Towards Continuous Simulation Credibility Assessment

Maurizio Ahmann[1] Van Thanh Le[1,4] Frank Eichenseer[1,2] Frederik Steimann[1] Martin Benedikt[3]
[1]SETLabs Research GmbH, Germany, {Maurizio.Ahmann,VanThanh.Le,Frank.Eichenseer, Frederik.Steimann}@setlabs.de
[2]iCONDU GmbH, Germany, Frank.Eichenseer@icondu.de
[3]Virtual Vehicle Research GmbH, Austria, Martin.Benedikt@v2c2.at
[4]Free University of Bozen-Bolzano, Italy, vanthanh.le@stud-inf.unibz.it

Abstract

With the growing demand for virtual-informed decision-making in the development process of many engineering domains, the evidence in simulation results and thus simulation credibility becomes a critical aspect, in particular for releasing safety-relevant systems. However, simulation credibility is often interpreted to be of subjective nature. This paper summarizes basic assumptions for enabling the expression of credibility for building evidence in a more objective way. Based on these considerations, a concept is proposed that allows for an approximation of the credibility of simulations according to a discrete scale. The work is concluded by providing an implementation concept for a continuous simulation credibility assessment using a layered standard on top of the System Structure & Parameterization specification.

Keywords: credibility, credibility assessment, verification, validation, traceability

1 Introduction

With ever growing complexity of modern products across different industries and domains, the simulation of cyber-physical systems takes on an increasingly important role in the decision-making process. This can become critical for safety-relevant applications, like the simulation of automated driving systems (Knauss, 2017; Koopman, 2017), or simulation of medical devices (Rogers, 2019; FDA, 2021), where wrong decisions may have fatal consequences.

To mitigate the risk of making unreliable decisions based on insufficiently valid simulations, an added effort of verification and validation must be applied to models and simulations, to assure credibility in the simulation of complex systems.

1.1 Problem Statement and related work

Taking up complexity in state-of-the-art cyber-physical systems, it does not exclusively manifest through the technical complexity of the product itself, but also through the complexity of the product's underlying development process. More particularly, the product development in industries like the automotive industry typically has a strong distributed character, represented by complex supply chains, where simulation models are shared across organizational borders. This does not only go along with losing direct access to model sources, if models are provided as Black-Box models like Functional Mock-Up Units[1] (FMU), but also with a lack of knowledge about modeling assumptions, internal requirements, model design justification, or applied verification and validation techniques.

To keep this traceability information throughout the whole engineering process, the SET Level[2] project proposed a process framework for the execution of simulation-based engineering tasks (Heinkel and Steinkirchner, 2022) that supports for so called *credible* development of models and simulation, based on a detailed guideline focused on traceability, comprehensibility, and completeness of the documentation for modeling and simulation tasks.

However, to keep the framework generic and applicable to a wide range of engineering and simulation domains, this process framework is deliberately neither specifying the quality assurance any further nor does it define for distinguished methods to apply, dependent on the criticality of the simulation task.

The ITEA 3 project UPSIM[3] builds up its developments based on the SET Level result and smoothly extends this concept by introducing a formal quality assurance approach, targeting its integration into a collaborative, Continuous Integration (CI) environment for simulations and finally Digital Twins. In (Gall et al., 2021) the state-of-the-art and best practices in the development and management of credible Modelica models have already been identified within the UPSIM project, to be used as a basis for future improvements to work towards a well-documented, traceable development process for Modelica-based credible models.

The goal of the presented work is to introduce a concept for the continuous assessment of the credibility of simulations, using (among others) standards published by the Modelica Association and layered

[1]https://fmi-standard.org
[2]https://setlevel.de/projekt
[3]https://upsim-project.eu

standards on top of them. Furthermore, it will be shown that this concept can be applied domain-neutral and is extendable by design.

The remainder of this paper is organized as follows: First, an overview of the SET Level process framework for the realization of credible simulation tasks is given in Section 2, including its traceability concept. It is followed by a description of a credibility-based concept for quality assurance in Section 3, which will be applied to the recently proposed process framework that builds the foundation for distinguishing the applied degree of credibility. In Section 4, the generic implementation concept for credibility-based quality metrics will be sketched. These implementations will be finally used in a CI pipeline for the continuous assessment of the simulation's credibility.

2 Modeling and Simulation Process

The reliability and traceability of a decision-making process in engineering can be supported using reliable processes. If a simulation is involved in the decision-making process, an important requirement of a simulation process is to be embeddable into the overall development process frameworks. The SET Level *Credible Simulation Process* was – among other important assumptions, like taking into account the distributed character of the development and the necessity for traceability – built to fulfil this requirement. It therefore represents a lightweight and generic framework to be tailored to company specific handling of simulation.

2.1 Credible Simulation Process Framework

In (Heinkel and Steinkirchner, 2022) a complete framework is proposed to integrate a credible realization of simulation tasks into the overall product development process.

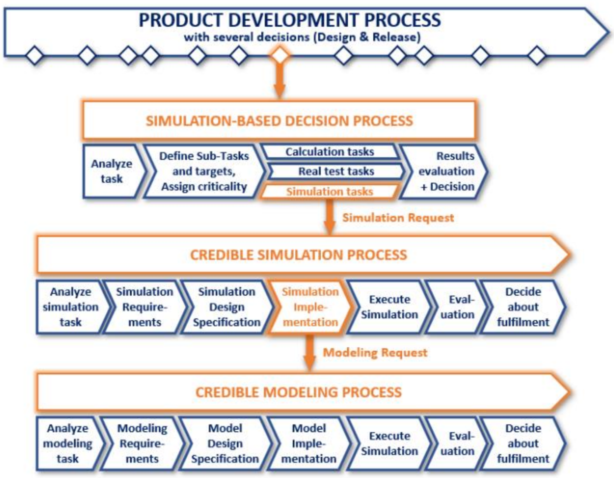

Figure 1. Credible Simulation Process Framework

Figure 1 illustrates the relationship between the product development and its underlying processes. While there are several decisions to be made during the product development process, some decisions will incorporate simulation in the decision-making process, i.e., representing simulation-informed decisions.

2.1.1 Simulation-based Decision Process

These decisions will be made within so-called S*imulation-based Decision Processes* (SbDP) and are characterized by the fact that they contain simulation tasks, but may contain other tasks which do not use simulations. Each SbDP is assigned a decision consequence that shall be used as an input for approximating the criticality of an underlying simulation task that will be governing the actions for quality assurance.

For each of the underlying simulation tasks a *Simulation Request* is submitted.

2.1.2 Credible Simulation Process

The Simulation Request can be considered as an interface between the SbDP and the Credible Simulation Process (CSP). A Simulation Request transfers information from the SbDP to the CSP. Furthermore, requirements, specifications, and even implementations, if available, can be specified in advance.

The CSP has different phases needed to be executed, where the process is illustrated in form of a linear approach, but is typically applied and executed in an inherently iterative way, where steps are repeated several times. When it comes to model implementation, a *Modeling Request* is issued for the credible development of the models to be used for simulation.

2.1.3 Credible Modeling Process

Equivalent to a Simulation Request, a Modeling Request represents the interface from the CSP to the *Credible Modeling Process* (CMP). A Modeling Request contains all necessary information from the simulation process that is required to create a model for the dedicated simulation task, where distinctive requirements, specifications, and even implementations can be specified in advance.

The CMP will be processed equivalently to the CSP and is to be considered as an iterative process, as well.

2.2 Traceability

To allow for collecting relevant information to reconstruct how simulation results have been generated by execution of the CSP and CMP, this relevant information must be made available by means of metadata. For this purpose, a metadata specification – the *SSP Traceability Specification*[4] – has been drafted within the SET Level project as a layered standard in the

[4]https://pmsfit.github.io/SSPTraceability

DOI 10.3384/ecp193171

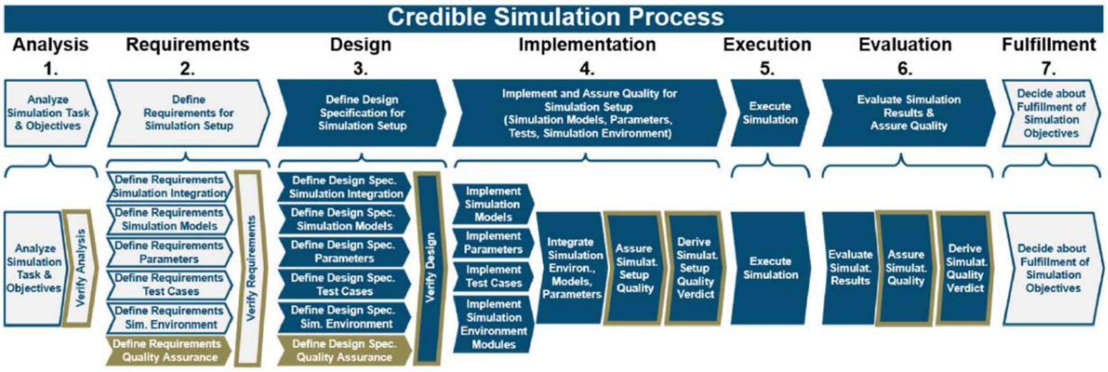

Figure 2. Phases and steps of the CSP (Heinkel and Steinkirchner, 2022)

SSP standard (Modelica Association, 2019) that can be used alongside the CSP and CMP. This specification defines the *Simulation Task Meta Data* (STMD) in form of an XML schema to store relevant meta data of process steps throughout the CSP and CMP. The specification provides for adding meta data regarding the processing scheme of each step in the CSP and CMP, namely for inputs, procedure, outputs, rationales, etc. Moreover, there is the possibility to add lifecycle information and linkage between certain resources of steps.

As the STMD will be used for the arrangement of the Continuous Integration pipeline in Section 4, the detailed application of this specification will be given within this further section.

3 Credibility Assessment

In order to be able to give an approximation about the simulation's credibility, it must be determined first how *credibility* can be specified and how to distinguish it from ordinary quality definitions. Further, a procedure must be derived on how to rank different degrees of credibility.

3.1 Distinguishing Credibility from Quality

For the term quality, there are existing many definitions that are widely accepted. From a generic point of view, *quality* can be defined as "the extent to which something has features which are good or bad, etc, especially features which are good" (Cambridge, 2022). From a technical point of view, quality is widely accepted to have two meanings (Vivek, 2005):

1. A characteristic of a product or service that bears on its ability to satisfy stated or implied needs.
2. A product or service free of deficiencies.

Following the above definitions, quality in simulation manifests itself through meeting its specified and unspecified requirements and being free from defects. To identify these target states, quality metrics and criterions are required. According to (Schütt, 2022), quality *metrics* are used to calculate metric results, based on data generated during test case execution, whereas a quality *criterion* is used to evaluate a metric result in relation to a threshold or evaluation scale.

The term *credibility* is interpreted more broadly, especially in the simulation domain. (Beisbart, 2019) notes that credibility may appear as something subjective since it can be reduced to being a property of a claim which deserves belief. He argues that however, the worthiness of belief is at least arguable that the degree to which a claim is credible in a certain context can be determined in an objective way.

Beisbart sharpens the term credibility by setting it in relation to the terms *truth* and *accuracy*: What users of simulation are interested in is simply the truth or, at least, that the outputs from their simulation come closest to the true values of the characteristics of interest. Nevertheless, the credibility of claims can only be established realistically based on the accuracy of the outputs. Therefore, credibility should be a function of the available evidence of a claim, in other words: The stronger the evidence of a claim, the more credible the claim.

(Oberkampf, 2019) supports the relation to truth and accuracy, as he states that simulation credibility deals with the assessment of the accuracy of certain system response quantities (SRQ) with respect to some true value or referent. He identifies three key issues on how to make credibility measurable:

1. How are the SRQ compared to the true values?
2. What is regarded as the true value?
3. What is the requirement for the simulation to be considered credible by the user or customer?

Whereas he carries out further that the first two issues are closely related to verification, validation, and uncertainty quantification, the third issue is rarely addressed in most simulation communities. He concludes that the requirement must be judged in relation to the accuracy of the simulation compared to the true value, even if the true value is also unknown or uncertain. He stresses further that the adequacy requirement should be set by the customer of the simulation.

(Gelfert, 2019) adds to this view that the assessment of model credibility needs always to be tentative and context-dependent – even for the rare case that a model may turn out to be successful and credible across a wide range of questions and applications.

3.2 Credibility Assessment concept

Based on the above statements, some basic assumptions for a credibility assessment can be formulated:

1. **Adequacy for purpose:** Credibility can only be formulated for a specific purpose of a simulation or a model and is never universally valid for a simulation or a model.
2. **Customer demand:** The required degree of credibility must be formulated in advance by the (external or internal) customer.
3. **Holistic approach:** Verification and Validation are crucial parts of a credibility assessment, but credibility should not be reduced to it, as a weak definition of requirements for example may be critical, if reference data is not available or limited.
4. **Collect Evidence:** To support credibility, evidence about the statement that is planned to be expressed with simulation must be collected. This evidence can be articulated with quality metrics and criterions.

We will follow a holistic approach, which means that for each relevant step of a process phase of the CSP or CMP (cf. Figure 2) evidence in form of evaluating quality metrics with quality criterions will be collected for assessing the credibility of the given objective of a simulation or model. The quality metrics will give supporting evidence about:

- How well founded and justified each development action is, for phases that will be carried out on the left side of the V-Model (VDI, 2021), namely the requirement definition and design specification phase; and
- how thoroughly the development actions are verified and validated (right side of the V-Model), namely for the implementation/integration and evaluation phase.

Another factor to be considered within the credibility assessment concept of this work is based on an insight from (Murray, 2015), gathered from the evaluation of several simulation case studies: For assessing the credibility of physically-based simulation models, a comprehensive view with respect to testing and validation procedures must be taken, as it is not enough to apply only few tests and validation methods, which leads to another principle of our concept, to distinguish the degree of credibility, based on:

- The collected amount of evidence; and
- the degree of formalization of the evidence.

This results in a discrete scale for the credibility assessment, consisting of three[5] *credibility levels* (CLs), where the lowest level provides for applying informal methods, usually based on expert opinion, whereas the highest level provides for applying metrics based on formal methods. The discrete scale is organized in a cumulative fashion: To reach the higher credibility level, the next lower credibility level needs to be accomplished before. This approach supports that the amount of evidence and the heterogeneity of applied methods rises with increasing credibility level.

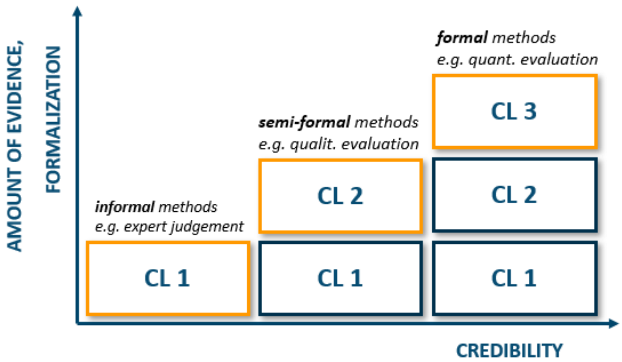

Figure 3. Discrete Credibility Level concept

The required credibility level for a specific simulation task will be determined by the customer in advance using a *Criticality Indicator*. This indicator is calculated in the course of a *Criticality Assessment* by evaluating:

1. The possible consequences in case of a wrong decision during the product development process;
2. the probability that a failure event happens at least once during the product lifetime that would lead to the described consequence; and
3. the influence of the simulation task on the decision of the associated engineering task.

This procedure is closely related to the M&S Criticality Assessment of the NASA Standard for Models and Simulations (NASA, 2016) and the criticality analysis of the Failure Mode and Effects Analysis (IEC, 2006).

For the formulation and application of quality metrics, many different quality metrics may exist that could be considered, depending on the applied simulation and model type and on the engineering domain. For this reason, the systematics are extended by a formulation concept for quality metrics (see Figure 4). On the first dimension, the applicability of the metrics will be differentiated. While there are metrics that can be applied to a wide range of simulation types (e.g., quality metrics that will give evidence about model

[5]The amount of three levels has been chosen in accordance with a process assessment that evaluates the degree to which a company has incorporated the CSP, similar to an A-SPICE assessment

DOI
10.3384/ecp193171

convergency), others will be very specific and will remain subject to a certain engineering domain.

On another dimension, we will distinguish between abstract and concrete quality metrics. Abstract quality metrics will represent an implementation guideline and will be valid for a specific phase within the CSP.

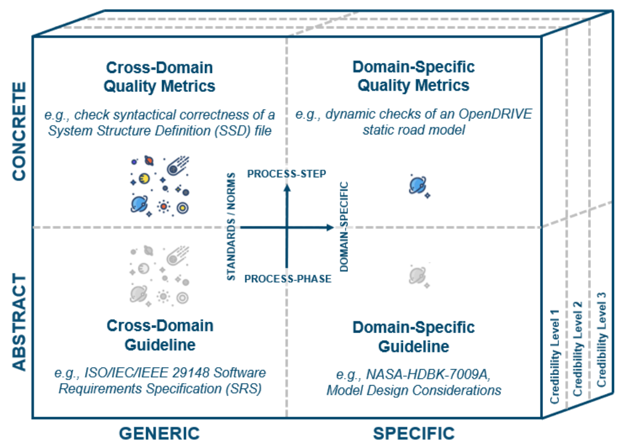

Figure 4. Systematics for formulating Quality Metrics to assess simulation credibility

For example, an abstract metric for the specification of requirements, like the ISO 29148 standard (ISO, 2011) will be valid for the complete requirements phase of the CSP, whereas for the specific steps within the requirements phase the concrete metrics may adapt specifics with respect to the formulation of model requirements or test case requirements.

An abstract metric from a technical point of view can never be used for direct evaluation, as comparatively an abstract class in object-oriented programming can never be instantiated.

3.3 Abstract, Generic Quality Metrics

In the following, we will give a short description of abstract generic quality metrics for the phases that require quality assurance in the CSP (see Figure 2), following the concept given in Subsection 3.2. All following abstract metrics are equivalently representative for the CMP in the same way as for the CSP.

3.3.1 Requirements Phase

During the requirements phase of the CSP (Define requirements for simulation setup, see Figure 2) the requirements of the simulation task are broken down into the individual requirements for the simulation integration, models, parameters, test cases, and simulation environment. Essential within this phase is the clarification of general conditions, relevant assumptions, and requirements that the simulation must fulfil.

For the credibility of the simulation, it is important that requirements are formulated clearly and unambiguously in order to narrow down the interpretational space. Moreover, requirements shall be well founded to allow for traceability and should ideally be communicated using a standardized format to mitigate losing information due to incompatibility of requirement management tools.

This results in the following guideline for the different credibility levels, mainly derived from ISO/IEC/IEEE 29148 (ISO, 2011) and the INCOSE Systems Engineering Handbook (Walden, 2015):

1. **Semantic check**: All single requirements must be formulated according to semantic [6] criteria: A requirement must be necessary, unambiguous, complete, singular, achievable, and verifiable. Further, the collection of all requirements must be complete, consistent, affordable, and bounded.
2. **Check of traceability attributes**: All single requirements must contain traceability information to their source of the task analysis, to parent requirements (if child requirement), to peer requirements and to verification/validation results.
3. **Formal check**: Requirements must be provided using a standardized implementation like ReqIf (OMG, 2016) and must contain an agreed set of attributes.

3.3.2 Design Phase

In this phase of the CSP (Define design specification for simulation setup), consistent, coordinated specifications for all artifacts, models, tools, and parameters are elaborated.

The documentation of justifications for the selection of a specific design is essential for the credibility of the outcomes of this phase. This can be done on different levels of abstraction and detail, which should be aligned beforehand between customer and supplier.

The following guideline must be implemented for the credibility assessment of the design phase for the following credibility levels:

1. **Basic justification checks:** Basic justification of design specifications (e.g., the decision for a specific approach when modeling an effect, the source of parameter values, why specific test cases are used, etc.) must be documented, to check if the simulation has been built according to its given purpose and if the requirements have been respected. Must contain design assumptions and constraints, where necessary.
2. **Traceability check:** Check if the design specifications are supported formally, using linkage to other process phases. Especially, a decision must be justified with requirements and

[6] A detailed description of how to interpret the criteria can be found in the mentioned references

results of the task analysis. Moreover, links to verification results must be given to support proof of evidence. The traceability check may use results from meta-models of the simulation task, like a Goal Structuring Notation (GSN) (Spriggs, 2012).

3. **Constraints and Assumptions check:** Check if design constraints and assumptions are supported using linkage within the process phase to specifications of other steps (e.g., between test cases and parameters or models and environment, etc.). The traceability check may use results from meta-models of the simulation task, like a GSN.

3.3.3 Implementation Phase

In the implementation phase, the different elements of the simulation setup (models, parameters, test cases, simulation environment) will be implemented and integrated according to the information from the design specification phase. The verification of the functionality of the elements individually and in their interaction in the simulation setup will be carried out within this phase.

Verification is one of the most discussed topics in the simulation community. However, a comparable methodology on how to approach verification in modeling and simulation can be observed: Conducting code verification first, embracing software quality assurance and numerical algorithm verification, followed by solution verification, focusing on the estimation of the numerical accuracy of discrete solutions compared to their mathematical model; cf. (Roy 2005; Rider 2019).

In this phase, the transition from collecting evidence by means of foundation and justification to collecting evidence by thorough verification and validation is made. Therefore, the abstract metrics will focus on verification:

1. **Informal verification:** Basic code verification, beginning with Software Quality Assurance focusing on reliability and robustness from the perspective of software engineering, as for example described in (IEEE, 2014). Must be followed by static code checks and basic dynamic code checks.
2. **Formal, qualitative verification:** Verification must be carried out according to formal methods, and results will be evaluated according to qualitative acceptance criteria.
3. **Formal, quantitative verification:** Verification must be carried out according to formal methods and results will be evaluated according to quantitative acceptance criteria, using benchmarks as quality criterions that have been agreed on between customer and supplier.

3.3.4 Evaluation Phase

The final process phase of the CSP that requires for collecting evidence is the evaluation phase (Evaluate simulation results & assure quality). In this phase, simulation results are processed and evaluated. On the one hand, the simulation results are evaluated to make an assertion about the question the simulation task is trying to answer (e.g., “is the torque of the electrical motor sufficient to start the combustion engine?") and on the other hand, a confidence range of the given assertion must be approximated.

We propose the credibility-level-guideline for validation and uncertainty quantification as following:

1. **Informal validation:** Using informal validation techniques, as described in the taxonomy of (Balci, 1997), to assess if the simulation is a sufficient representation of the system. Evidence about the confidence of the given assertion must be given by approximating and propagating the worst-case configuration (in terms of uncertainties).
2. **Formal validation:** Formal techniques must be carried out for quantitative assessment of the validity of the simulation, using a validation benchmark as quality criterion that has been predefined and agreed upon. The validation domain must be predefined by a subject-matter expert (SME) to identify critical validation points. Evidence about the confidence of the assertion must be given by propagation of the upper and lower boundaries of the uncertainty range, that have been approximated by an SME.
3. **Uncertainty quantification:** The assertion must be supported by performing an uncertainty quantification, following the guideline proposed in (Roy and Oberkampf, 2010) for specific inputs that have been agreed upon.

These guidelines can be implemented specifically, on the one hand by different domains (e.g., the automotive, aerospace, or medical domain) and from another perspective with respect to different model types (i.e., for continuous models: Surrogate models, models based on algebraic equations, models based on ordinary differential equations, models based on partial differential equations, etc.).

Furthermore, the guidelines can also be implemented for system simulations, which will result in implementations that are widely based on evaluations using simulation and modeling standards.

3.4 Examples for Concrete, Generic Quality Metrics using Modelica Standards

The usage of standards can help to ease the implementation of the above proposed guidelines. In the following, some basic examples are provided on how to use standards of the Modelica Association to collect evidence. The following should be understood as examples on how to implement the abstract guidelines and do not have any claim to completeness of

DOI
10.3384/ecp193171

implementing all possible quality metrics, based on standards of the Modelica Association.

The System Structure Definition (SSD) is defined as part of the SSP specification and describes a nested hierarchy of interconnected (sub-)systems and atomic components (Modelica, 2019), which can be used for the implementation of system models.

To verify the implementation of the system structure, based on an SSD file, the following steps need to be carried out that can be considered as very basic CL1 quality metrics of the implementation phase for models (static code checks):

- **Syntax check:** Check if the SSD file implements the corresponding XSD [7], defined in the SSP specification.
- **Logic check:** Even if an SSD file implements the XSD correctly, it is not ensured that the proposed structure can be implemented. Therefore, logical checks need to be done: Check if all inputs are connected (if so required); check if connectors specified in the connections exist; check if the data types of wired connectors are consistent; check if connections are kept within their relevant subsystem.

To perform analogous static code checks for FMU model descriptions, the procedure for CL1 quality checks is similar:

- **Syntax check:** Check if the model description of the FMU implements the corresponding FMI (Functional Mock-up Interface) description schema.
- **Logic check:** Besides checking for a valid implementation of the XSD, the FMI specification defines some requirements and boundaries for the implementation (e.g., for definition of units or the allowed combination of attributes for specific variables). Therefore, some basic logic checks will be performed: Check if all units, used in the variable definitions are well defined with SI units; check if all types, used in the variable definitions are well defined; check if attribute combinations for variable definitions are valid.

When it comes to integration, it must be further ensured that the system structure and the underlying elements are compliant. Therefore, a basic integration check, based on static code analysis can be carried out.

- **Integration check:** Check if connectors, defined in the SSD, are consistent with variables/ports of the underlying component implementation (in the case of referenced FMUs this must match the name of the relevant variable in the referenced FMU).

The above-described quality metrics are basic examples of how to use implementation checks of Modelica Association standards within a credibility assessment, even if these checks can be considered state-of-the-art of many tools that implement these standards. For further examples, implementations and applications of concrete quality metrics, we refer to the repository of the so-called *Credibility Development Kit* that is outlined in Section 4.

4 Implementation Concept

The implementation of the concept for a credibility assessment proposed in Section 3 will intentionally be kept agnostic towards specific software applications and systems to enable broad usability. Within the UPSIM project, we are initiating implementations of the proposed concept that will result in a software development kit that provides quality metrics for each credibility level and each process phase of the CSP with the goal to provide reusable quality metrics for a credibility assessment in a transparent manner. It will be denoted as *Credibility Development Kit*[8] (CDK) in the following.

4.1 Credibility Development Kit

The core component of the CDK is a collection of Concrete Quality Metrics, mapped to process phases/steps and credibility levels that can be used to collect evidence for a credible statement of a simulation.

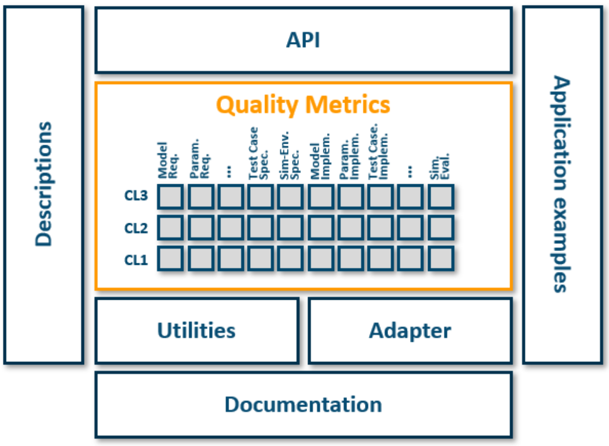

Figure 5. Components of the CDK

To support for the correct and unambiguous usage of Quality Metrics, further components are part of the CDK:

- **Descriptions, Documentation:** Descriptions of what Quality Metrics aim to measure and additional code documentation, using JSDoc[9].

[7]XML Schema Definition, https://www.w3.org/TR/xmlschema11-1

[8]https://github.com/virtual-vehicle/Credibility-Assessment-Framework/tree/main/Credibility-Development-Kit

[9]https://jsdoc.app

- **Utilities:** A collection of reusable helper functions that are used across different Quality Metrics with the purpose to have reproducible, traceable procedures, e.g., on how data is pre/post-processed.
- **Adapters:** A collection of functions that transform individual input data structures (may be standardized data structures like SSD or proprietary formats) into the data structures expected by the Quality Metrics implementations as an input (see Subsection 4.2).
- **API:** Facades as entry points to control pre-defined workflows, automatically using the correct adapters for individual Quality Metrics (cf. Figure 6).
- **Application examples:** Collection of best practices of proposed and former usage of Quality Metrics.

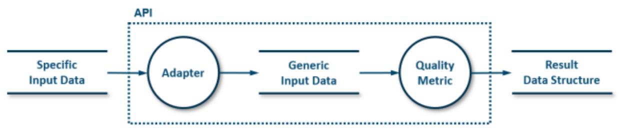

Figure 6. Basic data flow using Quality Metrics for credibility assessment. Notation: (DeMarco, 1979)

4.2 High-Level-Design

Taking into account the general considerations to enable broad applicability of Quality Metrics for a credibility assessment, the implementation of the CDK is carried out as a collection of Node.js[10] packages.

To support the applicability, especially to avoid insisting on too specific file formats, further considerations have been taken into account: A network-friendly data interchange format like JSON[11] is used and generic input data structures are defined that will be used as input to Quality Metrics. To allow for usage of different (standardized and proprietary) file formats, adapters can be provided that translate specific input data into the expected input data structure.

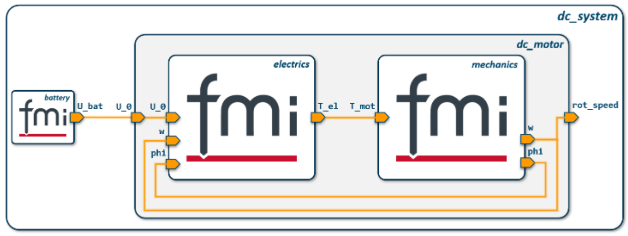

Figure 7. Simple example of a system model

As an example, for the simple system model in Figure 7, both the system and model connections are represented by the SSD standard and the proprietary format of the tool Model.CONNECT[12] will result in the same generic data structure (see Figure 8) that can be used as an input for a Quality Metric, once serialized.

As stressed in Subsection 3.2, some tests may require expert judgements. In this case the quality criterion will not directly be evaluated in the software function, but instead by an expert beforehand. To integrate this implicit evaluation, the statement of an expert (given for example as another serialized JSON structure) must be digitally signed by the expert, indicating the hash algorithm and signature encoding used (see Listing 1). This way the customer can verify if the judgement originates from an authorized expert, by checking against the individual public keys from expert's certificates that have been agreed upon before being accepted to carry out expert judgements.

Figure 8. Adapter application example for a standard and proprietary format

[10] https://nodejs.org/en/about

[11] https://www.ecma-international.org/publications-and-standards/standards/ecma-404

[12] https://www.avl.com/-/model-connect-

DOI
10.3384/ecp193171

Equivalently, an adapter is provided for signing expert judgements and transforming the judgement to the expected structure, as presented in Listing 1.

```
{
  "expert_judgement": "The func is fine",
  "signature": "9f066d7654fnk8hgc59f…",
  "hash_algorithm": "SHA256",
  "signature_encoding": "hex"
}
```

Listing 1. Example for a digitally signed expert judgement (shortened for better readability)

Similar to the unification of the input data that is passed to Quality Metric functions, outputs of Quality Metrics are (serialized) JSON structures and will always keep the following specific schema: The result, that indicates if the criterion of the Quality Metric has been matched or not, as well as logging information that adds valuable information to be used as feedback.

The outputs can be used for continuous assessment of the simulation's credibility (see Subsection 4.3).

4.3 Continuous Credibility Assessment

In the following, a proposal for the application of the outlined concept of this work for the continuous assessment of a system simulation will be sketched. In this application example, a multi-supplier scenario is presented that is using a Continuous Integration/ Deployment (CI/CD) pipeline for a continuous credibility assessment.

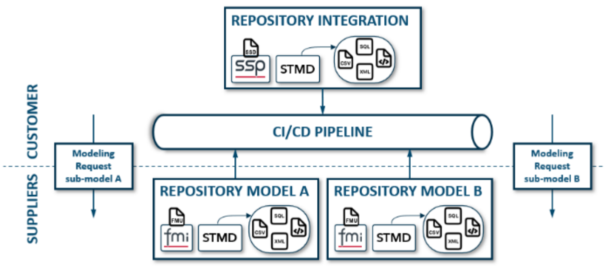

Figure 9. Example scenario

In the example application, we consider a distributed development of a system simulation, where a customer will integrate models from different parties – these parties will typically be suppliers that will provide black-box models (to keep the complexity of our example low, we consider only two sub-models).

The customer must have performed a criticality assessment (cf. Subsection 3.2) that will provide the required credibility level for the simulation task the customer plans to execute. In agreement with each supplier, the customer will select Quality Metrics (and quality criterions) from the CDK, according to the required credibility level that the corresponding sub-model needs to fulfill. This way, the customer and the supplier will have a bilateral, unified agreement on the interpretation of the credibility of this specific sub-model.

4.3.1 STMD for unique Artifact Identification

During the development of the sub-models, the suppliers must – in addition to the bare provision of the model – provide additional credibility documentation that can be used as inputs for the selected Quality Metrics of the CDK, especially for those process phases where data cannot be produced directly within the continuous integration pipeline (like for example simulation results, generated for verification and validation). To ensure the correct mapping of these credibility documentation artifacts to corresponding Quality Metrics, we propose to provide an STMD file (see Section 2) for unique identification.

For each step of the Credible Simulation and Credible Modeling Process the STMD schema enables for providing credibility documentation via the *Rationale* element. We propose to add the sources of all additional artifacts required for the credibility assessment within a Rationale element as *Resource* element and specify the mapping as a *MetaData* element of the Resource. The following elements and arguments shall be used with the subsequent conventions:

- **Resource element:** The argument *kind* must be specified as "credibility-documentation"; the argument *type* must indicate the MIME-Type of the resource, as required in the STMD specification; the argument *source* must indicate the URI [13] of the credibility documentation resource
- **Metadata element:** The argument *kind* must be specified as "metric-mapping"; the argument *type* must be specified as "text/xml"
- **Content element:** This element is allowed to contain user-defined elements. We propose to use an element called *cdk:Task*
- **cdk:Task element**: The argument *level* must be provided to indicate the corresponding credibility level
- **cdk:ValidationFunction element:** The argument *function* must be provided to indicate the target Quality Evaluation function. The argument *adapter* must be specified if the resource must be transformed to the expected file format and data structure; it must indicate the name of the function of the adapter to use

Figure 10 is presenting an excerpt of the proposed unique resource identification for a Quality Metric of the Design Specification phase that requires the provision of an expert judgement for CL1 and a graph for CL2.

4.3.2 Continuous Assessment and Deployment

In this manner, the credibility assessment must be performed for each phase and step of the process. The

[13] https://www.w3.org/TR/uri-clarification

```xml
<stmd:SimulationTaskMetaData name="CMP-Submodel-A" GUID="6d9b7dde-57fe-4e68-b016-3eaba05f07db" ...>
    ...
    <stmd:DesignPhase>
        <stmd:DefineModelDesignSpecification>
            <stc:Rationale>
                <stc:Resource kind="credibility-documentation" type="application/json" source="./design_spec/model_design_justification.json">
                    <stc:MetaData kind="metric-mapping" type="text/xml">
                        <stc:Content>
                            <cdk:Task level="1">
                                <cdk:ValidationFunction function="checkModelDesignJustification"/>
                            </cdk:Task>
                        </stc:Content>
                    </stc:MetaData>
                </stc:Resource>
                <stc:Resource kind="credibility-documentation" type="application/rdf+xml" source="./design_spec/model_design_graph.rdf">
                    <stc:MetaData kind="metric-mapping" type="text/xml">
                        <stc:Content>
                            <cdk:Task level="2">
                                <cdk:ValidationFunction function="checkModelDesignTraces" adapter="rdfxmlToJson"/>
                            </cdk:Task>
                        </stc:Content>
                    </stc:MetaData>
                </stc:Resource>
                <stc:Resource>
                    ...
                </stc:Resource>
            </stc:Rationale>
            ...
        </stmd:DefineModelDesignSpecification>
        ...
    </stmd:DesignPhase>
    ...
</stmd:SimulationTaskMetaData>
```

Figure 10. Example of using the Rationale element within a STMD file to ensure unique identification of Quality Metric inputs

concrete Quality Metrics that have been agreed upon between customer and suppliers will be evaluated to aggregate the results for being able to derive a condensed credibility level for the individual sub-models.

By providing the logging information, next to the results of the Quality Metric evaluation (see Subsection 4.2), developers are able to get feedback and can iterate on developing towards the required credibility level. For each iteration the credibility level will be determined, based on the changes done; thereby the credibility is assessed continuously.

It is important to point out that the interpretation of the overall credibility of a simulation or a model depending on the achieved atomic credibility levels of each process step is the responsibility of the applying parties. As we see the risk of error propagation in a simulation process, we propose to use a minimum rule, which means that the overall credibility level is equal to the lowest credibility level of a single process step. This does not apply only for the sub-models by execution of the CMP on the supplier side, but for the overall simulation by execution of the CSP on the customer side, as well.

The final step of the CI/CD pipeline is the automatic deployment of an SSP package. This package is having a unique identifier with the purpose to enable an unambiguous mapping of the deployed SSP package to the assessed credibility level. Again, it must be emphasized that the credibility level connected to this package is only valid for the given purpose of the simulation and does not represent a globally valid certification for the simulation model.

5 Conclusions and Outlook

In this paper, a proposal is presented on how to continuously assess the credibility of models and simulations. As the term *credibility* is interpreted differently in the community, we clarified some basic assumptions that our concept for credibility assessment is built on.

A central aspect of these assumptions is that credibility increases with the amount of evidence given about the statement that is planned to be expressed with simulation. However, even if we recommend distinguishing the degree of credibility by using a discrete level scale that is separated according to the formal degree of the applied methods and aims at increasing the amount of collected evidence with increasing credibility level, it must be stressed that this classification can only represent an approximation of the credibility. Simulation tasks will differ in their modeling approach, complexity, and prior knowledge. Therefore, we emphasize taking these conditions into account when applying the presented concept, especially when it comes to selection of Quality Metrics.

From implementational point of view we presented an approach on how to document and reference to additional information that will be required for assessing the credibility. In our concept, we adapted to the concept of STMD that references additional traceability and credibility documentation by adding it as a layered standard on top of the specification of Modelica standards. Still, there's a trend to add credibility information directly to models (cf. Gall et al., 2021), which is also subject of investigation within the UPSIM project. These developments can be considered to be used in further developments of the continuous credibility assessment in this work.

Acknowledgements

This work has been partly supported by the European ITEA3 Call6 project UPSIM[14] – Unleash Potentials in Simulation (number 19006). The work was funded by the German Federal Ministry of Education and Research (BMBF, grant numbers 01IS20072H and 01IS20072G).

References

Osman Balci (1997). Verification, Validation and Accreditation of Simulation Models. In: *Proceedings of the 1997 Winter Simulation Conference*. Blacksburg, Virginia, USA. DOI: 10.1145/268437.268462

Claus Beisbart (2019). What is Validation of Computer Simulations? Toward a Clarification of the Concept of Validation and of Related Notions. In: *Beisbart, C., Saam, N. (eds) Computer Simulation Validation. Simulation Foundations, Methods and Applications*. Springer. DOI: 10.1007/978-3-319-70766-2_2

Cambridge University Press (2022). *Cambridge Dictionary*. Accessed July 20, 2022 [Online]. URL: https://dictionary.cambridge.org

Tom DeMarco (1979). Structured Analysis and System Specification. Yourdon Press, New York. P. 352. ISBN: 978-0138543808

FDA (2021). Assessing the Credibility of Computational Modeling and Simulation in Medical Device Submissions, Draft Guidance for Industry and Food and Drug Administration Staff. U.S. Department of Health and Human Services Food and Drug Administration

Leo Gall et al. (2021). Continuous Development and Management of Credible Modelica Models. In: *Proceedings of the 14th Modelica Conference*. Linköping, Sweden, pp. 359–374. DOI: 10.3384/ecp21181359

Axel Gelfert (2019). Assessing the Credibility of Conceptual Models. In: *Beisbart, C., Saam, N. (eds) Computer Simulation Validation. Simulation Foundations, Methods and Applications*. Springer. DOI: 10.1007/978-3-319-70766-2_10

Hans-Martin Heinkel and Kim Steinkirchner (2022). Credible Simulation Process Framework. Tech. rep. Robert Bosch GmbH and PROSTEP AG. URL: https://gitlab.setlevel.de/open/processes_and_traceability/credible_simulation_process_framework

IEC (2006). Analysis techniques for system reliability - Procedure for failure mode and effects analysis (FMEA), Second Edition. International Electrotechnical Commission. International Standard IEC 60812:2006.

IEEE (2014). Software Quality Assurance Processes. Institute of Electrical and Electronics Engineers. International Standard IEEE 730. DOI: 10.1109/IEEESTD.1998.88284

ISO/IEC/IEEE (2011). Systems and software engineering - Requirements engineering. Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission/Institute of Electrical and Electronics Engineers (IEEE), (IEC). ISO/IEC/IEEE 29148

Alessia Knauss et al. (2017). Paving the roadway for safety of automated vehicles: An empirical study on testing challenges. In: *Proc. IEEE Intelligent Vehicles Symposium (IV)*. Los Angeles, USA. pp. 1873–1880. DOI: 10.48550/arXiv.1708.06988

Philip Koopman and Michael Wagner (2017). Autonomous vehicle safety: An interdisciplinary challenge. In: *IEEE Intelligent Transportation Systems Magazine*, Volume 9, No. 1, pp. 90–96. DOI: 10.1109/MITS.2016.2583491

Modelica Association (2019). System Structure and Parametrization. Report 1.0. URL: https://ssp-standard.org/publications/SSP10/SystemStructureAndParameterization10.pdf

David Murray-Smith (2015). Testing and Validation of Computer Simulation Models. In: *Simulation Foundations, Methods and Applications*. Springer. pp. 233-343. DOI: 10.1007/978-3-319-15099-4_13

Vivek Nanda (2005). Quality Management System Handbook for Product Development Companies (1st ed.). CRC Press, pp. 1-3. DOI: 10.1201/9781420025309

NASA (2016). Standard for Models and Simulations. NASA-STD-7009A, Rev. A, Change 1. pp. 50-54. National Aeronautics and Space Administration. URL: https://standards.nasa.gov/sites/default/files/standards/NASA/w/CHANGE-1/1/nasa_std_7009a_change_1.pdf

William L. Oberkampf (2019). Simulation Accuracy, Uncertainty and Predictive Capability: A Physical Sciences Perspective. In: *Beisbart, C., Saam, N. (eds) Computer Simulation Validation. Simulation Foundations, Methods and Applications*. Springer. DOI: 10.1007/978-3-319-70766-2_3

Object Management Group (OMG) (2016). Requirements Interchange Format (ReqIF). Standard Specification 1.2. URL: http://www.omg.org/spec/ReqIF/1.2

William J. Rider (2019). The Foundations of Verification in Modeling and Simulation. In: *Simulation Foundations, Methods and Applications*. Springer. pp. 271-293. DOI: 10.1007/978-3-319-70766-2_11

Erica N. Rogers (2019). Risk Assessment for Medical Devices. In: *Shayne C. Gad. Integrated Safety and Risk Assessment for Medical Devices and Combination Products.* Springer. DOI: 10.1007/978-3-030-35241-7

Christopher John Roy (2005). Review of code and solution verification procedures for computational simulation. In: *Journal of Computational Physics*, Volume 205, Issue 1. pp. 131-156. DOI: 10.1016/j.jcp.2004.10.036

Christopher John Roy, William L. Oberkampf (2010). A Complete Framework for Verification, Validation, and Uncertainty Quantification in Scientific Computing (Invited). In: 4*8th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition.* Orlando, USA. DOI: 10.2514/6.2010-124

Barbara Schütt et al. (2022). A Taxonomy for Quality in Simulation-based Development and Testing of Automated

[14]https://itea3.org/project/upsim.html

Driving Systems. In: *IEEE Access 10*, pp. 18631-18644 DOI: 10.1109/access.2022.3149542

John Spriggs (2012). GSN – The Goal Structuring Notation: A Structured Approach to Presenting Arguments. Springer London. DOI: 10.1007/978-1-4471-2312-5

VDI (ed) (2021). Development of mechatronic and cyber-physical systems. Technical guideline. Verein Deutscher Ingenieure. VDI/VDE 2206

David Walden, Garry Roedler and Kevin Forsberg (2015). Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities. 4. Hoboken, NJ: Wiley. ISBN: 978-1-118-99940-0