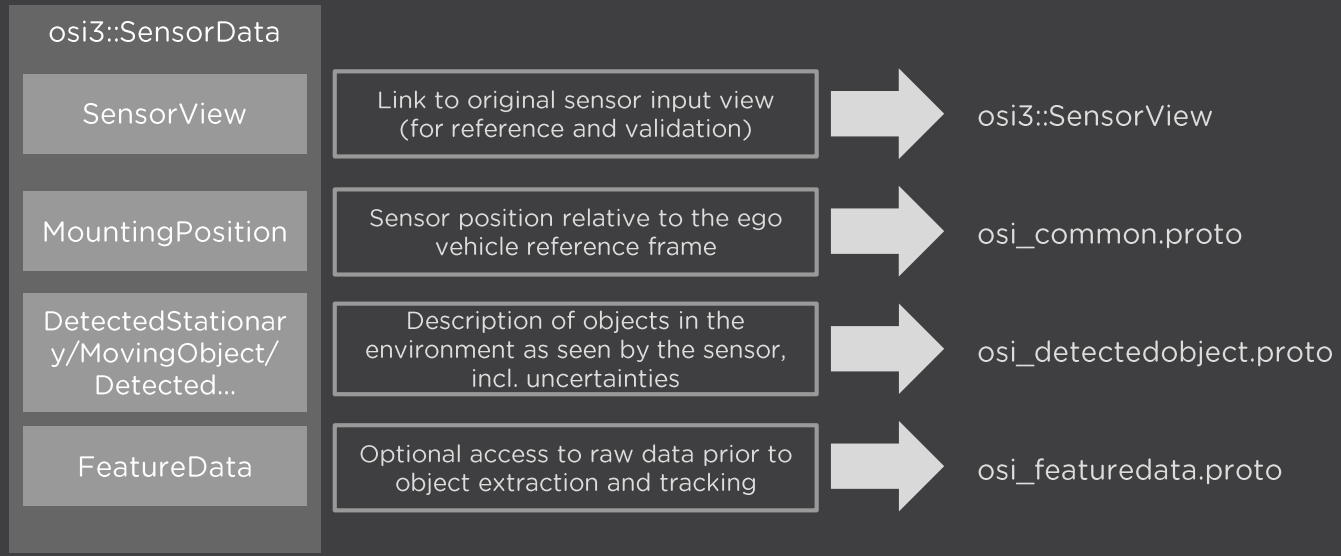
- **SensorView** describes the object data in the environment relative to one specific sensor, through an excerpt of the global ground truth and any sensor-relative additional data, like e.g. ray-tracing data, rendered camera images, …



osi3::SensorData Description

PMSF IT Consulting

- **SensorData** describes the detected object data in the environment relative to one (or more) specific sensors, i.e. it is the output of a sensor model after sensing, object extraction and tracking.



OSI Sensor Model Packaging (OSMP)

PMSF IT Consulting

- **Packaging of Models:** OSI Sensor Model Packaging (**OSMP**)
- **Location:** https://github.com/OpenSimulationInterface/osi-sensor-model-packaging

- **Base Standard** for Packaging: **FMI 2.0 Co-Simulation FMUs**

- **Extends FMI** in **minimal ways** to allow:
  - Efficient exchange of OSI data
  - Automatic connections in model setup/integration
  - Automatic configuration of environment simulation based on sensor model configuration data

OSI Sensor Model Packaging (OSMP)

PMSF IT Consulting

- FMI packaging deals with common integration issues:
  Co-Simulation semantics, init, parametrization, source/binary packaging, …
- Leverage current FMI support, related projects:
  - MAP SSP project allows exchange of fully connected systems of FMUs (including their parameters and parametrization, other resources)
  - FMI support present in all relevant platforms already, can be leveraged
  - By careful definition simulation in FMI environments is possible
  - Support for distributed simulation (e.g. through RFMI protocol, DCP)
- Handling of complex, dynamic data (e.g. object lists, raw data, video, …) not very practical with FMI 2.0, solution for FMI 3.0: Binary Types
- No common definitions for forward/backward-compatibility of FMI interfaces

- Basic Idea: Pass OSI data as ProtoBuffers (potentially: FlatBuffers/DDS-XType)

OSMP Kinds of Models

PMSF IT Consulting

The current specification supports the following kinds of models, that can be packaged as FMUs:

- **Environmental effect models:**
  Consume **osi3::SensorView** as input and produce **osi3::SensorView** as output
- **Sensor models:**
  Consume **osi3::SensorView** as input and generate **osi3::SensorData** as output
- **Logical models:**
  Consume **osi3::SensorData** as input and produce **osi3::SensorData** as output
  (e.g. sensor fusion models, fault-injection models, etc.)

Additionally **complex models** that combine various aspects of the model kinds above are possible, however configuration and setup of such FMUs will require manual intervention.

OSMP Example

PMSF IT Consulting

```
<?xml version="1.0" encoding="ISO-8859-1"?>
- <fmiModelDescription variableNamingConvention="structured" generationDateAndTime="2019-03-07T23:45:23Z" generationTool="PMSF Manual FMU Framework" version="2.0" author="PMSF" description="Demonstration C++ Sensor FMU for OSI Sensor Model Packaging" guid="bbd7b2bd251f430797b8f36208d20c4b" modelName="OSMP Dummy Sensor FMU" fmiVersion="2.0">
  - <CoSimulation canNotUseMemoryManagementFunctions="true" canHandleVariableCommunicationStepSize="true" modelIdentifier="OSMPDummySensor">
    - <SourceFiles>
        <File name="OSMPDummySensor.cpp"/>
      </SourceFiles>
    </CoSimulation>
  - <LogCategories>
      <Category description="Enable logging of all FMI calls" name="FMI"/>
      <Category description="Enable OSMP-related logging" name="OSMP"/>
      <Category description="Enable OSI-related logging" name="OSI"/>
    </LogCategories>
    <DefaultExperiment stepSize="0.020" startTime="0.0"/>
  - <VendorAnnotations>
    - <Tool name="net.pmsf.osmp" xmlns:osmp="http://xsd.pmsf.net/OSISensorModelPackaging">
        <osmp:osmp version="1.0.0" osi-version="3.0.0"/>
      </Tool>
    </VendorAnnotations>
  - <ModelVariables>
    - <ScalarVariable name="OSMPSensorViewIn.base.lo" variability="discrete" causality="input" valueReference="0">
        <Integer start="0"/>
      - <Annotations>
        - <Tool name="net.pmsf.osmp" xmlns:osmp="http://xsd.pmsf.net/OSISensorModelPackaging">
            <osmp:osmp-binary-variable role="base.lo" name="OSMPSensorViewIn" mime-type="application/x-open-simulation-interface; type=SensorView; version=3.0.0"/>
          </Tool>
        </Annotations>
      </ScalarVariable>
    - <ScalarVariable name="OSMPSensorViewIn.base.hi" variability="discrete" causality="input" valueReference="1">
        <Integer start="0"/>
      - <Annotations>
        - <Tool name="net.pmsf.osmp" xmlns:osmp="http://xsd.pmsf.net/OSISensorModelPackaging">
            <osmp:osmp-binary-variable role="base.hi" name="OSMPSensorViewIn" mime-type="application/x-open-simulation-interface; type=SensorView; version=3.0.0"/>
          </Tool>
        </Annotations>
      </ScalarVariable>
    - <ScalarVariable name="OSMPSensorViewIn.size" variability="discrete" causality="input" valueReference="2">
        <Integer start="0"/>
      - <Annotations>
        - <Tool name="net.pmsf.osmp" xmlns:osmp="http://xsd.pmsf.net/OSISensorModelPackaging">
            <osmp:osmp-binary-variable role="size" name="OSMPSensorViewIn" mime-type="application/x-open-simulation-interface; type=SensorView; version=3.0.0"/>
          </Tool>
        </Annotations>
      </ScalarVariable>
    - <ScalarVariable name="OSMPSensorDataOut.base.lo" variability="discrete" causality="output" valueReference="3" initial="exact">
        <Integer start="0"/>
      - <Annotations>
        - <Tool name="net.pmsf.osmp" xmlns:osmp="http://xsd.pmsf.net/OSISensorModelPackaging">
            <osmp:osmp-binary-variable role="base.lo" name="OSMPSensorDataOut" mime-type="application/x-open-simulation-interface; type=SensorData; version=3.0.0"/>
          </Tool>
        </Annotations>
      </ScalarVariable>
      ...
```

OSMP Top Level Information

PMSF IT Consulting

```
  <DefaultExperiment stepSize="0.020" startTime="0.0"/>
- <VendorAnnotations>
  - <Tool name="net.pmsf.osmp" xmlns:osmp="http://xsd.pmsf.net/OSISensorModelPackaging">
        <osmp:osmp version="1.0.0" osi-version="3.0.0"/>
    </Tool>
  </VendorAnnotations>
```

- Default Experiment gives suitable sensor model communication rate (stepSize)
- Vendor Annotation marks FMU as OSMP compliant and gives default OSI version for OSI interfaces
- Naming must be structured naming

OSMP SensorView Inputs

PMSF IT Consulting

- Sensor view inputs MUST be named with the prefix **OSMPSensorViewIn**. If more than one sensor view input is to be configured, the prefix MUST be extended by an array index designator, i.e. two inputs will use the prefixes **OSMPSensorViewIn[1]** and **OSMPSensorViewIn[2]**.
  - <prefix>.base.lo (Discrete Integer Input): 32-bit LSB of OSI ProtoBuffer Pointer
  - <prefix>.base.hi (Discrete Integer Input): 32-bit MSB of OSI ProtoBuffer Pointer (64bit only)
  - <prefix>.size (Discrete Integer Input): Size of OSI ProtoBuffer Pointer

- The *guaranteed lifetime* of the sensor view protocol buffer pointer provided as input to the FMU MUST be from the time of the call to **fmi2SetInteger** that provides those values until *the end of the following* **fmi2DoStep** call, i.e. the sensor model can rely on the provided buffer remaining valid from the moment it is passed in until the end of the corresponding calculation, and thus does not need to copy the contents in that case (zero copy input).

OSMP Auto Configuration

PMSF IT Consulting

Auto configuration (e.g. Sensor Resolution, Range, ...) for SensorView Inputs is made available via a pair of calculatedParameter and Parameter notional OSMP Binary Variables transporting config data in ProtoBuffer format:

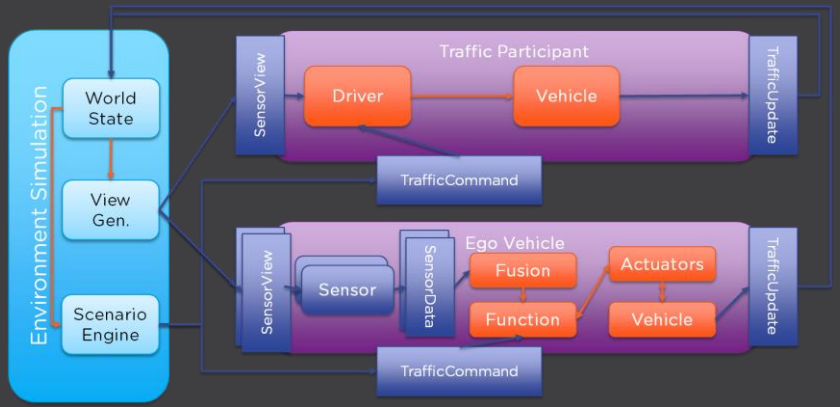
- **OSMPSensorViewInConfigRequest (calculatedParameter):**
  This variable provides the sensor view configuration that the sensor requests as a **osi3::SensorViewConfiguration** message. It can be enquired by the simulation environment prior to setting the actual configuration.

- **OSMPSensorViewInConfig (parameter):**
  This variable is set by the simulation environment after it has retrieved the requested configuration from **OSMPSensorViewInConfigRequest**, taking into account both the requested and supported configuration. It is supplied as a **osi3::SensorViewConfiguration** message. If the supplied configuration is not satisfactory, the FMU can produce an error, aborting the simulation.

Outlook: OSI Beyond Sensors (-> SETLevel4To5)

PMSF IT Consulting

- New **Traffic Participant** OSMP Model Type:
  - Inputs:
    - 1..n osi3::SensorViews
    - 0..1 osi3::TrafficCommand
  - Outputs:
    - 1 osi3::TrafficUpdate
  - Covers both traffic agents and EGO vehicles



- osi3::TrafficUpdate: State update to environment simulation
- osi3::TrafficCommand: Control of traffic participant by scenario engine

OSI Validation Project

PMSF IT Consulting

- GitHub: https://github.com/OpenSimulationInterface/osi-validation
  Documentation: https://opensimulationinterface.github.io/osi-documentation/osi-validation/README.html

OSI Validator

build passing

OSI Validator checks the compliance of OSI messages with predefined rules. The full documentation on the validator and customization of the rules is available here.

Usage

```
usage: osivalidator [-h] [--rules RULES]
                    [--type {SensorView,GroundTruth,SensorData}]
                    [--output OUTPUT] [--timesteps TIMESTEPS] [--debug]
                    [--verbose] [--parallel] [--format {separated,None}]
                    [--blast BLAST] [--buffer BUFFER]
                    data
```

OSI Validator: Theory of Operation

PMSF IT Consulting

OSI Validator: Theory of Operation

PMSF
IT Consulting

OSI Validator: Theory of Operation

PMSF IT Consulting

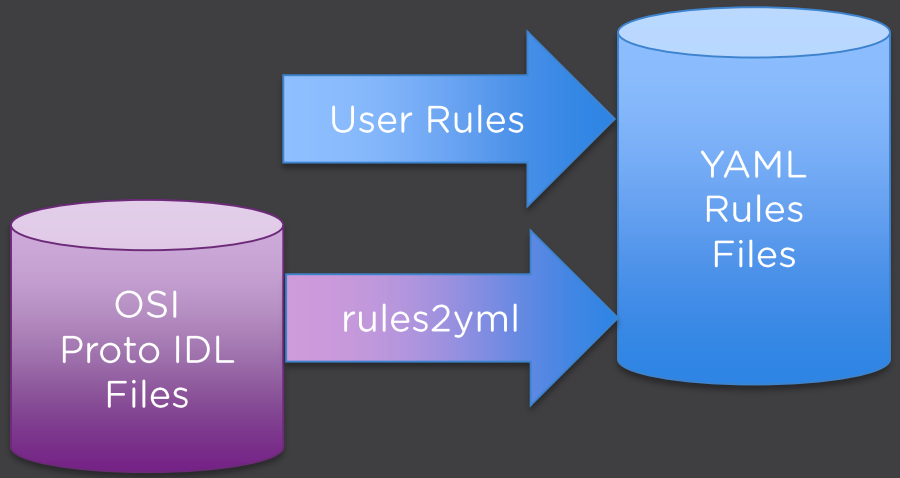
```
// Specific information about the vehicle.
//
// \note This field is mandatory if the \c #type is
// #TYPE_VEHICLE .
//
// \rules
// check_if this.type is_equal_to 2 else do_check is_set
// \endrules
//
optional VehicleAttributes vehicle_attributes = 5;
```

OSI Validator: Theory of Operation

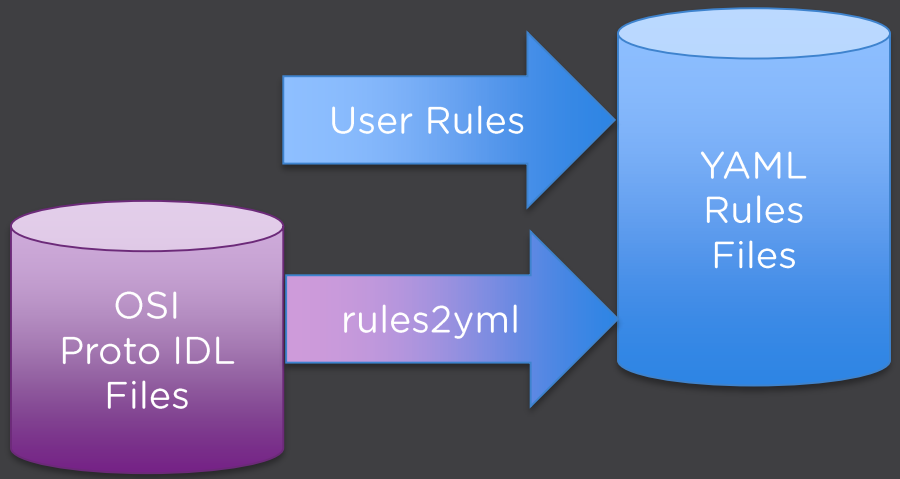
PMSF
IT Consulting

OSI Validator: Theory of Operation

PMSF
IT Consulting



OSI Validator: Theory of Operation

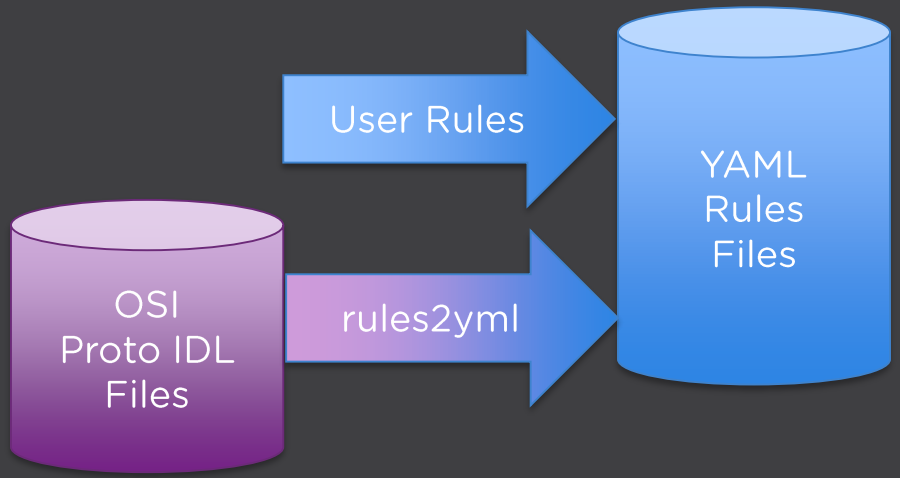



```yaml
MovingObject:
  id:
    - is_globally_unique:
  base:
  type:
  assigned_lane_id:
  vehicle_attributes:
    - check_if:
      - is_equal_to: 2
        target: this.type
      do_check:
      - is_set:
```

PMSF
IT Consulting

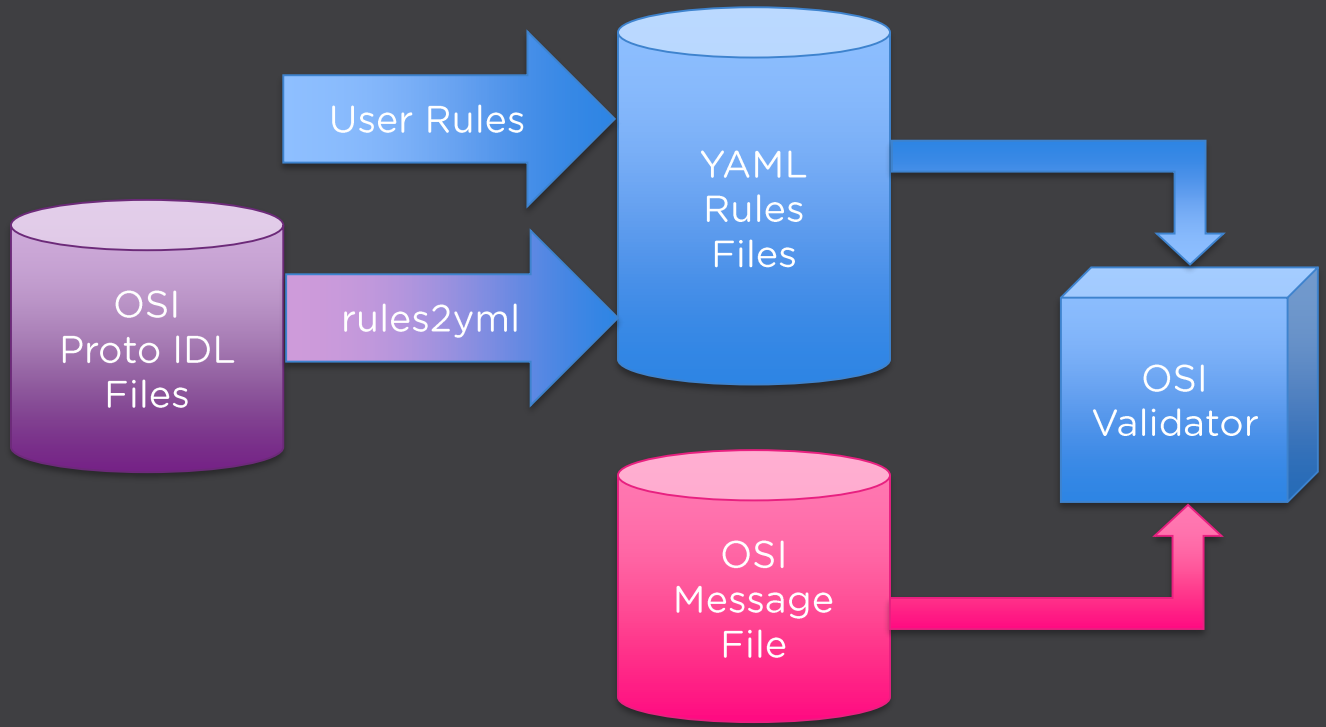
OSI Validator: Theory of Operation

PMSF
IT Consulting



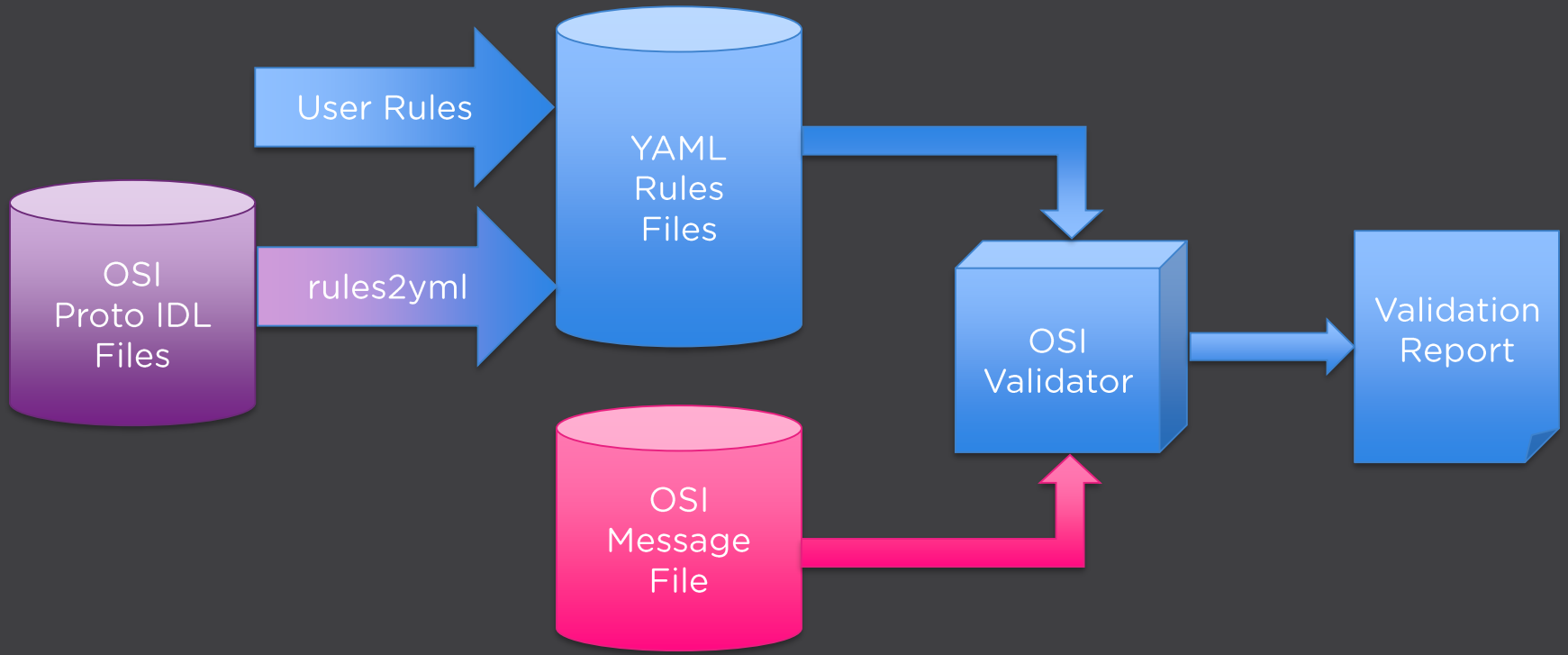
OSI Validator: Theory of Operation

PMSF IT Consulting



OSI Validator: Theory of Operation

PMSF IT Consulting



OSI Validator Notation (Rules)

PMSF IT Consulting

- YAML Files

```yaml
StationaryObject:      ainar, a year ago • First r
  id:
    - is_globally_unique:
  base:
  classification:
  model_reference:
  Classification:
    type:
    material:
    density:
    color:
MovingObject:
  id:
    - is_globally_unique:
  base:
  type:
  assigned_lane_id:
  vehicle_attributes:
    - check_if:
      - is_equal_to: 2
        target: this.type
      do_check:
      - is_set:
```

- Per message ruleset
- Rules for each field:
  - is_set
  - is_globally_unique
  - refers_to
  - is_equal_to / is_different_to
  - is_less_than / is_greater_than
  - is_iso_country_code
  - Conditionals: check_if
- Rules can be user-defined (Python)
- OSI embedded rules can be converted (rules2yml.py)
- Embeddable in SSP (annotations)

OSI Visualizer Project

PMSF IT Consulting

- GitHub: https://github.com/OpenSimulationInterface/osi-visualizer
  Documentation: https://opensimulationinterface.github.io/osi-documentation/osi-visualizer/README.html

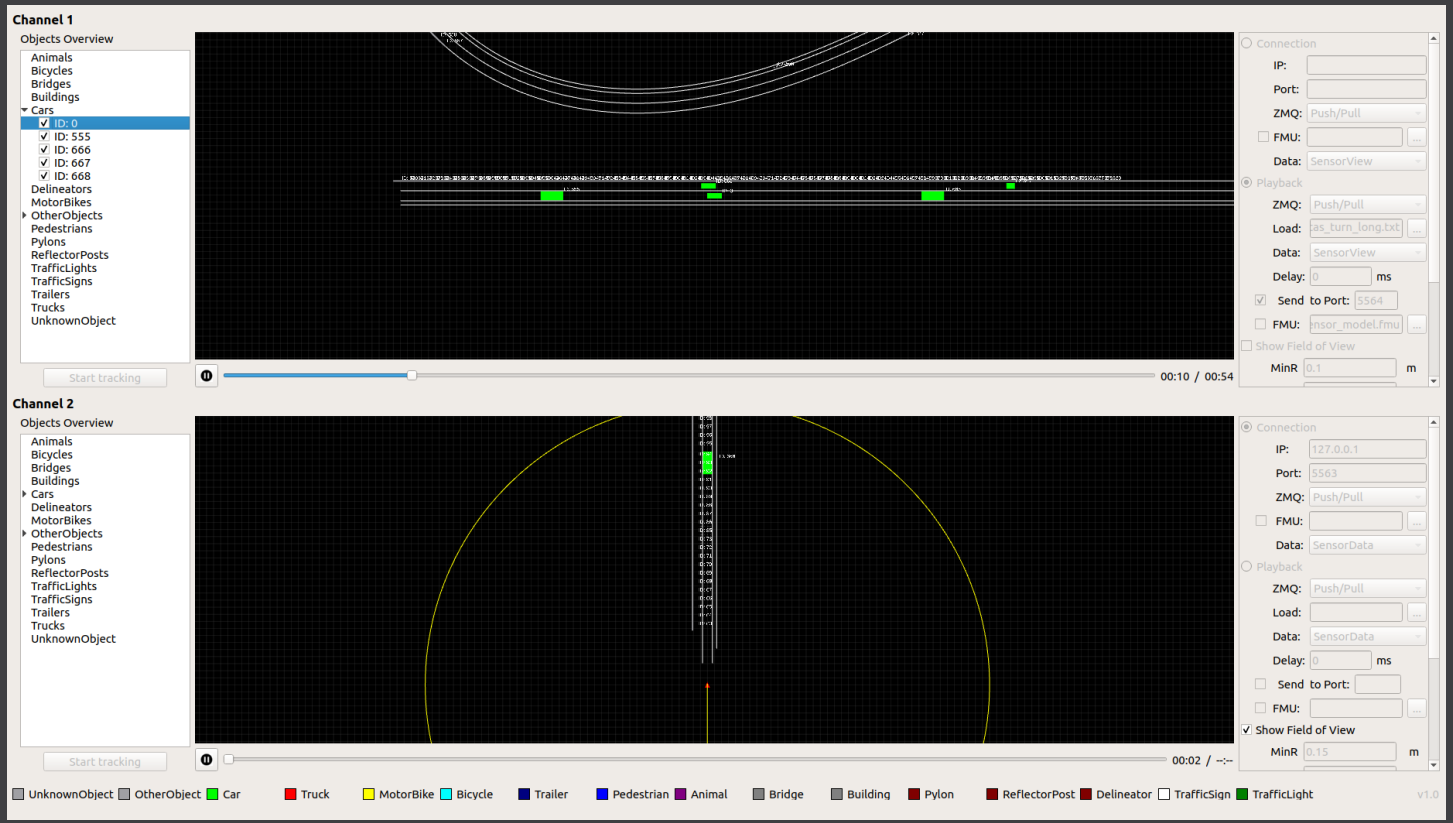
OSI Visualizer

build passing

OSI Visualizer serves as a visualization tool for the current implementation of OSI (Open Simulation Interface) messages. It supports `GroundTruth`, `SensorView` and `SensorData` messages and allows the visualization of two independent data channels using different input types (file and network stream). For more information see the documentation here.

OSI Visualizer

PMSF
IT Consulting



Questions?

PMSF
IT Consulting

Pierre R. Mai
Owner / Director, PMSF IT Consulting

Phone: +49 8161 976 96 - 11
Email: pmai@pmsf.de