EC[6]

Methodological Guideline for Operational Design Domain

Document reference number for ANR

contact@confiance-ai.fr | www.confiance.ai

Document reference: XXX

Contributors

	Name	Organisation	Role
Scientific responsible	Morayo ADEDJOUMA	CEA LIST	EC6.20 leader
Co-authors	Bernard BOTELLA	CEA LIST	
	Kevin MANTISSA	IRT SystemX	
Reviewers	Chauk-Mean PROUM	Naval Group	

Document Control

Revision	Date	Commentary	Author
v0.1	31/08/2024	Release for review	Lead and all co-authors
v1	04/09/2024	Final version after review	Chauk-Mean PROUM

Contents

A. Introduction

A.1. General introduction to trustworthy AI challenges

Trustworthiness in AI within critical systems (systems that can directly or indirectly affect human life and moral entities) is essential for its widespread adoption (by the industry, the decision makers, the general public, etc.) and poses the following significant challenges.

- First, how to design AI models, so that, by construction, they satisfy trustworthy properties (accuracy, robustness. . .).
- Secondly, how to characterize these AI models, for example to understand and explain their behavior and their adequacy to the operational domain.
- Then, how to implement and embed those AI models on hardware, by making them fit for the target without losing their trustworthy properties.
- Another question is, what methods of data engineering to apply in order to, among other topics, manage important volumes of data and adapt to the evolution of the operational domain.
- At system level, what verification and certification processes to consider specifically for AI-based systems.
- Finally, a federation of all these matters is necessary to build an end-to-end methodological approach, supported by a consistent engineering environment compatible with industrial practices.

These are the challenges, among others, that the Confiance.ai program addresses.

A.2. Context and motivation

Within the Confiance.ai program, an objective is to revisit the existing IVVQ processes with regard to the challenges that pose the AI integration into complex systems. These challenges concerned not only the Verification, Validation and Qualification processes, but also the definition of the operational domain where the systems is intended to operate. The present document addresses particularly the challenge on the definition of the operational domain. In practice, the scenario-space, i.e., the number of possible scenarios that have to be managed by an automated systems tends to be infinite. For example, in the case of data-driven AI, the intended purpose of the system is in general learned from data. Nonetheless, it is impossible to ensure that these data capture the infinite number of scenarios in which automated systems must operate, which makes their safety evaluation challenging. Then, one needs a mean to define the scenario-space in which the automated system must operate safely without having to enumerate the different scenarios individually. The scenario-space is specified through the Operational Design Domain (ODD).

A.3. Added value of the proposed method

The Operational Design Domain (ODD) gathers the operating conditions under which a given system feature is specifically designed to function SAE Mobilus (2018). It is of crucial importance to define the ODD of AI systems features in order for them to operate correctly within the specified domain, recognizing that no guarantees can be provided outside their ODD. *At design time*, one must ensure that the system can function in the operational conditions included in its ODD. It is defined during the design process and verified and tested during the V&V phase. *At operational time*, monitoring the ODD helps ensure that the system remains in the operational environment in which it is designed to function. Indeed, the definition of ODD is identified as a key element that contributes to all AI system trustworthiness attributes.

A.4. Rationale for this methodological guideline

The concept of ODD originates from autonomous driving domain. In this deliverable, we extend the notion of the ODD to AI-based systems for all application areas. The state of the art regarding the concept is largely documented, regarding its description, but no detailed elements permit to define the associated engineering process. Besides, there are different usages of the ODD by the system engineer, the data engineer, the V&V engineer, the safety engineer, the end-user, the manufacturer, the certification body. This results in different viewpoints per engineering domains. The current document presents the methodological guidelines to define the Operational Design Domain of an AI-based system through the system development lifecycle. We first present a generic approach for defining the ODD and its characteristics. Then we present how this approach must be reconducted at different stages in the system development cycle, so that the ODD is characterized at different granularities through a refinement process to fit each stakeholder's specific needs. Additionally, we present the quality attributes to look into for evaluating the ODD quality and validity, including the criteria with regards to AI trusworthiness attributes.

A.5. Limitations and perspectives

It is worth mentioning all methodological guidelines provided in the deliverable have not the same level of maturity. While the generic approach has been proven in industrial contexts on data-driven automotive and naval AI applications, the refinement process through the system lifecycle has only been evaluated at a theoretical level. To conveniently validate the iterative method and its generalizability to any AI based system, more rigorous experimentations on different use cases must be conducted to judge about its soundness and applicability. Findings of these evaluations will enable to strengthen our guidelines. Future work include to strengthen our return of experience (retex) on the proposed approaches on others industrial use cases, including how the ODD definition processes interplay in practice with other engineering processes developed in the framework of the program.

A.6. Document organization and How to use the guideline

The deliverable is organized as follows. Chapter A introduces the context and the motivation to develop a method for ODD definition. The core elements of the deliverable are presented from Chapter B to Chapter E:

- Chapter B presents the method to define an ODD. The ODD definition method is a consolidation of both the taxonomy-based and the analytical approaches developed in Adedjouma et al. (2021) and Adedjouma et al. (2022), respectively. Note that, the state-of-art makes case of a data-driven approach to define an ODD Zhang et al. (2021). Nonetheless, our expertise does not allow us to argue on such data-based approach (see F). There is also no

publicly available sufficient documentation that explains the engineering process attached to such an approach to allow us a critical analysis or an integration into our reflections.
- Chapter C highlights how the ODD process can interact with the proposed overall End-to-end approach proposed by the confiance.ai program.
- Chapter D presents the principles and activities to refine the ODD at different engineering phases. The chapter is organized in sections per engineering phase.
- Chapter E presents the desirable quality properties for an ODD and the relations of the ODD definition with key trustworthiness properties.
- Chapter F reports on the evaluation of the ODD methodological guidelines and related artefacts through industrial use cases.
- Finally, Chapter G resumes the deliverable contribution and discusses some future perspectives.

A.7. Acronyms

Acronym	Signification
AI	Artificial Intelligence
BPMN	Business Process Modelling Notation
DL	Deep Learning
HW	Hardware
IVVQ	Integration Verification Validation Qualification
KPI	Key Performance Indicator
ML	Machine Learning
NN	Neural Network
ODD	Operational Design Domain
SW	Software
UC	Use Case
UML	Unified Modeling Language
V&V	Verification and Validation

B. ODD definition process for an automated feature

In past years, we developed two different approaches to define an Operational Design Domain (ODD) : a taxonomy-based approach and an analytical approach.
Through this guideline, the aim is to unify these approaches and to consolidate the resulting definition process through an industrial experimentation.
Naval Group has experimented the definition of an ODD through two proprietary Use Cases and has formalized the practiced process which combines the previous different approaches. Subsequently, the process followed by Naval Group has been challenged and enriched by the whole ODD team.
The description of the ODD definition process is structured in 3 sections :

- the process presentation
- the process meta-model
- an example of ODD definition obtained from the process application based on an excerpt from a Naval Group use case

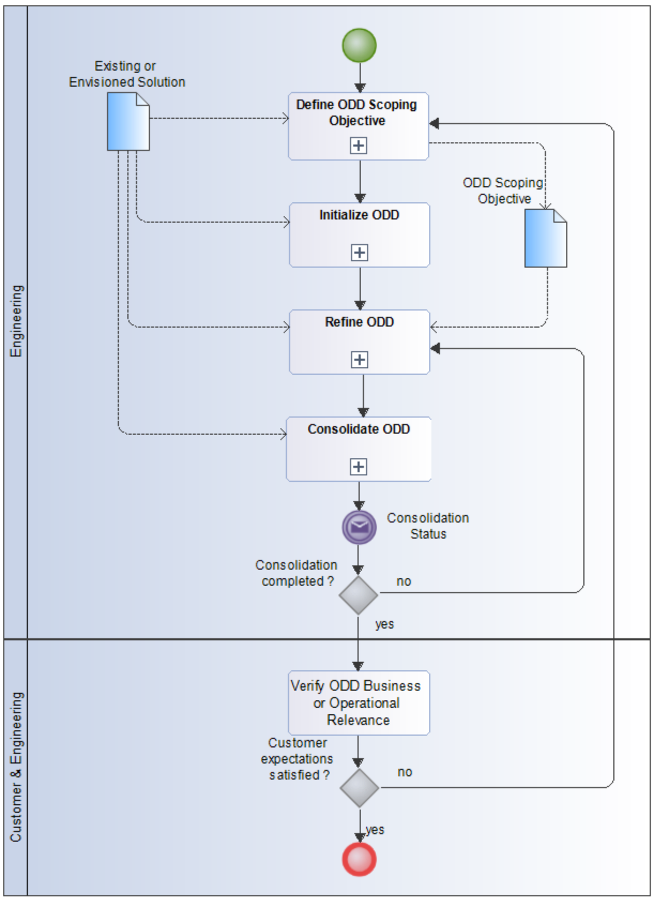

Figure B.1: Overall ODD definition process

B.1. Process description

Figure B.1 presents the process for defining an ODD via BPMN (Business Process Modelling Notation) process diagrams. The process is composed of 5 steps:

- ODD scoping objective definition
- ODD initialization
- ODD refinement
- ODD consolidation
- ODD business or operational relevance verification

The first 4 steps are conducted by the engineering team with an existing or an envisioned system solution as input. The last step is conducted by the engineering team in collaboration with the customer or his representative. These steps are detailed in the following subsections.

B.1.1 ODD scoping objective definition

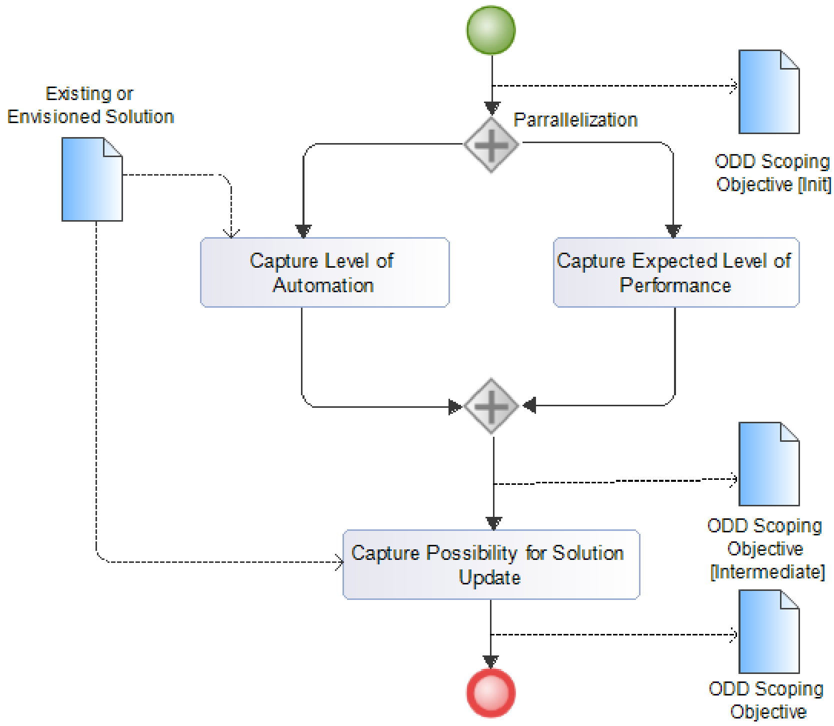

Figure B.2: ODD scoping objective definition

Description

An ODD may be defined for different "kinds" of system solutions and corresponding purposes. In a perspective of an already *existing solution*, the ODD will serve to define the scope for its

qualification. For an *envisioned solution*, which is typically based on a prototype, the ODD may serve to identify areas where adjustments can be made to the solution to fulfill the expected level of automation or expected level of performance, according to the available scale per application domain.
As shown on Figure B.2, the ODD scoping objective aims at capturing these two different perspectives through 3 criteria :

- level of automation : decision aid or autonomous decision making (and possible action).
- expected level of performance, which is interdependent on the level of automation. A given level of automation may require some levels of performance. Some levels of performance may enable a level of automation.
- possible solution update: in the case of an envisioned solution, the area(s) where functional improvement is possible or conversely excluded.

Involved actors

- The engineering team

Inputs

- The specification of the existing or envisioned solution. The specification may include the operational context description, the system requirements, the architecture and design information, known constraints on recommended or required technologies/components from existing or similar systems.

Activities

- Capture level of automation
- Capture expected level of performance
- Capture possible solution update

Outputs

- The ODD scoping objective

B.1.2 ODD initialization

Description

The ODD initialization (see Figure B.3) borrows elements from the taxonomy-based approach where a hierarchical structure of attributes is defined Adedjouma et al. (2021).
This step captures all attributes from the customer expectations, (i.e. customer needs and requirements), through three aspects borrowed from the BSI/PAS 1883 British Standard Institution (2020) and that have been consolidated in the ISO 34503 standard for the automotive ODD ISO 34503:2023 (2023):

- Scenery : the scene or static elements surrounding the system of interest, such as road structure or drivable area for an autonomous car
- Dynamic Elements : the moving elements within the scene, such as other drones or flying animals for a drone system
- Environmental Conditions, such as illumination and vibration for a production system

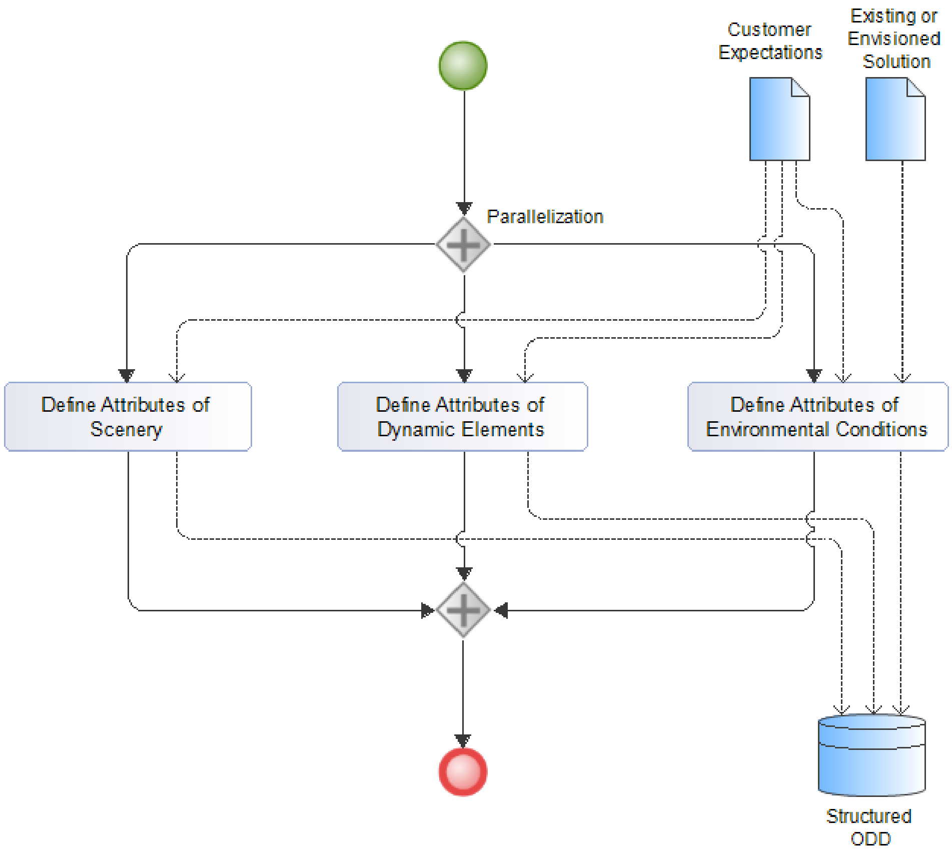

Figure B.3: ODD initialization

A decomposition of these attributes and their characterisation via values are defined as needed. This step captures also environmental condition attributes that are taken into account or imposed by the existing or envisioned solution.

Involved actors

- The engineering team

Inputs

- Customer expectations
- Description of the existing or envisioned solution

Activities

- Define attributes of Scenery
- Define attributes of Dynamic Elements

- Define attributes of Environmental Conditions

Outputs

- ODD attributes with possible sub-attributes and/or attribute values stored in the "Structured ODD" through a database, or any other form of knowledge base support.

B.1.3 ODD refinement

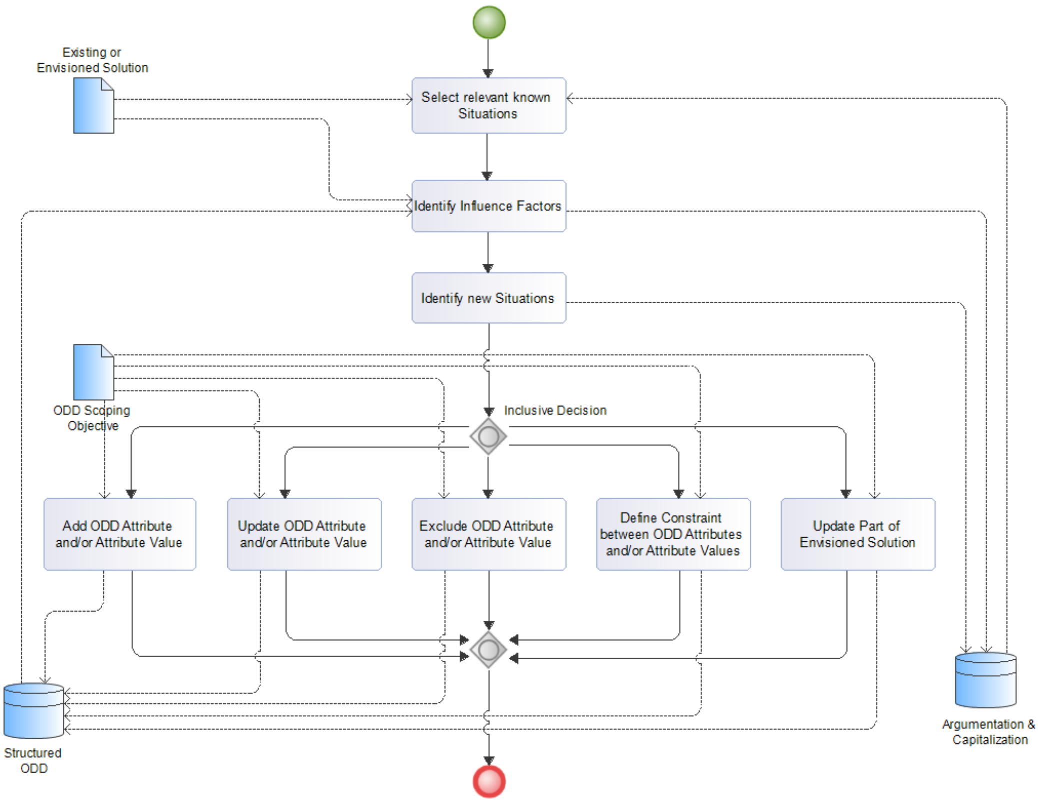

Figure B.4: ODD refinement

Description

This ODD refinement (see Figure B.4) borrows elements from the analytical approach, e.g. *influence factors* concept Adedjouma et al. (2022) for refining the ODD previously initialized. First, known situations (such as the tunnel entry for an autonomous car) are retrieved from an "Argumentation and Capitalization" database[1] and selected if relevant for the *existing or envisioned solution* with the corresponding *influence factors*, such as, e.g, a variation of illumination. Then each ODD attribute from the "Structured ODD" database is analysed to identify new influence factors which are stored in the "Argumentation and Capitalization" database for later use. Each new influence factor is analysed to identify an existing or a new situation characterising

[1]The argumentation and capitalization aims at capturing the necessary argument, supported by underlying evidences, intended to justify the specification of the ODD attributes and attribute values

its context. Each new situation is stored in the "Argumentation and Capitalization" database and links between situations and influence factors in this database are created accordingly.
Finally from each selected influence factor (corresponding to relevant known situations) and each new influence factor, the ODD is refined to meet the ODD scoping objective through :

- Additional, update or exclusion of ODD attributes and/or attribute values
- Constraints between ODD attributes and/or attribute values
- Update Part of envisioned solution

Involved actors

- The engineering team

Inputs

- Description of the existing or envisioned solution
- The ODD scoping objective
- ODD attributes, sub-attributes and/or attribute values from the "Structured ODD" database
- Known situations from the "Argumentation and Capitalization" database

Activities

- Select relevant known situations
- Identify influence factors
- Identify new situations
- Add ODD attribute and/or attribute value
- Update ODD attribute and/or attribute value
- Exclude ODD attribute and/or attribute value
- Define constraints between ODD attributes and/or attribute values
- Update part of envisioned solution

Outputs

- Additional, updated and excluded ODD attributes, sub-attributes and/or attribute values into the "Structured ODD" database
- Constraints between attributes and/or attribute values into the "Structured ODD" database
- Update request on part of the envisioned solution
- Identified influence factors and situations into the "Argumentation and Capitalization" database

B.1.4 ODD consolidation

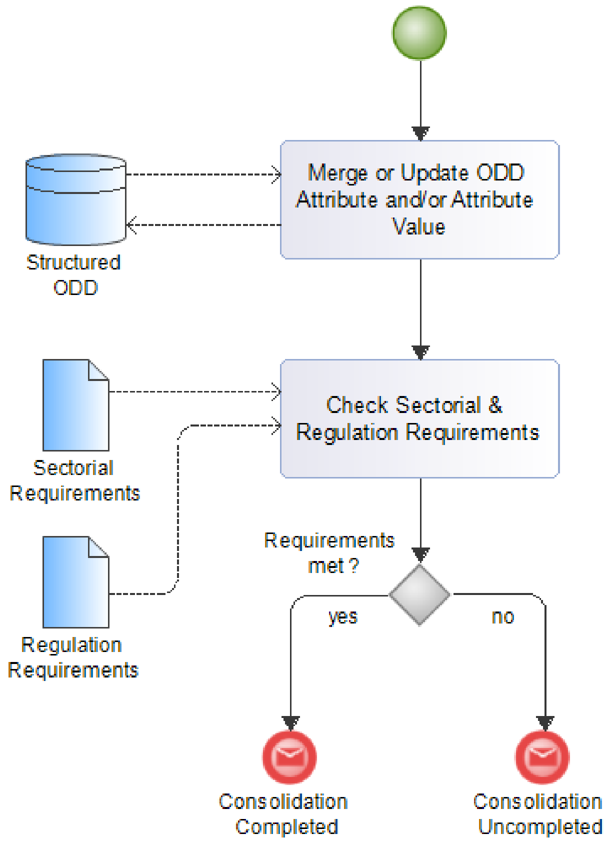

Figure B.5: ODD consolidation

Description

This step covers a twofold consolidation as presented in Figure B.5. First, the ODD attributes and attributes values from the "Structured ODD" database are retrieved to identify potential redundant or overlapping elements. The identified attributes or attribute values are updated to obtain a complete but concise ODD. It consists in defining the optimum limits for the ODD attributes consistent with the operational goal of the system feature. This gathering also helps reduce the number of ODD attributes to handle while maintaining the current limits.
Second, the resulting ODD is checked against sectorial requirements and regulation requirements corresponding to the system of interest, as described in the analytical approach from Adedjouma et al. (2022). It is worth noting that these requirements should have already been taken into account in the definition of the existing or envisioned solution for the system of interest.

Involved actors

- The engineering team

Inputs

- ODD attributes, sub-attributes and/or attribute values from the "Structured ODD" database
- Sectorial requirements
- Regulation requirements

Activities

- Merge or update ODD attributes and/or attribute values
- Check sectorial and regulation requirements

Outputs

- consolidated ODD attributes, sub-attributes and/or attribute values into the "Structured ODD" database
- The consolidation status report (completed or uncompleted)

B.1.5 ODD business or operational relevance verification

Description

The last step of the ODD process (see Figure B.1) aims at verifying that the consolidated ODD satisfies the customer needs. In other words, does the consolidated ODD represent a satisfying trade-off for its expectations. Otherwise, it is required to re-adjust the ODD scoping objective and re-perform the ODD definition process accordingly.

Involved actors

- The engineering team
- The customer or his representative

Inputs

- ODD attributes, sub-attributes and/or attribute values from the "Structured ODD" database

Activities

- Verify ODD business or operational relevance

Outputs

- The verification report

B.2. Example of resulting ODD

This section presents an excerpt of the ODD corresponding to the Naval Group use case LCMA (Lutte Contre les Menaces Asymétriques). LCMA is an asymetric warfare system for a surface ship. It is based on a panoramic video surveillance for detecting and identifying different kinds of threats. Please refer to the following article for a more detailed description of the system.
Figure B.6 and Figure B.7 presents as UML object diagram an excerpt of the LCMA ODD at the initialization and the refinement phase, respectively. The modelling are based on the

ODD metamodel. Note that the situation concept is not represented in this ODD as it has been introduced after the experimentation.

In the following subsections, we present the ODD initialization and ODD refinement results. For confidential reason, we are not detailing the ODD scoping objective, ODD consolidation and ODD business and operational relevance stages as they include sensitive data (customer needs and requirements) and proprietary knowledge (ODD values) of Naval Group.

ODD initialization

During the ODD initialization, customer expectations have been captured as ODD attributes and organized as shown in Figure B.6 in the 3 different categories :

- Scenery : the ship can be in the deep sea, i.e., Haute Mer or in the shore, i.e., Zone Littorale/Portuaire.
- Dynamic Elements capture the different surface threats - e.g. Voilier, Boutre, Jet Ski - and aerial threats - e.g. Chasseur, Helicoptere, Avion de patrouille.
- Environmental Conditions capture the meteorological conditions, the sea state and the visibility zones. For example, the LCMA solution implies different visibility zones such as "Zone de recouvrement" and "Zone normale".

ODD refinement

During the ODD refinement, influence factors are identified from the ODD attributes such as :

- the "seabed complexity" related with the ship position
- the "availability of data" (video recording) with regards to the meteorological conditions

Hence, ODD constraints may be induced from an influencing factor, e.g. the "meteorological constraint" (Contrainte Météo in Figure B.7) limiting the Operational context to "clear weather" is induced from the available data (Availability of data in Figure B.7).

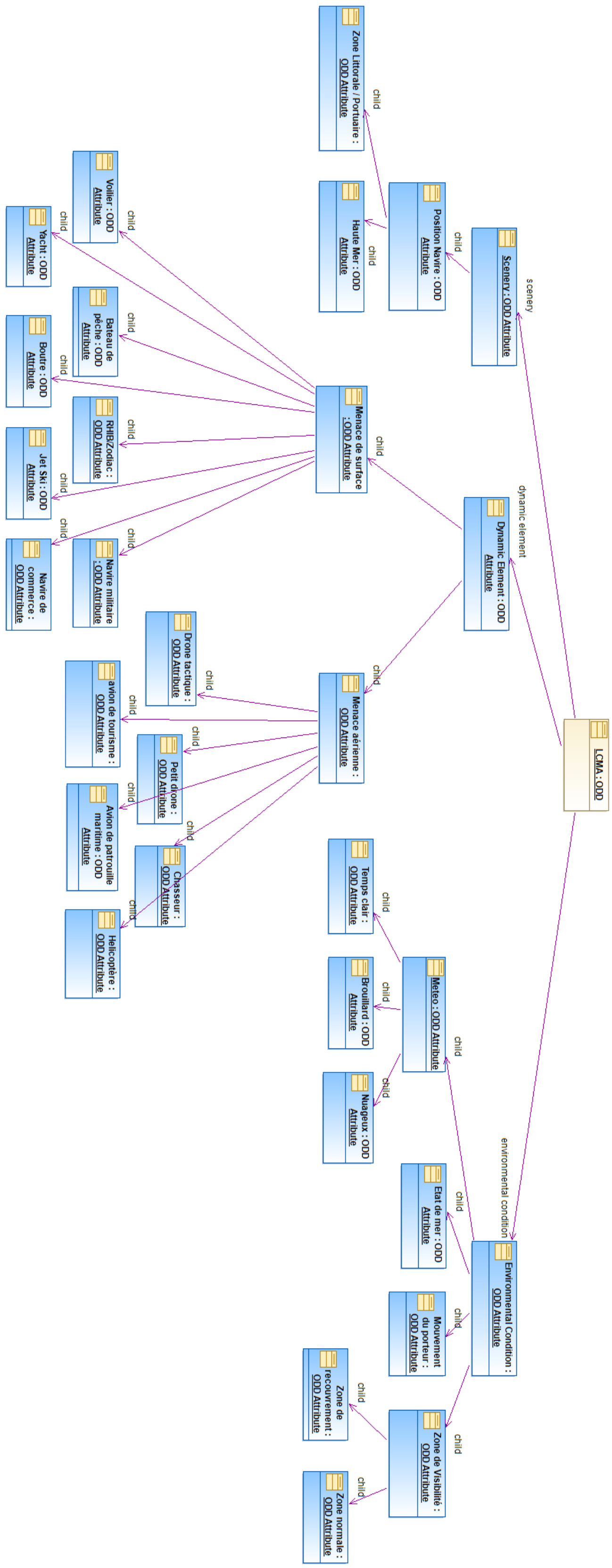

Figure B.6: LCMA ODD at the initialization stage

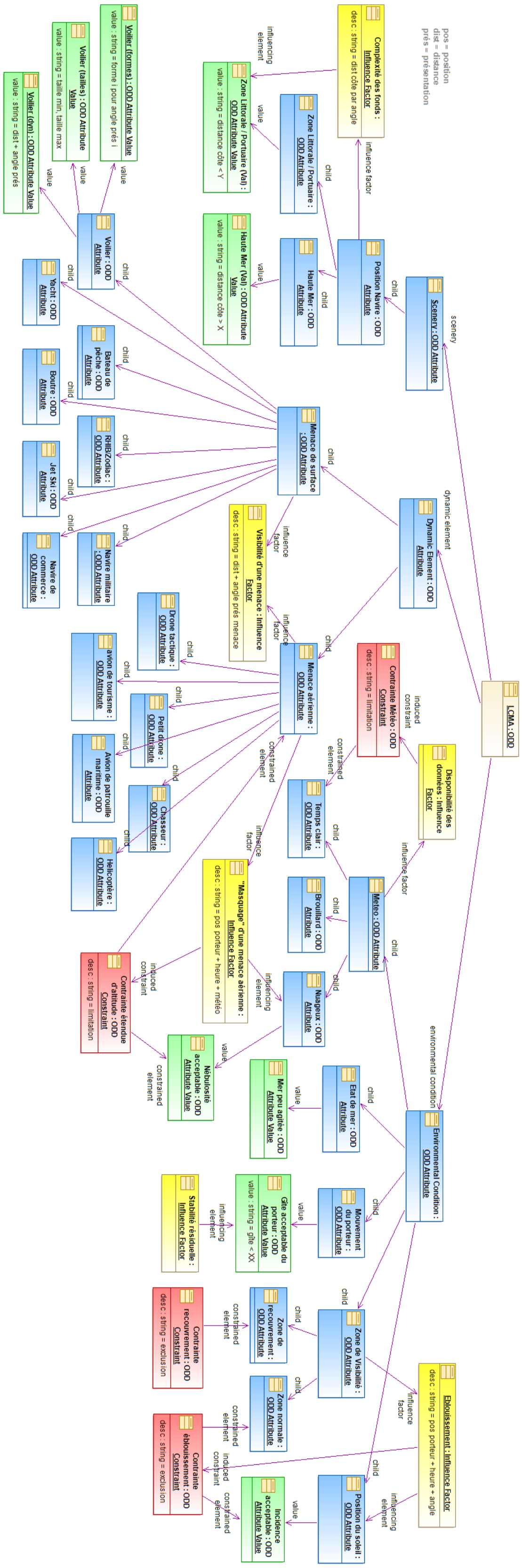

Figure B.7: LCMA ODD at the refinement stage

C. Link of ODD process with End-to-End Approach

The goal of this section is to highlight how the developed ODD process can interact with the proposed End-to-end approach proposed by the confiance.ai program. It also shows that although this deliverable is focused on the suggested ODD approach worked on in Batch 3, also called "systematic ODD approach", a certain number of criteria may lead the user to choose between this Approach and the one developed in Batch 2 called "targeted ODD approach". We will see in the following subsections what are the expected activities to perform and what are the prerequisites in order to select an approach.

C.1. Integration of the ODD analysis in the End-to-End approach

The role of the ODD analysis is to help define the appropriate operating conditions upon which the automated system is expected to function properly. It relies on the preliminary knowledge of the design of the expected solutions, as well as on the needs specified by stakeholders, that are subject to a trade-off.
For this reason, starting the specification of the ODD as early as the Operational Analysis is key in order to provide a framework of study that will be tested against the proposed ODD. Building an ODD is an iterative process, since the knowledge of the system of interest is built and improved during its development. This means that potential choices made regarding the ODD could be invalidated or challenged depending on the revealed implementation of the system known at a later stage. Additional insights regarding the choices made in the ODD could be brought up thanks to expertise feedback, which can help improve further or bring additional evidence to the table. Figure C.1 highlights how the ODD activities of this deliverable can be integrated in the End-to-End Approach. Theses activities span from the Operational Analysis to the first steps of the ML Component Specification. They are also subject to iterations during the development of the system.
We expect that a preliminary ODD will first be devised in order to help capture at a high level the main operating conditions for the studied system. This is key in order to ensure that the engineering teams (Safety, Cybersecurity, etc.) are provided with a unique vision to work on at the beginning. This preliminary ODD will then be iterated on, as the resulting requirements from this preliminary ODD are refined and allocated to lower-levels components. As an example, excluding the rain from a preliminary ODD implies to define several things:

- How can we detect rain?
- What are the consequences of rain on the system operation?
- What are the characteristics that make rain unacceptable as a weather condition in the ODD?

In the context of an automated driving system (ADS), we know that cameras and moisture sensors, combined with weather data, can be used to detect the presence of rain. Rain is often a challenge for ADS as it limits the visibility of cameras, and decreases car grip on the road, which could be hazardous at high speeds. However, rain has more precise characteristics such as the duration of the rain episode and the level of rainfall, that must be considered in order to identify if raining systematically entails an ODD exit or not. A short episode (less than 10 seconds) of light rain is not enough to be considered as hazardous for the ADS. But it is the goal

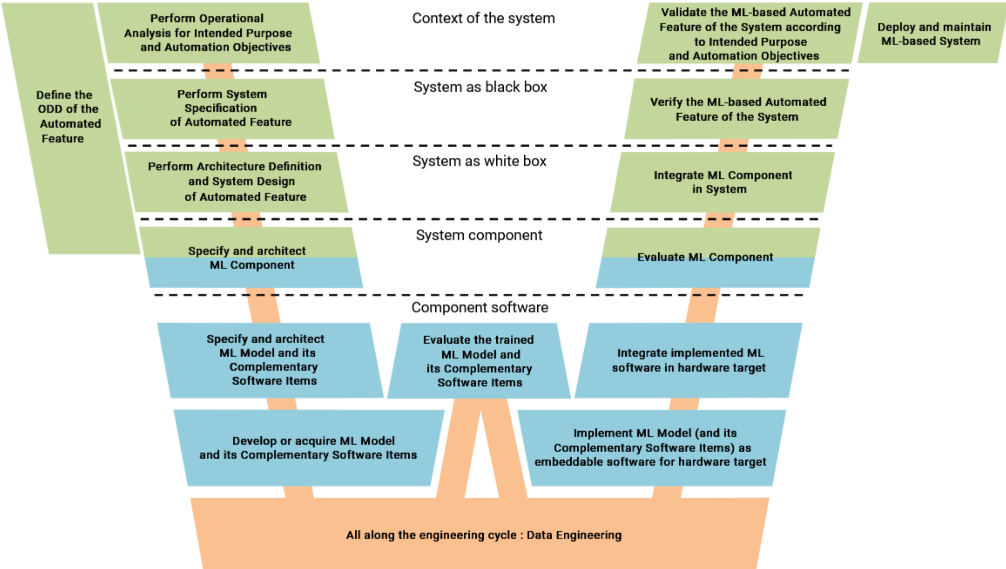

Figure C.1: End-to-End W cycle with integration of ODD activities

of in-depth ODD analyses to help identify these sub-attributes and define their acceptable ranges within or outside of the ODD.

C.2. Positioning with regard to the the targeted ODD approach from Batch 2

The ODD Approach currently developed in this document is called "Systematic ODD Approach based on taxonomy". This name highlights the role played by a systematic in-depth analysis of the main components of the ODD based on an existing taxonomy. This taxonomy helps to give a guidance on presenting the stakes of the ODD to stakeholders that are not familiar with this notion.
As introduced in Section B, Confiance.ai first developed an analytical approach for the ODD that served as a pillar to build the Systematic ODD Approach. This first approach, fully dubbed "Targeted Analytical Approach", has been presented in Batch 2 in a previous EC6 deliverable. For this reason, it will not be presented again in this deliverable, but reference can be found Adedjouma et al. (2022). It aims to build an ODD by identifying key topics for the studied systems and by relying on known expertise and feedbacks. Depending on the development context, Confiance.ai has identified that both the Systematic ODD Approach and the Targeted ODD Approach could coexist, based on several criteria of selection.

C.3. Criteria of selection of both ODD approaches

Depending on the development context of the stakeholders, we recommend the users of this method to choose one ODD approach rather than the other, as they may better suit their needs. This recommendation is based on three criteria:

- Amount of available feedback: the targeted approach relies on amounts of industrial knowledge (industrial state of practice), expertise feedback and field feedback based on the preliminary design of the envisioned system and existing reference system. Thus, having access to this data is a prerequisite in order to apply this approach. If this is not the case, the Systematic approach may be preferred instead.

- Level of expertise of the stakeholders with automation and ODD notions: the Systematic is appropriate when dealing with stakeholders that have no to little knowledge on ODD or the automation of their studied system beforehand. It serves as a guidance on how to capture relevant ODD information that could quickly serve as a basis for discussion before more in-depth analysis. On the contrary, the Targeted Approach can be considered with teams that have a high level of expertise with these subjects, and thus can help identify quickly in the raw topics the ones that are relevant for building a proper ODD.

C.4. ODD meta-model

Since the Systematic Approach is based on the Targeted Approach, they share similar notions. The similarities are presented in the following metamodel:

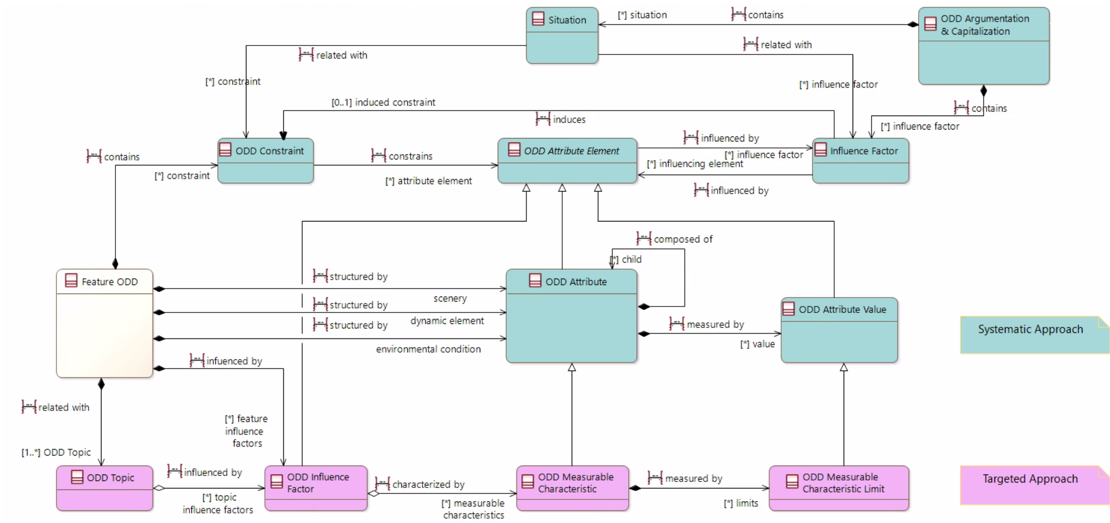

Figure C.2: ODD meta-model for Systematic and Targeted Approach

Figure C.2 presents the meta-model used for the ODD description in both ODD Approaches. As depicted in the corresponding UML class diagram, the meta-model comprises the following metaclasses, starting with the Systematic Approach (in blue on the diagram) :

- Feature ODD : the metaclass for an ODD. An ODD comprises a root attribute for scenery, dynamic element and environmental condition and also comprises constraints.
- ODD attribute : the metaclass for an attribute. An attribute is a special kind of attribute element and may have child attributes and attribute values.
- ODD attribute value : the metaclass for an attribute value. An attribute value is a special kind of attribute element for defining a measurable value.
- ODD Constraint : the metaclass for a constraint constraining multiple attributes and/or attribute values.
- Influence Factor : the metaclass for an influence factor induced from and influencing one or several attribute(s) or attribute value(s). An influence factor may induce a constraint.

- Situation : the metaclass for a situation describing the context of one or several influence factor(s) or the context of one or several constraint(s).
- ODD Argumentation and Capitalization: the metaclass for a database used for evidence by integrating both Situations and Influence Factors.

The justification for an attribute, an attribute value, a constraint or an influence factor may be captured through their respective rationale property.

In addition to these classes, other classes related inherited from the Systematic Approach can be defined for the Targeted Approach (in pink on the diagram):

- ODD Topic: the metaclass for a topic that cover either operational scenarios (similar to Situations in the Systematic Approach) or intrinsic (technological) limitations to the system (such as sensors limitation, choice of sensors, etc.).
- ODD Influence Factor: the metaclass for an influence factor that is derived from an ODD topic, and that is one of the three components of the ODD in the Systematic Approach. ODD Influence Factor is an instance of ODD Attribute Element.
- ODD Measurable Characteristic: the metaclass for a measurable characteristic that is derived from an ODD Influence Factor, and that is one of the three components of the ODD in the Systematic Approach. ODD Measurable Characteristic is an instance of ODD Attribute.
- ODD Measurable Characteristic Limit: the metaclass for the limit of a measurable characteristic that is derived from an ODD Measurable Characteristic, and that is one of the three components of the ODD in the Systematic Approach. ODD Measurable Characteristic is an instance of ODD Attribute Value. It can be a value or a range of values.

D. ODD Engineering process through the system components lifecycle

The level of detail of the ODD is defined according to the targeted audience, resulting to informal ODD descriptions (that can potentially be incomplete and ambiguous) to more structured and formal description that can be used for testing and validation purpose. In this section, we propose an approach to refine an ODD from the early engineering phases to reduce ambiguity and incompleteness until a machine-readable stage where it can be used to support the engineering activities such as safety analysis or V&V.
The approach is at an early stage of proposal. Indeed, we further have to validate its soundness through expert review and evaluation on industrial applications.

D.1. Overall ODD engineering lifecycle

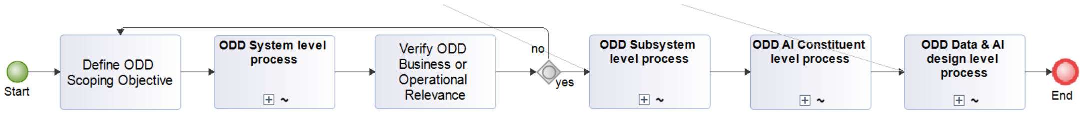

Figure D.1: ODD refinement process through engineering lifecycle

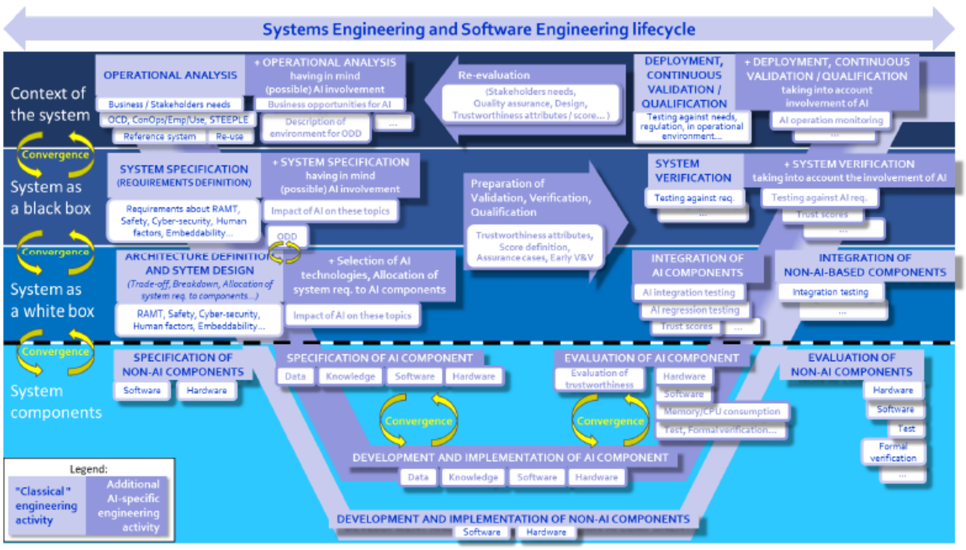

Figure D.2: Confiance.ai End-2-End process from Confiance.ai EC2 (2022)

Figure D.1 presents the overall process for refining the ODD through the engineering lifecycle. The process comprises 6 main steps that one can map to the different engineering levels defined

in the confiance.ai End-2-End process[1] (see Figure D.2 excerpt from the end-to-end approach for engineering trusted AI-based systems Confiance.ai EC2 (2022) report).
The first 3 processes are based on the activities defined in the generic ODD definition process presented in Chapter B, namely:

- the ODD scoping objective definition process that refers to the first activity defined in the generic ODD process. This activity happens during the "Context of the System" phase of the End-2-End approach with the operational analysis
- the ODD system level process that is mapped into the "System as a Black box" phase of the End-2-End process, and contributes to the system specification. This process includes as sub-activities the ODD Initialisation, Refinement and Consolidation of the ODD generic definition process of Chapter B
- The ODD business & operational relevance verification is a verification activity that also come during the "System as a Black box" phase of the End-2-End process.

As depicted in Figure D.1, in case the system level ODD satisfied the customer expectations, some subsequent refinements can be pursued at the lower level engineering phases to take into account specific requirements and constraints pertaining to the related engineering phase:

- The ODD subsystem level process aims at refining the prior ODD during the "System as a white box" phase of the End-2-End approach
- the ODD AI constituent level process aims at further refining the ODD during the "AI component specification" phase of the End-2-End approach
- the ODD Data & AI Design level process is the final refinement stage that is conducted during the "AI component development and implementation stage of the End-2-End approach

Each of these refinement stage processes includes an ODD refinement and an ODD consolidation as per the ones performed during the ODD system level process. However, they have the particularities to consider additional sources of specification and requirements depending of the engineering phase. Besides, during the consolidation activities, one must ensure also that the refined ODD is still in conformance or compliant to the ODD defined at the upper engineering phase (see the convergence loop on the End-2-End process). Indeed, the ODD refinement during the development cycle should not have an impact on the conformity with the primer specification, i.e. the same service must always be guaranteed in the refined ODD perimeter. However, in some cases, compliance with the initial set of requirements cannot be satisfied even after refining the ODD (due to stringent restriction of the scenario space). In this case, the system specification must be revised in agreement with the business stakeholders.
In the subsequent sections, we give some details about the expected objective at each refinement phase.

D.1.1 ODD process at (sub-)subsystem engineering level

Description

The main use of an ODD as described at this stage is to aid defining the requirements for the architecture and design of the system. The objective is to refine the ODD attributes and constraints with regard to the specifications that fall to this level of development. Each parameter that characterizes an ODD attribute must be more precisely detailed and the possible range of

[1] For further details on the End-2-End process, please refer to the end-to-end approach for engineering trusted AI-based systems Confiance.ai EC2 (2022) document

their values (qualitative and/or quantitative) must be refined and documented - as well as their possible combination, when technically relevant - with regard to the technical constraints of the system's components.
The ODD attributes refinement based on architecture and design choices does not mean that they cannot be defined at the system level before these choices are made. Thus, a system aimed at detecting pedestrians necessarily mentions "pedestrian" in its specification. The pedestrian can be identified from the system specification as an ODD attribute, while the precise nature of the interactions between pedestrians and the system (radiative (image, radar, heat), or mechanical) could be determined later during the design and further constrained the ODD attribute. It is up to the stakeholders intervening at that level of the development cycle to take into account new restrictions imposed, for example, by design, operational and technical constraints, for the refinement of the ODD.

Involved actors

Multidisciplinary team involving the system and specialist engineers, the system and component Architect.

Inputs

- ODD specification from the upper engineering level
- System requirements, architecture and design information necessary from upper engineering phase. This may include specification of the sensors, actuators, etc. (to be) used by the AI system.
- Technical constraints (design, hardware, interface) off required components or out-of-the-shell products to be embed in the AI system
- Architectural & design choices
- Applicable standards
- Technology readiness
- Known constraints [2]

Activities

- ODD refinement process (see Section B.1.3)
- ODD consolidation (see Section B.1.4), including a verification that the refined ODD conforms to the defined ODD at the upper level

Outputs

- A refined ODD attributes, sub-attributes and/or attribute values into the "Structured ODD" database, relevant to the architectural and design choices.
- A verification report

[2]Known constraints can be related either to the application domain providing information on operational constraints, or related to technical domain providing information on feasibility and technical limitations.

D.1.2 ODD process at AI-based component level process

Description

The main use of an ODD as described at this stage is to aid defining the requirements for AI component specification. To refine the ODD, one may take into account the design and AI implementation choices. Some constraints resulting from the design and implementation must be reinterpreted in environmental terms as ODD attributes/ sub attributes and values.

Involved actors

Engineer specialist , SW architect, SW/HW engineer

Inputs

- ODD specification from the upper engineering level
- AI-based System/subsystem requirements, architecture and design information from upper engineering phase.
- HW specification
- Safety & performance requirements
- Technical design constraints for the AI-based subsystem
- Technology readiness
- Recommended AI implementation choices

Activities

- ODD refinement process (see Section B.1.3)
- ODD consolidation (see Section B.1.4), including a verification that the refined ODD conforms to the defined ODD at the upper level

Outputs

- A refined ODD attributes, sub-attributes and/or attribute values into the "Structured ODD" database, relevant to the performance requirements, and the AI design choices
- A verification report

D.1.3 ODD process at Data & Model engineering level

Description

The ODD refinement at the Data and AI Design development processes is closely related to the data collection and the known properties of the AI model, in particular the constraints on the model's input parameters. Each ODD attribute must be traceable to the datasets' elements, and their values must satisfy the AI training and implementation constraints. For example, to define a satisfactory dataset where the "observability property" is important, a reinterpretation of some ODD attributes may be necessary when those attributes do not refer to observable concept/properties through images, e.g. the percentage of humidity in the atmosphere.
The main usage purpose of an ODD as described at this stage is to aid validating the data and model engineering phase. This will be the as-defined ODD that must be used for subsequent V&V activities.

Involved actors

Data engineering and AI engineering team

Inputs

- ODD specification from the upper engineering level
- AI-based component requirements, architecture and design information from upper engineering phase.
- Available datasets
- Data related constraints
- Expected AI model
- AI properties constraints
- Operational context specification

Activities

- ODD refinement process (see Section B.1.3)
- ODD consolidation (see Section B.1.4), including a verification that the refined ODD is conforms to the defined ODD at the upper level

Outputs

- A refined ODD attributes, sub-attributes and/or attribute values into the "Structured ODD" database, relevant to the datasets and training model constraints
- A verification report

E. ODD quality properties and relation to trustworthiness characteristics

This chapter presents the ODD desirable quality properties and relation to trustworthiness characteristics[1].

E.1. ODD desirable quality attributes

An Operational Design Domain (ODD) is intended to remain valid throughout the entire lifecycle of the product in question. The description of the ODD is utilized and updated during various stages, such as product specification, safety analysis, system design, and validation. To assess the quality of the ODD, an ODD quality model can be employed. This model evaluates whether the ODD possesses key properties that can be categorized as follows:

1. Effectiveness: This category focuses on whether the ODD successfully fulfills its intent of defining the conditions under which the system should operate. Effectiveness can be further broken down into the following properties:
   - Completeness: The ODD covers all relevant conditions and scenarios necessary for the system's operation.
   - Accuracy: The ODD accurately represents the operating conditions and requirements of the system.
   - Consistency: The ODD is internally consistent, with no conflicting or contradictory information.
   - Operability: The ODD is acceptable by all the stakeholders.
2. Clarity: This category assesses how well the ODD describes the defined conditions in a comprehensive and understandable manner for all stakeholders involved. It ensures that the ODD is clear and unambiguous, facilitating effective communication and understanding. It comprises the concision, understandability and explainability quality properties.
3. Usability: This category focuses on the representation of the ODD and the information it carries, ensuring that they are suitable and adapted to the engineering activities that will utilize the ODD. It encompasses Verifiability, traceability and measurability, modifiability, reusability and maintanibility properties. The ODD should be easily navigable, organized, and structured in a way that supports the intended engineering tasks.

Further details on the individual ODD quality properties cited above was provided in deliverable Adedjouma et al. (2022) and the deliverable Confiance.ai EC6 (2023b) distributed to the standardization bodies.

By evaluating the ODD based on these categories, the quality of the ODD can be assessed, ensuring that it effectively defines the operating conditions, provides clear descriptions, and is usable in the relevant engineering activities.

[1]The ODD quality properties and relation to trustworthiness characteristics presented in this chapter were integrated in the scope of the CEN-CENELEC Joint Technical Committee 21 'Artificial Intelligence'/WG4- AI trustworthiness framework draft standard, and led to a New Work Item Proposal in the CEN-CENELEC JTC21/WG3, and to a Preliminary Work Item in the ISO/IEC Joint Technical Committee SubCommittee 42 Artificial Intelligence as well.

E.2. ODD relation to trustworthiness attributes

Trustworthiness characteristics can only be assessed if the Operational Design Domain (ODD) is clearly defined. The ODD specifies the conditions in the physical and cybernetic environment in which the AI sub-system is intended to operate. Many AI prototypes neglect to adequately describe their ODD or define it vaguely as the domain covered by the data used during training. However, it is crucial to analyze and understand if the real environment aligns with the training data distribution in order to predict the performance or risks of using the system in an operational context.
Therefore, it is necessary to either extract a characterization of the situations in which the trained model performs from the training data or explicitly define the situations in which the model should perform. Subsequently, representative training datasets should be constructed to cover these defined situations. The assessment of many trustworthiness characteristics relies on the definition and quality of the ODD. It is essential to establish a clear understanding of the ODD in order to evaluate and ensure the trustworthiness of the AI system.

AI system quality management

From a quality management perspective, the ODD entails the following requirements:

- ODD quality is an essential characteristic of an AI system quality model.
- The ODD of an AI system must be specified and verified in accordance with an agreed-upon quality model.
- The ODD of an AI system should be included as part of its technical documentation.
- Monitoring and reporting whether an AI system stays within its ODD boundaries should be mandated.
- Any instance of the AI system functioning outside its ODD, even if it has not resulted in an accident, should be treated as a major incident. It must be logged and reported. The logging process should provide an explanation of the environment and the reasons why the AI system deviated from its ODD.
- The ODD should be integrated into the risk management and quality management of an AI system.
- The conformity of an AI system to its ODD should be evaluated and assessed.

These requirements highlight the significance of the ODD in quality management and emphasize the need for adherence to the defined ODD in order to ensure the proper functioning and risk mitigation of an AI system.

Data quality

The ODD serves to define the boundaries of domains, both within and outside of it, where data will be required. By defining the ODD, data governance is facilitated. The ODD provides a framework for establishing the criteria used to assess data completeness, thus guiding the determination of the data coverage needed. In essence, the ODD plays a crucial role in data governance by delineating the scope within which data should be considered. It aids in setting the criteria for evaluating the completeness of data, ensuring that the necessary data coverage is achieved.

Dependability

Safety and security analyses require the definition of the system, especially its application domain, as provided by the ODD. The ODD serves as crucial input for these analyses. Additionally, to ensure the system's maintainability in the face of changing environments, it is necessary to consider the characteristics of the expected nominal environment conditions. Evaluating the impact of environmental evolution on the system relies on understanding these characteristics. By incorporating this information, the system can be designed and maintained to effectively adapt to evolving conditions. An example of relying on ODD for building a Safety argumentation is found in the UL 4600, which is a guideline suggesting what could be in the assurance case of an automated vehicle feature UL 4600 (2019). In this guideline, the ODD is identified as a key element to detail with aspects that are related to Safety.

Operability

Drawing an analogy with the concept of "design by contract," where a system guarantees functional behavior based on certain preconditions being met by the environment (inputs), the ODD serves as the precondition for the system's nominal functional behavior. The correctness of this behavior should be evaluated in situations that adhere to this precondition. The ODD defines the limits of the input space and contributes to defining the variability axes of these inputs.
In order to ensure trustworthiness, the accuracy evaluation of the system requires careful selection of test data that thoroughly cover this input space. To evaluate functional completeness, it is necessary to explore the functional scenarios that fall within the scope of the ODD. By considering and examining these cases, the system's functional coverage can be assessed.

Robustness

An essential aspect of robustness is the system's ability to handle unexpected inputs. This includes scenarios where inputs fall outside the expected range defined by the ODD. Understanding the behavior of the system beyond the boundaries of the ODD is crucial and must be carefully considered. Evaluating the system's performance and response in such situations is a key aspect of ensuring its robustness. By examining and assessing the system's behavior outside the ODD, potential vulnerabilities or limitations can be identified and addressed to enhance its overall robustness.

Human centered quality

The ODD serves as a means for users to comprehend the capabilities and expected behavior of the system. It enables users to understand the system's useful capacity and anticipate its behavior in different situations. The clarity properties of the ODD play a vital role in evaluating the usability and interpretability of the system. These properties contribute to the system's overall user-friendliness and the extent to which users can interpret and make sense of its functioning. By ensuring clarity in the ODD, the system's usability and interpretability can be enhanced, allowing users to effectively engage with and utilize the system.

Ethics

Transparency is an important attribute that depends on the documentation of the ODD and the justification of the choices made during its definition. By thoroughly documenting the ODD and providing justifications for the decisions made, transparency can be enhanced, allowing stakeholders to understand the system's objectives and constraints.

Fairness and absence of bias are often associated with the analysis of characteristics in the data that should remain neutral with respect to the system's outcomes. The definition of the ODD aids in identifying and focusing on these characteristics, ensuring that data collection efforts reflect this neutrality. By considering fairness and bias-related considerations within the ODD, steps can be taken to minimize potential biases in the system's functioning and decision-making processes, thereby promoting fairness and impartiality.

Accessibility

The ODD document should be distributed to stakeholders in a timely manner, ensuring that they have access to it when needed. Effective communication of the ODD is essential to ensure that stakeholders understand its content and implications. The format in which the ODD is communicated should be accessible to all stakeholders, taking into consideration their needs and preferences. Clear and concise explanations, visual aids, or other appropriate means of communication can be employed to facilitate understanding and engagement with the ODD. By ensuring timely distribution and effective communication, stakeholders can actively participate and contribute to the development and implementation of the AI system in line with the defined ODD.

Monitorability, Controllability, Record-keeping

One of the crucial aspects to monitor is that the system operates within its specified ODD, as this condition ensures the system's performance. This monitoring relies on two key properties of the ODD: effectiveness and measurability. Effectiveness ensures that the ODD accurately describes the conditions in which the system is capable of functioning. Measurability implies that the constraints expressed by the ODD on the environment are defined in terms of measurable characteristics, and the evaluation procedures for these constraints are specified.

Monitoring efforts should focus on capturing relevant data and performance metrics within the defined ODD to identify anomalies or deviations from expected behavior. This may involve monitoring specific sensors, internal states, or decision-making processes within the operational domain. By monitoring the system's adherence to the ODD, potential violations of the system's safety requirements can be detected before significant losses occur, allowing for the implementation of necessary mitigation strategies. Additionally, ODD monitoring provides the benefit of collecting data that can be used to identify areas for improvement or optimization, leading to iterative design iterations and enhancing the system's performance over time.

F. ODD usage in confiance.ai End2End Use cases

This section reports on the experience feedback on the proposed approaches on industrial use cases, in particular how the ODD definition processes interplay in practice with other engineering processes developed in the framework of the program.

F.1. ODD (system level) definition

We work with our industrial partner Valeo to define the ODD for an Rear Automated Parking (RAP) System. The RAP is an autonomous car-maneuvering system that moves a vehicle from a traffic lane into a parking spot to perform parallel, perpendicular, or angular parking. The feature works when one is in reverse, and the driver must remain in the driver's seat, ready to take control if necessary. The feature uses ultrasonic sensors, and cameras to take autonomous control of specific parking tasks or the entire parking exercise, helping drivers safely and securely store their vehicle without damaging it or other cars parked nearby. The parking maneuver is achieved by means of coordinated control of the steering angle and speed which takes into account the actual situation in the environment to ensure collision-free motion within the available space. One can turn this feature on and off using via a button on the dashboard or accessed via the infotainment screen in the vehicle settings menu.
To define the ODD of such a system, we follow the reasoning process as described in Chapter B. We use several workshops to end up with the ODD as shown in tabular format in Figure F.1.

Category	Attribute	Sub-attribute	Sub-attribute	Sub-attribute	sub-attribute	Capability	Component AI
Scenery	Zone	regions or states				Anywhere on Earth	Yes
		public area					Yes
	Drivable area	Type	Parking	underground		Y	Yes
				OnStreet		Y	Yes
				surface		Y	Yes
		specification	longueur			5m + 0,10m	No
			largeur			2,3m +0,10m	No
			hauteur			3m	No
			orientation /Configuration	Niche - Creneau		Yes	No
				Battle - Bataille		Yes	No
				Slopping - Epi (45°, 60°,		No	No
			ground marking	PMR, Electrical, Priority		No	Yes
			stationary or moving agent and special			No	No
		geometry	vertical plane	up-slope		up to 15 cm	No
				down-slope		up to 15 mm	No
				level plane		yes	No
			transverse plane	pavement (pavé, dallage)		yes	Yes
			horizontal plane	straight lines		Yes	No
				curves		up to 1/500 m (radius of curvature)	No
		surface	type	uniform (asphalt)		Yes	Yes
				segmented (e.g. concrete		Yes	Yes
				loose		No	Yes
				vegetation		Yes	Yes
			features	temporary road structures		No	Yes
				cracks, potholes, ruts or		Yes, from 5 cm in-depth	Yes
			induced road surface conditons	icy;		Yes	Yes
				ii) flooded roadways;		up to 10 mm	Yes
				iii) mirage;		Yes	Yes
				iv) snow on drivable area:		up to 10 mm	Yes
				v) standing water;		up to 10 mm	Yes
				vi) wet road;		up to a limit	Yes
			temperature			up to 45 celsius	No
		signs	Information signs			yes	Yes
		Edge	fixed road structure			Yes	Yes
			parked vehicle			Yes	Yes
			line markers			Yes	Yes
	fixed road structure	street furniture	bench, Street lamp, motorcycle stands,			Yes	Yes
		shoulder, sidewalk				up to some minimum height	Yes
		Vegetation				up to some minimum height	Yes
		special facilities	Caddy shelters, storage space for bicycle,			up to some volume (HxLxl)	No
		Wall, Security fence				up to some volume (HxLxl)	No
		Ceiling				No	Yes
	temporary road	road signage,				Yes	Yes
Environmental conditions	weather	type	rain			up to 50 mm/h	Yes
			wind			up to 10.7 mls	Yes
			snow			up to 42 mm/h	Yes
			hail			up to 42 mm/h	Yes
			solar flares			Yes	Yes
		temperature				"-20 to 50 celsius"	No
		humidity				between 0 to 100%	No
	illumination	illuminance				> 5 lux	Yes
		sun position	right, front, left, rear			Yes	No
		sun elevetion				up to some limit	No
		cloudiness				< 7 oktas	Yes
		type	natural (sun), artificial			Yes	Yes
	Particulates	Type	smoke, pollution, dust, leaves, mist, fog,			Yes	Yes
		visibility				up to 1 km	Yes
Dynamic elements	subject vehicle	size	L x Hx l				Yes
		speed				< 20 km/h	Yes
	Traffic	category	vehicle	type	priority vehicles	No	Yes
					private (standard, truck,	< 20 km/h	Yes
					twoand 3-wheels	< 20 km/h	No
					special (tractor	No	Yes
				speed		yes, when < 20 km/h	No
				size		4,7 x 2 x 1.7 (LxHxl)	No
				flow rate		up to some limit	No
			agent	type	vulnerable road users	Yes	Yes
					animal		Yes
				density		up to 4 agents	Yes
				speed		up to some maximum speed	No
			special object	stroller		No	Yes
		volume (all type				Yes	No

Figure F.1: Excerpt of ODD definition for a Rear Automated Parking (RAP). Some of the capability values have been falsified for confidentiality reasons.

Even though, we did not strictly follow the iterative refinement process steps as described in Chapter D, we can relate some ODD attributes refinement to the targeted objectives of different engineering phases. Below, we explain how, theoretically, one can apply the iterative ODD refinement method on the RAP use case.
At system/feature level, we initialize the ODD from the PAS 1883 taxonomy British Standard Institution (2020). During the refinement stage, we exclude some scenery attributes, e.g., junctions, special structures and some roadway types. We may also exclude the parking attribute from scenery, because the illumination attribute from environmental conditions was found sufficient to capture the influence factor that impacts the need to consider the aforementioned scenery attributes.
We consider some sub-components constraints to refine other ODD attributes. Hence, the humidity attribute was identified as important to calibrate as it may impact the performance of the system at the perception and object recognition level. We also analyse the sensor technical limitation to capture a luminosity intensity below 5 lux to restrict the acceptable value for the ODD particulate visibility attribute. Such technical considerations can come at the sub(subsystem

engineering level).
At the AI-based component level, a sensor physical property, e.g., the camera latency was interpreted in terms of environment feature perturbation, e.g., a desirable level of vibration. Similarly, the computing power of the micro controller induces some limited time processing of the images (number of images per time unit) it receives. This constraint induces to define some maximum speed for the dynamic elements attribute, including the ego vehicle, for a better performance at the perception level. We accordingly refine the ODD attribute with such constraint.
At the Data and AI engineering refinement phase, the ODD attributes initially expressed in terms of temperature and humidity level in the atmosphere may be reinterpreted as the presence of water or ice on the road. Indeed, the temperature level cannot be observable on an image while the presence of water or rain may be. At this level, while the lane specification attribute was not found relevant at the feature level, it will be necessary, for datasets collection, to define some sub-attributes as dimension, type, color of the attributes

```
ADD Weather TO EnvironmentalConditions AS OperatingFeature
// define conditions for weather
ADD type TO Weather AS OperatingFeature
ADD rain TO type AS Property
MESURE rainfall rain Type Real UNITS SpeedUnitKind.kmh // change unit to mm/h
DETERMINE State allowedrainfall WHEN [ rain < 10 ]
ADD wind TO type AS Property
MESURE windspeed wind Type Real UNITS SpeedUnitKind.kmh // change unit to mls
DETERMINE State allowedwindspeed WHEN [ windspeed < 10]
ADD temperature TO Weather AS Property
MESURE degree temperature Type Integer UNITS TemperatureUnitKind.Celsius
DETERMINE State AllowedTemperature WHEN [-20 < temperature < 50]
ADD humidity TO Weather AS Property
MESURE percent humidity Type Integer // be able to define a percentage metric
DETERMINE State allowedHumidity WHEN [0 < humidity < 100]
//define conditions for particulates
ADD Particulates TO EnvironmentalConditions AS OperatingFeature
ADD type TO Particulates AS Property
ADD Visibility TO Particulates AS Property
MESURE mesure Visibility Type Real UNITS LengthUnitKind.cm // add km
//define conditions for Illumination
ADD Illumination TO EnvironmentalConditions AS OperatingFeature
ADD luminance TO Illumination AS Property
MESURE luminanceLevel luminance Type Integer UNITS IlluminanceUnitKind.Lx
DETERMINE State AllowedLuminance WHEN [luminance > 5]
ADD sunPosition TO Illumination AS Property
ADD sunElevation TO Illumination AS Property
MESURE orientation sunElevation Type Integer UNITS OrientationUnitKind.deg
DETERMINE State allowedOrientation WHEN [orientation > 60]
```

Figure F.2: Excerpt of a structured ODD with Papyrus4ODD tool

Note that such tabular format used in Figure F.1 falls short in capturing the dependencies between ODD attributes. For example, dynamic element "agent and special object" attribute is not acceptable in the ODD when agents or objects are present on a parking lot, or when the agent density is above some threshold. Similarly, it is not possible to define through the table the constraint that the parking specification is an acceptable attribute only when there is at least 3 identified parking edges. To resolve those shortcomings, we develop a tool support to help define an ODD using a structured and more flexible textual format (see Papyrus4ODD User Manual

Confiance.ai EC6 (2023c)), as depicted in Figure F.2. A summary about the RAP ODD defined through the tool is provided in the Confiance.ai EC6 (2023a)) report.

F.2. ODD usage at AI component level engineering

Different use cases in the confiance.ai program specified or used an as-specified ODD in their experimentation and evaluation of the program's methodological and technical artefacts. The program focusing mainly on the AI engineering workflow, we do not have industrial feedback on the ODD usage at (sub-)subsystem engineering level through the program's use cases experimentation. Renault defined an ODD and reused the identified constraints in subsequent activities related to data engineering and ML engineering. Naval UC defined an ODD following the Batch 3 approach. The ODD was manually defined, so the Papyrus4ODD tool was not used for the ODD formalization. During Batch 4, Valeo reused the ODD defined during batch 3 (see previous section). In particular, some ODD constraints were used to guide the definition of the data scope and elicitate data quality attributes during data engineering phase. Safran defined an ODD but they did not follow any confiance.ai approach. Indeed, their ODD definition in an excel file did not present the necessary information for an ODD formalization as required by confiance.ai methods and tool support.
The Figure F.3 summarizes the Workflow activities at AI component engineering level, where ODD and related constraints and concepts were used through the use cases.

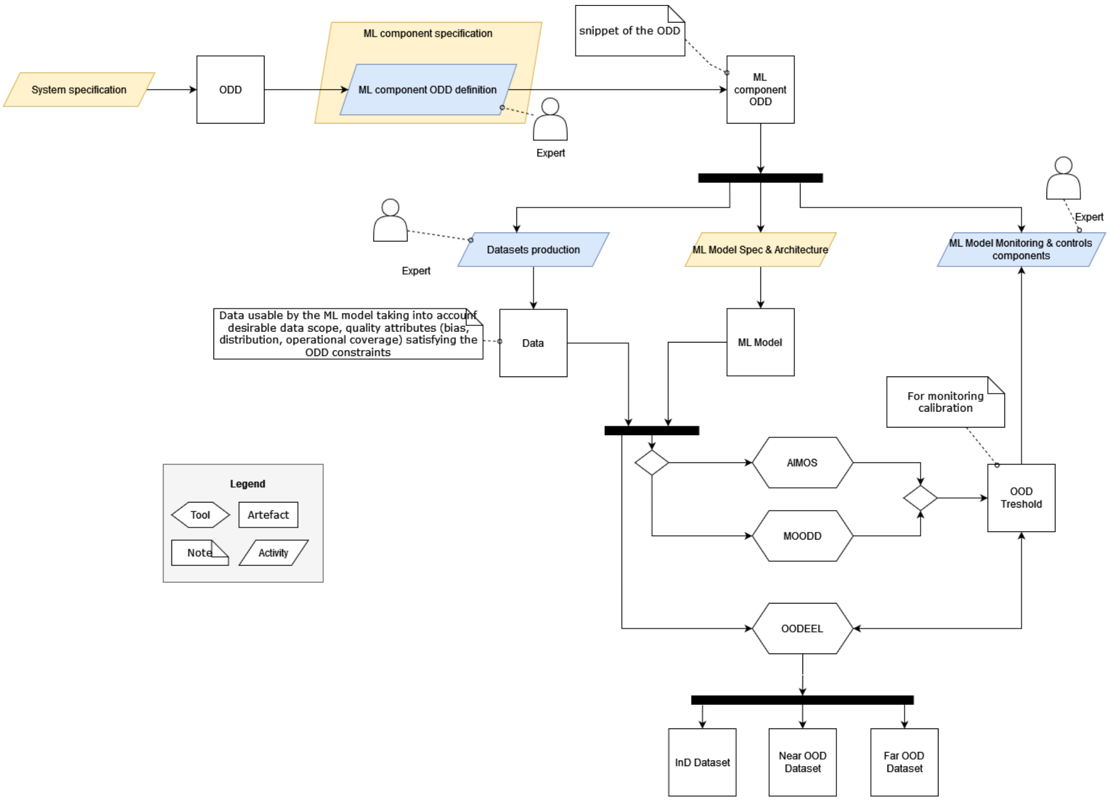

Figure F.3: Excerpt of the ODD usage workflow through component level engineering activities

As one can see, the ODD as-specified at system engineering level is used during the ML component specification, in particular for the ML component ODD definition activity, to define an ML component ODD, which is a snippet of the overall ODD that extracts only the constraints pertaining to the ML component. Note that there is no straight forward usage of the ODD (defined at system level) into the ML engineering activities. the UC confirms that indeed, the interpretation of ODD constraints from a system point of view into data or ML point of view required heavily expertise knowledge, e.g.:

- to identify which ODD constraints is applicable to data, or to AI model, or to monitoring component
- to refine an system level ODD constraints relying on operational context, into constraints on dataset, on AI model design, etc?

A (snippet of) ML component ODD artefact is further used for several activities: Data specification, ML Model Specification and Architecture, and ML Model Monitoring and controls components. These Data and ML engineering activities have strongly interrelated tasks, so they must be performed in parallel. Usage of the ML component ODD in these activities required also dedicated skills and expertise. For example, the main challenge of the Dataset specification with regards to ODD is to identify a set of images satisfying the ODD constraints. An expert has to reinterpret some ODD attributes into data quality properties, when those attributes do not refer to direct observable concepts/properties of images, e.g. the percentage of humidity in the atmosphere, the speed level of an object, the rain level, etc. Data and ML Model, when defined considering ODD constraints, enable to define OOD (Out-Of-Distribution) thresholds useful for the ML Model Monitoring and Controls activities. The ODD thresholds can be defined from robustness analysis with AIMOS or MOODD tool, by introducing various perturbations related to ODD constraints on the dataset, e.g. regarding the luminosity property. The OODEEL library, taking as input the ML Model and the Dataset can also be used to verify that the ML Model distinguishes Near and Far OoD from InD (In-Distribution) data, which helps to identify or validate existing OOD thresholds.

F.3. Thoughts about an ODD definition approach from datasets

Several industrial partners demand an approach for ODD definition from datasets. Our insight is that such an approach will not be efficient to define properly an ODD, and this for for several reasons. First, the baseline for an ODD definition at system level is the operational context consideration as perceived by an end user, while the baseline for data engineering activities are images. However, an image represents an abstract perception of a (partial) operational context and do not gather all physical properties that exist in the real operational context.
Secondly, as one has seen from the use cases experimentation's, in a top down approach as we recommend, only a snippet of the ODD definition is "used" for data engineering. So, in a bottom up approach (from data to system level ODD), one will not be able to identify all constraints pertaining to the complete system level ODD,
Thirdly, it will be hard to correctly define the real system level ODD constraint values from an operational point of view from datasets. The ODD snippet relevant for data engineering activities is defined based on some expertise skills that analyse the system level ODD constraints, identify the ones that may be relevant (mostly for the dynamic elements, the environmental elements), i.e., affecting data, and then interpret such constraints as related quality attributes on images. For example, some level of luminosity at operational stage may be reinterpreted as degrees of brightness on images. Identically, a level of humidity, the object speed level, or visibility distance at operational stage, can be affecting image quality attributes. This interpretation is necessarily project and expert knowledge dependent, and cannot be straightforward or automated. So, how

do one derive a correct corresponding level of luminosity, visibility parameter, sun parameters (orientation, position, elevation, etc.) from the image properties, knowing that a particular data quality attributes can be dependent of those operational properties, all together ?

G. Conclusion

The deliverable presents the methodological guideline for the ODD definition of an AI-based system through its lifecycle. It is of interest of any stakeholder that is involved in the user needs elicitation, specification, design, training, V&V and qualification of an AI system. The deliverable includes a general introduction that explains the need for a ODD definition for an AI-system. We have then a chapter that presents a generic process for an ODD definition, accompanied with a return of experience of its application at Naval Group. The next chapters presents the relation of the ODD process with the overall End2End engineering processes designed in the confiance.ai program, and the ODD refinement approach at different engineering phases through the AI system development lifecycle. From the feedback collected on industrial use cases developed in the framework of the program, we are enough confident that our ODD approach including the iterative refinement process is sound to address the particularities of the different engineering teams that may use the ODD artifact for their activities, in particular how the ODD definition processes interplay in practice with Data and ML engineering processes. The deliverable also includes the quality attributes to evaluate a defined ODD, and how the ODD would contribute to AI trustworthiness properties.

Bibliography

Adedjouma, M. et al. (2021). Risk management for ai-based system – the confiance.ai program.

Adedjouma, M. et al. (2022). Operational design domain: Methods and tools – the confiance.ai program.

British Standard Institution (2020). PAS 1883 Operational Design Domain (ODD) taxonomy for an automated driving system – Specification.

Confiance.ai EC2 (2022). Confiance.ai end-to-end approach for engineering trusted AI-based systems – v1.

Confiance.ai EC6 (2023a). ODD framework evaluation.

Confiance.ai EC6 (2023b). Operational Design Domain: definition, quality model and relation to trustworthiness.

Confiance.ai EC6 (2023c). Papyrus 4 ODD User Manual.

ISO 34503:2023 (2023). Road Vehicles - Test scenarios for automated driving systems - Specification for operational design domain . Standard, International Organization for Standardization, Geneva, CH.

SAE Mobilus (2018). SAE J3016 Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles.

UL 4600 (2019). Proposed First Edition of the Standard for Safety for the Evaluation of Autonomous Products, UL 4600 . Standard, Underwriters Laboratories.

Zhang, R., Albrecht, A., Kausch, J., Putzer, H. J., Geipel, T., and Halady, P. (2021). Dde process: A requirements engineering approach for machine learning in automated driving. In *2021 IEEE 29th International Requirements Engineering Conference (RE)*, pages 269–279.

Title:	Methodological Guideline for Operational Design Domain

Keywords: Operarional Design Domain (ODD)

The deliverable is about the methodological guideline for the ODD definition of an AI-based system through its lifecycle. It presents a generic process for an ODD definition, as well a refinement process of the ODD at different engineering phases through the AI system development lifecycle. The deliverable also includes the Key performance indicator to evaluate the quality of a defined ODD, as well as the ODD would contribute to AI trustworthiness properties. The guideline is of interest of any stakeholder that are involved in the user needs elicitation, specification, design, training, V&V and qualification of an AI system.

Our partners: