# Contents

1. Introduction 11
2. Acronyms and Abbreviations 12
3. Related documentation 13
   1. Input documents & related standards and norms 13
   2. Further applicable specification 13
4. [Constraints and assumptions](#_bookmark0) 14
   1. [Limitations](#_bookmark1) 14
      1. [ISO Optionals](#_bookmark2) 14
      2. [Sensor Control Interfaces](#_bookmark3) 14
      3. [Sensor Capabilities](#_bookmark4) 14
   2. [Applicability to car domains](#_bookmark5) 14
5. [Dependencies to other modules](#_bookmark6) 15
6. Requirements Tracing 16
7. [Functional Specification](#_bookmark7) 36
   1. [Outline](#_bookmark8) 36
      1. [Goals and scope](#_bookmark9) 36
      2. [Relation to other standards](#_bookmark10) 38
   2. [AD Sensor Service Design](#_bookmark11) 39
      1. [ISO mapping to Sensor Services](#_bookmark12) 39
         1. [Sensor Types](#_bookmark13) 39
         2. [Sensor Data level](#_bookmark14) 39
         3. [Supportive Sensor interfaces](#_bookmark15) 40
      2. [Service Optional Elements](#_bookmark16) 40
8. [API specification](#_bookmark17) 42
9. Service Interfaces 43
   1. Type definitions 43
      1. General Header Definition 43
         1. CapabilityVector 43
         2. InterfaceVersionId 43
         3. InterfaceCycleTimeVariation 44
         4. InterfaceId 44
         5. DataQualifier 45
         6. RecognizedStatus 45
         7. TrackingMotionModel 46
         8. MotionType 46
         9. ColourModelType 46
         10. RadialVelocityAmbiguityDomain 47
         11. RangeAmbiguityDomain 47
         12. AngleAzimuthAmbiguityDomain 48
         13. AngleElevationAmbiguityDomain 48
         14. InterfaceApplicability 48
         15. VehicleCoordinateSystemType 49
         16. Point3D 49
         17. Point3DError 50
         18. Orientation3D 50
         19. Orientation3DError 50
         20. Point3DErrorVector 51
         21. Orientation3DErrorVector 51
         22. VanishingPoint 52
         23. VanishingPointError 52
         24. InformationSensorPose 52
         25. InformationSensorSurrounding 53
         26. InterfaceHeader 53
         27. SensorId 54
         28. SensorIdList 54
         29. InformationInterface 54
         30. InformationInterfaceExtension 55
         31. ProbabilityPercentage 55
         32. SensorCalibratableComponent 55
         33. SensorCalibrationStatus 56
         34. CaliComponentInfo 56
         35. CaliComponentInfoVector 57
         36. Calibration 57
         37. CalibrationProcessState 58
         38. SensorOriginTranslationCorrectionLimit 58
         39. SensorPoseAngleCorrectionLimit 59
         40. SensorCluster 59
         41. Point2D 59
         42. Point2DError 60
         43. InformationAmbiguityDomain 60
      2. Potentially Moving Objects Interface Definition 60
         1. MeasurementStatus 61
         2. ReferencePoint 61
         3. MovementStatus 62
         4. RoadLevel 62
         5. DimensionBox 63
         6. DimensionBoxError 63
         7. IncludedGeometricStructures 64
         8. PotentiallyMovingObjectClassificationType 64
         9. LightStatus 65
         10. LightType 65
         11. PersonPoseType 66
         12. ObjectLaneAssociation 66
         13. AngleBetweenObjectEdgeAndLaneRightEdgeLeft-

Lane 67

* + - 1. AngleBetweenObjectEdgeAndLaneRightEdgeLeft- LaneError 67
      2. PercentageSideLane 68
      3. PotentiallyMovingObjectsDynamics 68
      4. ObjectStatus 68
      5. PotentiallyMovingObjectsBoundingBox 69
      6. PotentiallyMovingObjectsInformation 69
      7. ValidPotentiallyMovingObjectClassification 70
      8. ValidPotentiallyMovingObjectClassificationVector 70
      9. PotentiallyMovingObjectsLight 70
      10. PotentiallyMovingObjectsLightVector 71
      11. PotentiallyMovingObjectsLights 71
      12. ValidPersonPose 71
      13. PersonPoseVector 72
      14. PotentiallyMovingObjectsPerson 72
      15. PotentiallyMovingObjectsLaneRelatedInformation 72
      16. PotentiallyMovingObjectsMotionRelatedInformation . 73
      17. ValidPotentiallyMovingObject 73
      18. ValidPotentiallyMovingObjectVector 74
      19. PotentiallyMovingObjects 74
      20. ValidObservation 74
      21. ValidObservationVector 75
      22. ObservationStatus 75
      23. PotentiallyMovingObjectsPosition 75
      24. PotentiallyMovingObjectsCameraSensorTechnolo- gySpecific 76
      25. PotentiallyMovingObjectsRadarSensorTechnology Specific 76
      26. PotentiallyMovingObjectsLidarSensorTechnolo- gySpecific 77
      27. PotentiallyMovingObjectInterface 77
    1. Road Objects Interface Definition 77
       1. RoadType 77
       2. RoadSurfaceClassificationType 78
       3. RoadConditionClassificationType 78
       4. RoadMarkingType 79
       5. ArrowDirection 80
       6. SignClassificationType 80
       7. SignValueUnit 81
       8. ConnectionType 81
       9. PolynomialCoefficient 82
       10. ColourTone 82
       11. RoadObjectInterface 82
       12. PolynomialRangeX 83

|  |  |  |
| --- | --- | --- |
| 9.1.3.13 | SupportedDataRangeX . . . . . . . . . . . . . . . . | 83 |
| 9.1.3.14 | PolylineInterpolationMethod . . . . . . . . . . . . . . | 83 |
| 9.1.3.15 | VertexPointConfidence . . . . . . . . . . . . . . . . . | 84 |
| 9.1.3.16 | RoadBoundaryType . . . . . . . . . . . . . . . . . . | 84 |
| 9.1.3.17 | RoadObjectLaneAssociation . . . . . . . . . . . . . | 85 |
| 9.1.3.18 | RoadBoundaries . . . . . . . . . . . . . . . . . . . . | 85 |
| 9.1.3.19 | RoadSurfaceClassification . . . . . . . . . . . . . . . | 86 |
| 9.1.3.20 | SignState . . . . . . . . . . . . . . . . . . . . . . . . | 86 |
| 9.1.3.21 | RoadSurfaceClassificationsVector . . . . . . . . . . | 87 |
| 9.1.3.22 | RoadSurfaceConditionClassification . . . . . . . . . | 87 |
| 9.1.3.23 | RoadSurfaceConditionClassificationsVector . . . . . | 87 |
| 9.1.3.24 | RoadSurface . . . . . . . . . . . . . . . . . . . . . . | 88 |
| 9.1.3.25 | ColourValueVector . . . . . . . . . . . . . . . . . . . | 88 |
| 9.1.3.26 | RoadMarkingClassification . . . . . . . . . . . . . . | 88 |
| 9.1.3.27 | RoadMarkingClassificationVector . . . . . . . . . . . | 89 |
| 9.1.3.28 | RoadMarkingsInformation . . . . . . . . . . . . . . . | 89 |
| 9.1.3.29 | ValidConnection . . . . . . . . . . . . . . . . . . . . | 90 |
| 9.1.3.30 | ValidConnectionVector . . . . . . . . . . . . . . . . . | 90 |
| 9.1.3.31 | ValidPolynomial . . . . . . . . . . . . . . . . . . . . . | 90 |
| 9.1.3.32 | SupportedDataRangeVector . . . . . . . . . . . . . . | 91 |
| 9.1.3.33 | ValidPolynomialVector . . . . . . . . . . . . . . . . . | 91 |
| 9.1.3.34 | Polynomials . . . . . . . . . . . . . . . . . . . . . . . | 92 |
| 9.1.3.35 | ValidVertice . . . . . . . . . . . . . . . . . . . . . . . | 92 |
| 9.1.3.36 | ValidVerticeVector . . . . . . . . . . . . . . . . . . . | 93 |
| 9.1.3.37 | ValidPolyline . . . . . . . . . . . . . . . . . . . . . . | 93 |
| 9.1.3.38 | ValidPolylineVector . . . . . . . . . . . . . . . . . . . | 93 |
| 9.1.3.39 | Polylines . . . . . . . . . . . . . . . . . . . . . . . . . | 94 |
| 9.1.3.40 | RoadMarking . . . . . . . . . . . . . . . . . . . . . . | 94 |
| 9.1.3.41 | RoadMarkingVector . . . . . . . . . . . . . . . . . . | 94 |
| 9.1.3.42 | RoadMarkings . . . . . . . . . . . . . . . . . . . . . | 95 |
| 9.1.3.43 | ValidRoadBoundaryClassification . . . . . . . . . . . | 95 |
| 9.1.3.44 | ValidRoadBoundaryClassificationVector . . . . . . . | 96 |
| 9.1.3.45 | RoadBoundariesInformation . . . . . . . . . . . . . . | 96 |
| 9.1.3.46 | ValidRoadBoundary . . . . . . . . . . . . . . . . . . | 96 |
| 9.1.3.47 | RoadBoundaryVector . . . . . . . . . . . . . . . . . | 97 |
| 9.1.3.48 | SupportedAxis . . . . . . . . . . . . . . . . . . . . . | 97 |
| 9.1.3.49 | SignClassification . . . . . . . . . . . . . . . . . . . . | 97 |
| 9.1.3.50 | ValidSignClassificationVector . . . . . . . . . . . . . | 98 |
| 9.1.3.51 | SupportedDataRange . . . . . . . . . . . . . . . . . | 98 |
| 9.1.4 Static | Objects Interface Definition . . . . . . . . . . . . . . . | 98 |
| 9.1.4.1 | GeneralLandmarkClassificationType . . . . . . . . . | 99 |
| 9.1.4.2 | SignGeometry . . . . . . . . . . . . . . . . . . . . . | 99 |
| 9.1.4.3 | TrafficSignsInformation . . . . . . . . . . . . . . . . . | 100 |
| 9.1.4.4 | TrafficLightsInformation . . . . . . . . . . . . . . . . | 100 |
| 9.1.4.5 | LaneRelevanceClassificationType . . . . . . . . . . . | 101 |
| 9.1.4.6 | SupplementarySignClassificationType . . . . . . . . | 101 |

* + - 1. RelativePosition 102
      2. StructureLightClassificationType 102
      3. ColourClassificationType 102
      4. GeneralLandmarksInformation 103
      5. LightShapeClassificationType 103
      6. LightModeClassificationType 104
      7. ValidGeneralLandmarkClassificationVector 105
      8. GeneralLandmarkClassification 105
      9. GeneralLandmarksPosition 105
      10. BoundingBox 106
      11. GeneralLandmark 106
      12. GeneralLandmarkVector 106
      13. GeneralLandmarks 107
      14. LaneRelevanceClassification 107
      15. MainSignClassification 107
      16. TrafficSignsPosition 108
      17. SubObjectStatus 108
      18. TrafficSignsSupplementarySignsInformation 109
      19. SupplementarySignClassification 109
      20. TrafficSignsSupplementarySignsPosition 109
      21. TrafficSignsSupplementarySign 110
      22. TrafficSignsSupplementarySignVector 110
      23. TrafficSignsSupplementarySigns 110
      24. TrafficSign 111
      25. ValidTrafficSignVector 111
      26. TrafficSigns 112
      27. StructureLightClassification 112
      28. TrafficLightsPosition 112
      29. ColourClassification 113
      30. ColourClassificationVector 113
      31. TrafficLightsSpotsColour 114
      32. LightModeClassification 114
      33. LightModeClassificationVector 114
      34. LightShapeClassification 115
      35. TrafficLightsSpotsInformation 115
      36. TrafficLightsSpotsPosition 115
      37. TrafficLightSpot 116
      38. TrafficLightSpotVector 116
      39. TrafficLightSpots 116
      40. TrafficLight 117
      41. ValidTrafficLightVector 117
      42. TrafficLights 117
      43. StaticObjectInterface 118
      44. ValidSupplementarySignClassificationVector 118
      45. ValidMainSignClassificationVector 118
      46. ValidLaneRelevanceClassificationVector 119

|  |  |  |  |
| --- | --- | --- | --- |
| 9.1.4.53 | | LightShapeClassificationVector . . . . . . . . . . . . | 119 |
| 9.1.4.54 | | ValidStructureLightClassificationsVector . . . . . . . | 119 |
| 9.1.5 Feature | | Level Interface Definition . . . . . . . . . . . . . . . | 120 |
| 9.1.5.1 | | ShapeType . . . . . . . . . . . . . . . . . . . . . . . | 120 |
| 9.1.5.2 | | ShapeClassificationType . . . . . . . . . . . . . . . . | 120 |
| 9.1.5.3 | | UltrasonicFeatureClassificationType . . . . . . . . . | 121 |
| 9.1.5.4 | | TrilaterationStatus . . . . . . . . . . . . . . . . . . . | 122 |
| 9.1.5.5 | | MeasurementStatusFeature . . . . . . . . . . . . . . | 122 |
| 9.1.5.6 | | ShapeClassification . . . . . . . . . . . . . . . . . . | 122 |
| 9.1.5.7 | | ValidShapeClassificationsVector . . . . . . . . . . . | 123 |
| 9.1.5.8 | | CameraFeaturesShapeInformation . . . . . . . . . . | 123 |
| 9.1.5.9 | | ShapePoint . . . . . . . . . . . . . . . . . . . . . . . | 123 |
| 9.1.5.10 | | ValidShapePointVector . . . . . . . . . . . . . . . . . | 124 |
| 9.1.5.11 | | ShapePoints . . . . . . . . . . . . . . . . . . . . . . | 124 |
| 9.1.5.12 | | ShapeReferencePoint . . . . . . . . . . . . . . . . . | 124 |
| 9.1.5.13 | | ValidShapeReferencePointVector . . . . . . . . . . . | 125 |
| 9.1.5.14 | | ShapeReferencePoints . . . . . . . . . . . . . . . . . | 125 |
| 9.1.5.15 | | FeatureStatus . . . . . . . . . . . . . . . . . . . . . . | 126 |
| 9.1.5.16 | | CameraFeature . . . . . . . . . . . . . . . . . . . . . | 126 |
| 9.1.5.17 | | ValidCameraFeatureVector . . . . . . . . . . . . . . | 126 |
| 9.1.5.18 | | CameraFeatureInterface . . . . . . . . . . . . . . . . | 127 |
| 9.1.5.19 | | UltrasonicSegmentInformation . . . . . . . . . . . . | 127 |
| 9.1.5.20 | | UltrasonicFeatureClassification . . . . . . . . . . . . | 128 |
| 9.1.5.21 | | SegmentPoint . . . . . . . . . . . . . . . . . . . . . . | 128 |
| 9.1.5.22 | | ValidSegmentPointVector . . . . . . . . . . . . . . . | 128 |
| 9.1.5.23 | | UltrasonicSegmentPoints . . . . . . . . . . . . . . . | 129 |
| 9.1.5.24 | | UltrasonicFeature . . . . . . . . . . . . . . . . . . . . | 129 |
| 9.1.5.25 | | ValidUltrasonicFeatureVector . . . . . . . . . . . . . | 129 |
| 9.1.5.26 | | UltrasonicFeatureInterface . . . . . . . . . . . . . . . | 130 |
| 9.1.5.27 | | ValidUltrasonicFeatureClassificationVector . . . . . . | 130 |
| 9.1.6 | Detection Level Interface Definition . . . . . . . . . . . . . . | | 130 |
|  | 9.1.6.1 Position3DSpheric . . . . . . . . . . . . . . . . . . . | | 131 |
|  | 9.1.6.2 Position3DSphericError . . . . . . . . . . . . . . . . | | 131 |
|  | 9.1.6.3 DetectionClassificationType . . . . . . . . . . . . . . | | 131 |
|  | 9.1.6.4 DetectionsPosition . . . . . . . . . . . . . . . . . . . | | 132 |
|  | 9.1.6.5 UltrasonicDetectionsPosition . . . . . . . . . . . . . | | 132 |
|  | 9.1.6.6 DetectionStatus . . . . . . . . . . . . . . . . . . . . . | | 133 |
|  | 9.1.6.7 DetectionClassification . . . . . . . . . . . . . . . . . | | 133 |
|  | 9.1.6.8 ValidDetectionClassificationVector . . . . . . . . . . | | 133 |
|  | 9.1.6.9 RadarDetectionsInformation . . . . . . . . . . . . . . | | 134 |
|  | 9.1.6.10 RadarDetection . . . . . . . . . . . . . . . . . . . . . | | 134 |
|  | 9.1.6.11 ValidRadarDetectionVector . . . . . . . . . . . . . . | | 135 |
|  | 9.1.6.12 DetectionsDynamics . . . . . . . . . . . . . . . . . . | | 135 |
|  | 9.1.6.13 RadarDetectionsInterface . . . . . . . . . . . . . . . | | 135 |
|  | 9.1.6.14 LidarDetectionsInformation . . . . . . . . . . . . . . | | 136 |
|  | 9.1.6.15 LidarDetection . . . . . . . . . . . . . . . . . . . . . | | 136 |

* + - 1. ValidLidarDetectionVector 136
      2. LidarDetectionsPosition 137
      3. LidarDetectionsInterface 137
      4. DetectionShapeClassification 137
      5. ValidDetectionShapeClassificationVector 138
      6. CameraShapesShapeInformation 138
      7. ShapePointDetectionLevel 138
      8. ValidShapePointDetectionLevelVector 139
      9. CameraShapesShapePoints 139
      10. CameraShape 139
      11. ValidCameraShapeVector 140
      12. ShapeReferencePointDetectionLevel 140
      13. ValidShapeReferencePointDetectionLevelVector 140
      14. CameraShapesShapeReferencePoints 141
      15. CameraDetectionsInterface 141
      16. UltrasonicDetectionsInformation 141
      17. UltrasonicDetection 142
      18. ValidUltrasonicDetectionVector 142
      19. UltrasonicDetectionsInterface 142
    1. Supportive Sensor Interfaces Definition 143
       1. SegmentAzimuth 143
       2. SegmentElevation 143
       3. AnglePoint3D 143
       4. BeamDivergence 144
       5. SegmentsStatus 144
       6. BlockageStatus 145
       7. FieldOfViewReduction 145
       8. FieldOfViewReductionReasonType 146
       9. ValidFieldOfViewReductionReasonsVector 146
       10. FieldOfViewReductionReasons 147
       11. RecognizedObjectType 147
       12. DetectionRange 148
       13. RecognisableObjectTypes 148
       14. ValidRecognisableObjectTypesVector 148
       15. RealWorldObjectRecognitionCapabilities 149
       16. ReferenceTargetType 149
       17. ReferenceTargetTypes 149
       18. ValidReferenceTargetTypesVector 150
       19. ReferenceTargetRecognitionCapabilities 150
       20. RelativeRadialVelocityRange 151
       21. PeformanceSegment 151
       22. SensorOperationMode 151
       23. SensorDefectDetected 152
       24. SensorDefectReason 152
       25. ValidSensorOperationModeVector 153
       26. StatusSupplyVoltage 153
       27. SensorTemperatureStatus 154
       28. ValidPeformanceSegmentVector 154
       29. SensorInputSignalType 155
       30. SensorInputSignalStatus 155
       31. SensorExternalDisturbed 156
       32. SensorTransmitPowerReduced 156
       33. StatusSensorHeating 157
       34. StatusSensorCleaning 157
       35. SensorTimeSync 157
       36. SensorPerformanceSegments 158
       37. SensorPerformanceInterface 158
       38. InputSignalStatus 159
       39. ValidInputSignalStatusVector 159
       40. SensorHealthStatus 159
       41. SensorHealthInformationInterface 160
  1. Service Interfaces 160
     1. Sensor Interfaces Port 161
     2. Object level Interfaces 161
     3. Feature level Interfaces 162
     4. Detection level Interfaces 163
     5. Supportive Interfaces 165

1. Capability Configuration 167
   1. Object Level Service 167
      1. PMObjectsService Capability Vector 167
      2. RObjectsService Capability Vector 170
      3. SObjectsService Capability Vector 175
   2. Feature Level Service 179
      1. CameraFeatureService Capability Vector 179
      2. USSFeatureService Capability Vector 182
   3. Detection Level Service 185
      1. RadarDetectionService Capability Vector 185
      2. LidarDetectionService Capability Vector 187
      3. CameraDetectionService Capability Vector 190
      4. USSDetectionService Capability Vector 192
   4. Supportive Service 195
      1. SensorPerformanceService Capability Vector 195
      2. SensorHealthService Capability Vector 197

# Constraints and assumptions

Dedicated interfaces for particular sensors like radar, lidar, USS and camera are pro- vided.

## Limitations

### ISO Optionals

All the ISO optionals are supported in the service defintiion, but the presence of an optional signal is decided during the design time. The optional signal presence is indicated by a capability vector. There is no change of optional presence during the run time.

### Sensor Control Interfaces

Sensor control interfaces are not supported, e.g. actions like reset, initialization, and calibration.

### Sensor Capabilities

The configuration of sensor capabilities is not supported, e.g. sensor opening angle and detection range.

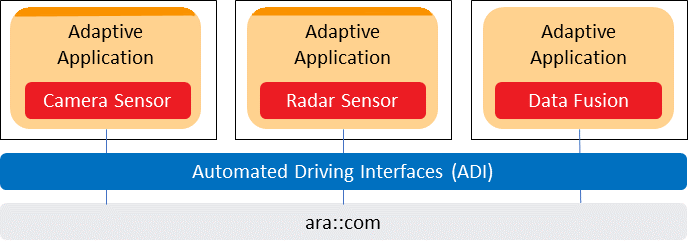
## Applicability to car domains

AUTOSAR Sensor Interfaces are used as service interfaces between dedicated sen- sors for enviroment precesion and the AUTOSAR Adaptive applications.

# Dependencies to other modules

The Automated Driving Interfaces serve as common interfaces for the AUTOSAR Adaptive applications. The sensor information is provided by a non-platform service.The sensor interfaces are exposed to client applications via the ara::com middleware. Communication Management,[5] uses Identity and Access Management

[6] to validate the authorization of requests made to the specific service interfaces, i.e. the Communication Management shall check if the invoker is allowed to access the re- quested service interface of a specific sensor. The following Figure shows a radar and camera sensor instance which provide sensor information to a Data Fusion instance consuming sensor data.



**Figure 5.1: Sensor Interfaces dependencies to other AUTOSAR Adaptive Applica- tions.**

For each sensor type like radar, lidar, USS or camera, the ISO standard will contain lots of optional sensor data or signals in addition to the required signals, e.g. the classification of an object might be optional. During development of a driving function or data fusion system, a specific set of optional signals is selected by the function or data fusion developer according to the requirements of the driving function. To reduce development costs, especially in terms of functional safety, the set of optionals and thus the resulting logical sensor interface are required to be fixed during design-time, i.e. the specified sensor signals must not suddenly disappear from the interface of the sensor during runtime of the system so that a data fusion system can rely on the presence of the specified signals. Vice versa, a sensor interface must not add additional signals during runtime, e.g. not to risk an unintentional behavior change due to the increased bandwidth requirements.

# Functional Specification

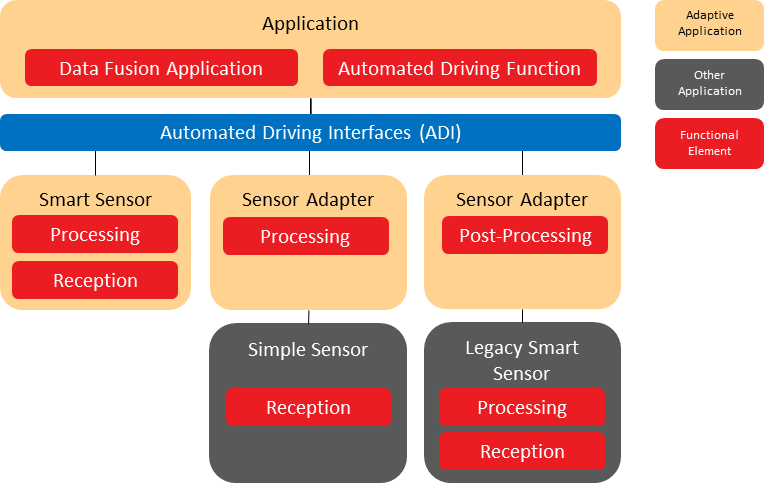
## Outline

### Goals and scope

The specification is to provide well-defined sensor interfaces which is compliant to the sensor interface specification [1]. The ISO document covers just semantic definitions of interfaces whereas this specification aims to cover all aspects of the interfaces to make them fully compliant to the AUTOSAR Adaptive platform.

The specification shall enable the compatibility of sensors and data fusion algorithms independently of the supplier. To ensure the compatibility between sensors the inter- faces are provided on implementation data type level.

The Automated Driving Interfaces serve as common interfaces for the AUTOSAR Adaptive applications. Consumers of the interfaces are AUTOSAR Adaptive applications containing functional elements as sensor data receivers, sensor data processors, data fusion applications, and automated driving functions. The sensor in- formation is provided by a non-platform service.The non-platform service consumer is running in the apdative platfom as an AUTOSAR adaptive application, where the sensor could be deployed on different AUTOSAR platforms (Classic and Adaptive) as shown in the following figure.



**Figure 7.1: Functional elements communicating via ADI**

Apart from the Automated Driving Interfaces (ADI), the other relevent elements are defined as follows :

* Sensor Data Reception: A functional element which receives or reads sensor data from a sensor. Such an element can provide raw sensor data. Raw sensor data means that it provides the actual data a sensor can measure. There is no processing of the sensor data. There are no data streams provided.
* Sensor Data Processing: This element which usually processes raw sensor data, provides pre-processed and post-processed sensor data. It provides data a sen- sor cannot directly measure but needs a processing stage for.
* Sensor Fusion Application: The data fusion algorithm is an application which col- lects sensor data and fuses it to provide a unified environmental model. Data fusion applications are typical consumers of sensor data. They provide their out- put to Automated Driving Functions.
* Automated Driving Function: Automated Driving Functions can also directly use the interfaces.

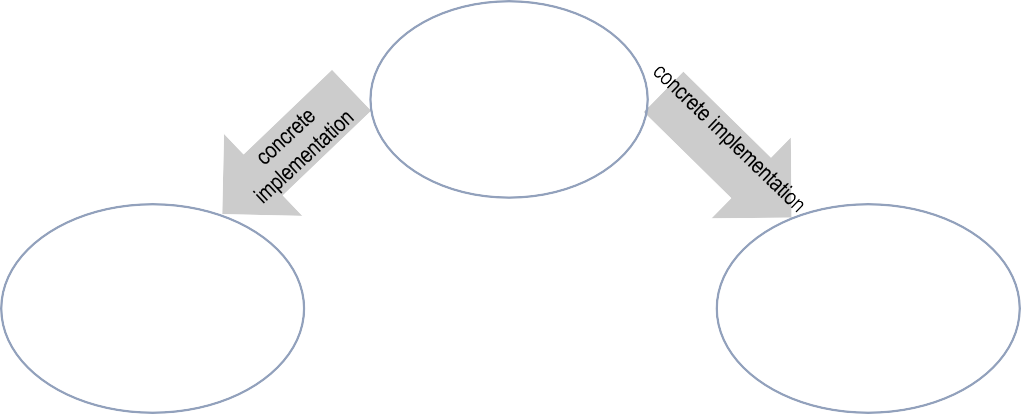
The Automated Driving interfaces can be used in following usages to enable the in- teroperability between different players and improve the efficiency for the development and validation of automated driving functions :

* Sensor Supplier Interface: The automated driving use case defines the require- ments to the fusion algorithm and the used sensors. The OEM wants to rely on a standardized interface which provides well defined sensor information which are defined by the ISO-23150 and on an AUTOSAR standardized interface. The scope shall be the data format between a sensor and a computing unit running an AUTOSAR Adaptive application.
* Standardized Sensor API: Sensor developers need well described and standard- ized interfaces to provide an implementation which can be used by automated driving applications and data fusion algorithms. So, the same information is pro- vided to all potential consumers of the data.
* Sensor Fusion Algorithm: To combine information from different sensors and to develop sensor fusion algorithms for automated driving a detailed description of all signals provided is needed.
* Sensor Fusion Integration: Sensor fusion algorithms shall be integrated on a high-performance computing platform within an Adaptive Application as the cen- tral unit of automated driving functions.
* Sensor Implementation Testing: Testing is performed at each stage of the de- velopment of the sensor system. To create a test specification for the system testable implementations are needed which are based on well-defined interfaces and descriptions.
* Sensor Simulation: By simulating the sensor fusion algorithms before actual tar- get integration, it is possible to ensure smooth integration during the design, de- velopment, test and safeguarding steps. This ensures compatibility of algorithm integration into AUTOSAR Adaptive ECUs.
* Sensor Data Record and Replay: To be able to test and debug sensor fusion algorithms offline it is necessary to replay sensor data recorded in real world sce- narios. This enables sensor fusion designers to reproduce failures and determine faults.

**Standardization Level of detail**

### Relation to other standards

The currently ongoing standardization of the international standard ISO 23150 "Road vehicles - Data communication between sensors and data fusion unit for automated driving functions - Logical interface" will be applicable to road ve- hicles with automated driving functions. It specifies the logical interface be- tween smart in-vehicle sensors that sense the environment (e.g. camera, li- dar, radar, ultrasonic, etc.) and the vehicle fusion unit. The interface is de- scribed in a modular, semantic representation allowing different types of sen- sor technologies and fusion concepts. The "Open Simulation Interface" (OSI) is a generic software interface for the environmental perception of automated driv- ing functions in virtual scenarios (https://github.com/OpenSimulationInterface/open- simulation-interface). OSI ensures modularity, integrability and interchangeabil- ity of the individual components. These can be environment simulation mod- els, sensor models, logical models for the detected environment or sensor fusion and models for automated driving functions. Additional informations can be found on https://opensimulationinterface.github.io/open-simulation-interface/.



**Standardization in ISO**

Semantic description of the Interfaces, current VDA proposal

**OSI**

**(Open Simulation Interface)** Simulation specific implementation of the

standardized interface in

github

**AUTOSAR FT-ADI**

Concrete implementation of the standardized interface towards application

**Figure 7.2: Scope of Standardization**

The Open Simulation Interface and the Autosar ADI will support the ISO 23150.

## AD Sensor Service Design

### ISO mapping to Sensor Services

### Sensor Types

To comply with the ISO 23150, the specific service is defined with different sensor types. Currently, Camera, Lidar, Radar, and Ultrasonic sercives are supported in the specification. The new service will be introduced along with the evolution of [1].

**[SWS\_ADI\_00001]**{DRAFT} **The Camera Sensor Data periodical Transmission** *[*A Camera senor with Automated Driving interfaces shall provide the camera sensor data periodically to the Adaptive Applications*♩(RS\_ADI\_00001)*

**[SWS\_ADI\_00002]**{DRAFT} **The Lidar Sensor Data periodical Transmission** *[*A Lidar sensor with Automated Driving interfaces shall provide the lidar sensor data pe- riodically to the Adaptive Applications*♩(RS\_ADI\_00002)*

**[SWS\_ADI\_00003]**{DRAFT} **The Radar Sensor Data periodical Transmission** *[*A radar sensor with Automated Driving interfaces shall be able to provide the radar sen- sor data periodically to the Adaptive Applications*♩(RS\_ADI\_00003)*

**[SWS\_ADI\_00004]**{DRAFT} **The Ultrasonic Sensor Data periodical Transmission** *[*A ultrasonic sensor with Automated Driving interfaces shall be able to provide the ul- trasonic sensor data periodically to the Adaptive Applications*♩(RS\_ADI\_00004)*

**[SWS\_ADI\_00005]**{DRAFT} **Receving periodical sensor data** *[*The Adaptive Ap- plications as the sevice consumer shall process the sensor data according to the timestamp in the package, not the package arrival time. The most recent data is always the highest priority for processing.*♩(RS\_ADI\_00018)*

### Sensor Data level

In ISO 23150, there are three ISO logic interface levels to support different levels of fusion: object level, feature level and detection level:

* + - * + Object level interface: Potentially moving object interface, Road object interface and Static object interface;
        + Feature level interface: Camera feature interface and USS feature interface;
        + Detection level interface: Radar detection interface,Lidar detection interface, Camera detection interface and USS detection interface.

A sensor could provide several ISO logical interfaces, and shall provide at least one ISO interface either on object, feature, or detection level.

**[SWS\_ADI\_00006]**{DRAFT} **Sensor specific services for different level ISO inter- faces** *[*To allow the flexibility of service providing by a sensor, each ISO interface shall

be mapped to a propriate ara::com means like events. The service interface shall support the defintions of ISO signals.The services for new types of sensors shall be added along with the evolution of [1]. The follwing AP services are defined in this specification:

* + - * + Camera,Lidar,Radar,Ultrasonic: Potentially moving object service, road object service, static object service
        + Camera: Camera features service, camera detections service;
        + Lidar: Lidar detections service;
        + Radar: Radar detections service;
        + Ultrasonic: Ultrasonic features service, Ultrasonic detection service.

*♩(RS\_ADI\_00012, RS\_ADI\_00013, RS\_ADI\_00014, RS\_ADI\_00019, RS\_ADI\_00005)*

### Supportive Sensor interfaces

In ISO, there are two supportive senor interfaces for the Adaptive Applications process the reliability of the used measurement method or data, which are also used to the implementation of safety concept. The two supportive sensor interfaces are sensor technology indepedent, i.e. common to camera, lidar, radar and ultrasonic sensor, but with sensor technology specific data.

**[SWS\_ADI\_00007]**{DRAFT} **Sensor independent supportive services** *[*The sup- portive interfaces shall be mapped to two sensor indpendent AP services and support the defintions of ISO signals.The follwing services are defined in this specification:

* + - * + Sensor Performance service;
        + Sensor Health information service.

*♩(RS\_ADI\_00012, RS\_ADI\_00013, RS\_ADI\_00014, RS\_ADI\_00019)*

### Service Optional Elements

An individual Sensor Service design faces the challenge that each ISO interface con- tains a lot of optional elements. The optional elements are known and fixed at design time by service providers and service users. The present of optional elements shall not be changed during run time.

**[SWS\_ADI\_00010]**{DRAFT} **Capability Vector** *[*A service capability vector indicates which optional element is provided by the Service provider. The optional elements indicated by the capability vector shall be always sent during the runtime of the service. The service consumer should check the vector at the initialization time and subcribe the service if it includes all the optional elements that the consumer requires.The consumer

may ignore the optional elements, if these elements are not required.*♩(RS\_ADI\_00006, RS\_ADI\_00015, RS\_ADI\_00016, RS\_ADI\_00017)*

**[SWS\_ADI\_00011]**{DRAFT} **Service Profiles** *[*In different use cases, there are dif- ferent requirements on the sensor data i.e.for different data Types, behavior, content, performance, resolution, etc. Different Service Profiles for a Sensor Type enables the flexiblity of the corresponding sensor service for different use cases. In this re- lease, only one profile is supported and more profiles should be considered in future releases.*♩(RS\_ADI\_00007)*

**[SWS\_ADI\_00012]**{DRAFT} **Service Versioning** *[*Service Versioning covers different configurations, i.e. optionals, additional data representation, etc. The same services with different service versioning, but with the same profile, shall be backward compati- ble.*♩(RS\_ADI\_00006, RS\_ADI\_00015, RS\_ADI\_00016, RS\_ADI\_00017)*

# API specification

There are no APIs defined in this release.