Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 00 (2025) 000–000

www.elsevier.com/locate/procedia

19th CIRP Conference on Intelligent Computation in Manufacturing Engineering

Analysis and Review of Use Case Representations in OPC UA Companion Specifications

Tonja Heinemann[a]*, Andreas Wortmann[a], Oliver Riedel[a], Armin Lechler[a]

[a]*University of Stuttgart, ISW, Seidenstraße 36, 70174 Stuttgart, Germany*

* Corresponding author. Tel.: +49711 685-84626. *E-mail address:* tonja.heinemann@isw.uni-stuttgart.de

Abstract

For industrial data management, OPC UA data models are increasing in popularity. Domain models (e.g., Robotics, Mining) are developed by groups of industry representatives and published as so-called Companion Specifications. These Companion Specifications mention use cases that are to be addressed, however with different approaches. In this work, these use case descriptions are analyzed regarding their benefit to users of the specification. The textual representations and the graphics used are compared and the use cases are categorized by expressiveness. A categorization of use case contents is performed. The findings are discussed and suggestions for use case representations are derived from the results. These suggestions serve as a tool to editors of the companion specifications and aim to improve understandability.

Peer-review under responsibility of the scientific committee of the 19th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 16-18 July, Gulf of Naples, Italy

Keywords: OPC UA, Standardization, Use Cases, Companion Specifications

1. Background

OPC UA Companion Specifications (CS) are developed in order to provide vendor-independent data communication and standardized data representation [1,2]. In this context, new CSs should consider existing CSs to reuse data representation where possible. Likewise, OPC UA servers, clients, publishers, and subscribers are implemented that utilize the CS models.

Conventionally, the CSs are read and have to be understood by the readers to either consider them in new models or to implement them. Assuming use cases aid understanding, this work focuses on the use case section suggested in the CS template [2]. Existing CSs are analyzed and their use case representations are discussed with reusing the specification and implementing the specification in focus. Recommendations for compiling the use case section are elicited.

2. OPC UA Brief and CS Template

The OPC UA Standard describes the combination of multiple concepts in order to enable information exchange. These span the communication protocols used, the functions and services for communication, the contents conveyed and security aspects fitting the respective concepts. [1]

Concerning the contents conveyed in communication, OPC UA defines a meta model with elements called “Nodes” to represent data and modelling rules for their usage. With these Nodes and modelling rules, domain experts can represent the data relevant in their field as an OPC UA model. [1]

The subset of these models that is published by the OPC Foundation is called Companion Specifications. For these CSs, it is required to adhere to a template document provided by the OPC Foundation [2]. Using the template format, the description of the model is two-fold: the Node information

2212-8271
Peer-review under responsibility of the scientific committee of the 19th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 16-18 July, Gulf of Naples, Italy

according to the meta model on the one hand and free text descriptions for the semantics of Nodes and the intended behavior of values over time, methods and the like on the other. The template suggests an overall structure that first introduces OPC UA and the conventions used in the document, then introduces the model's topic and overall design as an overview, describes all the model elements in detail, defines OPC UA Profiles for the model and in the end contains formal information concerning the namespace introduced by the model and the mapping information for data transport. [2]

OPC UA Profiles provide groupings of functionality on any level of the OPC UA concepts [1]. For a CS, this usually means topical groups of Nodes, so information as represented in the model defined by the CS.

Among the CS's topic overview, the template suggests to editors to include a section describing use cases. The prompt given reads "Insert the use cases that can be achieved by using OPC UA with the companion organization's information model". [2]

In connection with the other sections indicating the model's context and aim, the use case section may aid in working with the specification. When creating a new model and incorporating ideas and Nodes from existing models or when implementing an OPC UA communication partner using the model defined in the CS, a study of specifications takes place. In both cases, the description of the use cases can assist in the given task.

With this background, along with the brief template prompt for the use case section, this work explores how the section is used in published CSs and identifies good practices for providing benefits to readers of the CSs.

3. Method

CSs are reviewed for their use case section. The sample of CSs is taken from all specification documents published by the OPC Foundation that are not part of the OPC 10000 series and provide a use case section. The sample of specifications has been taken in December 2024, updates since then are not included. As the section structure of the template is no hard rule and as the recommendation for a use case section was not always included in the history of the template, not all documents provide this section. In total, 51 CSs with a use case section are found [3-53]. Two generalizations have been made. The parts of the plastics and rubber specifications are considered based on their main number [28-31] as the use case descriptions in the parts are nearly identical. For the mining series of specifications, the general part as well as the dedicated use case documents have been included, as all other documents reference the dedicated use case documents [47-51].

For the specifications, individual aspects, e.g., requirements or actors, mentioned in the use case descriptions are identified. Based on the number of different aspects mentioned, the specifications are categorized in describing the use cases in low detail (one aspect mentioned next to the use case title), medium detail (two or three aspects mentioned) or extensive detail (more than three aspects). The aspects elicited from all CS documents are combined for discussion.

For each specification, notable good practices are identified. Similarly to the aspects mentioned in the use cases, they are combined over all documents for discussion.

From the specification documents, individual use cases are determined (e.g., each point in a bulleted list, each subsection of the use case section etc.). As a measure to isolate the use cases, the most apparent section structuring present in the document was taken. For each use case gained this way, its title is added to a collection of use cases. The use case titles are then categorized in a qualitative content analysis style. Based on recurring concepts in the use case description, categories are defined. Each use case, represented by its title, is assigned exactly one category. If use cases remain, suitable categories are defined until each use case has a category.

Under the assumption that the contents of the use case section in CSs aid in understanding for readers, two different viewpoints are taken for discussion of the sets of aspects, good practices and use case categories. First, a reader that wants to create a model, either a CS or a model for a specific system. This reader should use existing models, if possible, which has the benefit of increased interoperability: the same concept is portrayed the same way in different CSs. The reader creating a model may also want to get inspiration for solving similar problems. This can also benefit interoperability in the long run, as it may lead to conventions and best practices.

Second, a reader studying the specification to implement it. Regardless whether the implementation provides or consumes data, the reader needs to understand the concepts used in the specification. Also, it's relevant to consider that typically not all contents of a CS are expected to be implemented in a single system. Often, the CSs describe optional contents. OPC UA Profiles are the methodical way to indicate what options there are to use the specification [1].

4. Analysis – Results

When regarding the aspects found per specification, the results shown in Table 1 are gained. Of the 51 specifications, 9 meet the requirements for extensive detail in the use case descriptions while 19 provide medium detail and 23 mention the use cases with low detail.

Table 1. Level of Detail of use case descriptions in CSs

Level of Detail	No. of CSs	CSs
extensive	9	[9], [13], [33], [40], [47], [48], [49], [50], [51]
medium	19	[6], [11], [19], [20], [21], [22], [26], [27], [32], [34], [35], [36], [37], [38], [39], [41], [44], [45], [53]
low	23	[3], [4], [5], [7], [8], [10], [12], [14], [15], [16], [17], [18], [23], [24], [25], [28], [29], [30], [31], [42], [43], [46], [52]

The aspects used to assess the level of detail are listed in the left column of Table 4. The background of a use case links its title to the domain on hand, while the scope mentions what can and can't be achieved. For some use cases, requirements that need to be fulfilled are mentioned. In some cases, the OPC UA mapping is mentioned, the way of data transmission used out of the options in the standard. Another aspect is the description of variants, different ways the use case can be realized. For many use cases, the data intended on the interface are mentioned. Especially the use cases described in extensive detail mention preconditions and postconditions, the actors involved, a trigger for execution of the use case and the sequence of actions involving the interface when execution occurs. Specifications with low detail in the description on the other hand mention the communication partner in some cases. In contrast to the actor which is the system or individual initiating or leading the execution of the use case, the communication partner is the system using the OPC UA interface. The goal of the use case is often stated, while sometimes also the goal/intention of the overall specification is highlighted. There are specifications that explicitly state how a use case contributes to that overall specification intention.

In addition to these aspects, from the notes of good practices there are four mentions that provide additional context to the descriptions, shown in Table 2. They consist of the use of a structured representation of all use cases in the same specification, the use of figures if applicable to support the contents, having extensive documentation about the use cases in an annex or even separate documents and having links to the specification contents. As a special case of the connection between the specification's contents and the use cases, some specifications also link the OPC UA Profiles they define to the use cases (e.g., [37],[28]).

In the 15 specifications providing figures, use case diagrams, sequence diagrams/activity diagrams/flowcharts and figures representing a system topology or communication structure are used. Only one specification provides figures, that don't fit these recurring diagrams [13]. In three specifications, use case diagrams are provided in a notably reduced form [3, 5, 42]. Three to four use cases are briefly mentioned and the use case diagram conveys the information that all use cases are associated with a general actor and that the firs use case extends the second one, which extends the third one (which extends the fourth one). Also, in contrast to the sequence diagrams/activity diagrams/flowcharts, the use case diagrams, topology diagrams and the figures in [13] describe aspects other than the data exchange the specification defines.

Two of the specifications provide context to use cases in an annex [32, 34], while the series of CSs in the domain of mining even uses individual documents [48-51].

In 20 of the specifications, links to the specification are used to connect the use cases to the model defined in the specification. These links are either references to the sections where model elements are defined or the names of the model elements.

Table 2. Identified Good Practices

Good Practice	No. of CSs	CSs
Structured representation	14	[7], [9], [13], [20], [21], [22], [23], [24], [33], [35], [36], [37], [40], [43]
Figures provided	15	[3], [5], [10], [12], [13], [19], [26], [27], [35], [39], [42], [48], [49], [50], [51]
Extensive documentation	6	[32], [34], [48], [49], [50], [51]
Links to specification	20	[3], [7], [11], [13], [14], [15], [16], [17], [18], [20], [22], [25], [32], [34], [37], [38], [41], [43], [45], [46]

From the specifications, 289 use cases were identified and categorized. Twelve categories have been defined; they are provided in Table 3.

In this categorization, about 30% of the use cases focus on observation. When the use cases of "Maintenance", "Identification" and "Asset management", that all similarly provide data over the interface, are considered in addition to the "Observation" category, more than half of the use cases describe data to be transmitted with no further constraints concerning the OPC UA specification.

On the other hand, use cases in the categories "Operation", "Engineering" "Write values" and "Service" require more than just a reading access on data. Here, interaction between the OPC UA communication partners with writing and using values, sending files and method calls is expected. A large number of such use cases in the operation category concerns providing and controlling jobs on the devices.

The use cases in the category "Analyse/Use data" differ from the other categories as they focus on data interpretations that are out of the scope of the interface. They motivate the data transmitted over the interface with subsequent processing and resulting benefits.

The categories "Adhere to 3rd party protocol" and "Standardized interface generic" differ significantly from the other categories. In these two categories, the use cases focus less on what data to consider and rather on their placement in the data infrastructure of cyber physical production systems.

The use cases of "Adhere to 3rd party protocol", "Standardized interface generic", "External system communication" and "Identification" emphasize a focus of the CSs on being available as a manufacturer independent data interface standard for the domain. Together, they make up around 20% of all use cases identified.

Table 3. Use Case Categories

Category	Use Case Title Concerns	No. of Use Cases
Observation	Data reporting over the interface	90
Operation	Process of the device	50
Maintenance	Phase of downtime that shall be ended	38

Engineering	Phase before using the device	23
External System Communication	Communication with a target system	22
Identification	Identifying devices, nameplate information	17
Analyze/Use data	Goal or method of analysis	15
Adhere to 3rd party protocol	Mapping of a protocol other than OPC UA	11
Standardized interface generic	Emphasis on existence of a standardized interface for the domain	8
Asset management	Asset management specifically	8
Write values	Using the OPC UA write service from a client	5
Service	Keeping devices in running order	2

5. Discussion

Regarding the level of detail of the use cases in the specifications, equally many specifications provide medium or low detail, but just about a fifth of specifications provides extensive detail. This leads to the assumption that the use case section is rather considered an accessory to the specification. The use case section is in fact an addition, as the main content of the specification documents consists of the model definition and the OPC UA Profiles. In relation, the use case section provides context to the model.

Table 4 shows an assignment of the use cases to the viewpoints. Several of the aspects are relevant to both viewpoints. Background information about the use case helps to classify the application scenario of the specification. The description of the use case's scope clarifies what is possible with the data of the specification. For creating a model, that helps the comparison with the goals of the model to create and when implementing a specification, it helps to integrate the model contents to the context of the software to be implemented.

If the requirements to a use case are stated, both kinds of reader can check whether these requirements are fulfilled in their situation. For creating a model, this check can reveal differences and similarities in the scenario and provide pointers what to consider. For implementing a model, these requirements need to be met to benefit from the use case.

An OPC UA mapping is also relevant from both viewpoints. When creating a model this can make or break an inclusion – depending on whether constraints in OPC UA options are set for the model to create. For the reader implementing a specification it can shorten the time to understand the specification and reduce errors due to misunderstandings.

The description of variants aids readers that create a model in the decision to incorporate or assimilate the specification. When implementing the model, in some cases one of the options can be chosen. In other cases the implementation has to be prepared for all the variants. A description of variants in the specification provides an orientation.

Mentioning the data intended for the use case is also helpful from both viewpoints. The data provide a clear indication what both the use case and the model want to convey. This aids in comparison and understanding to a reader that wants to integrate the model into a different specification, and it is crucial to a reader that wants to implement the use case.

Three aspects seem more helpful when creating a model than when implementing it: the communication partner, the intention of the specification and the goal of the use case.

The communication partner provides information about the use case context, which helps to align to the reader's own goals. When implementing a communication partner, the other partner is either set (e.g., equipment with an OPC UA server) or irrelevant – a server will provide data to any authorized client, regardless of what the client uses the data for.

In a similar way, the overall goal of the specification and the goal of an individual use case provide context to the data and the modelling choices taken and can aid understanding. When the reader's intention is to create a new model and possibly to incorporate the model in question, this context can help to align the overall vision. When implementing, the relevance of the specification goal is somewhat diminished. The data may be used out of context as long as there are no problems due to misinterpretation.

Some other aspects seem of greater interest when implementing a specification, than when reading it as a prerequisite to creating a model. These are the preconditions and postconditions to a use case, the actors, triggers and sequence as well as the contribution of a use case to the specification's intention.

The preconditions, postconditions, trigger and sequence can likely be incorporated into the software specification. This helps to clarify the usage of the interface and an implementation that leads to a recognizable behavior for communication partners.

Similarly, if actors are explicitly mentioned, responsibilities are stated. If the system to be implemented represents one of the actors, the respective behavior is expected by the other communication partner. Otherwise, if an actor is a different system, the implementation has to take the actions of that actor into account, but not provide the actions itself.

The contribution of the use case to the specification's overall goal helps in aligning the use case with the overall goal of the software project. The description may hint to the effects provided by the use case. Even if in implementation the goal of the specification or the goal of the use case are interpreted differently, the information what the use case contributes can aid in defining its goal in the implementation project.

Using a structured approach to use cases – always mentioning the same aspects in order in descriptive paragraphs or representing use cases in tabular form leads to similar descriptions for each use case in a specification. It prevents the editors to forget to mention aspects for individual use cases. Readers can more easily compare the use cases. Keeping the same structure for all use cases was apparent in about 30% of the specifications, so it is rather common.

While about 30% of specifications use figures to convey information about use cases, some designs align better with the viewpoints of specification readers than others. Use case diagrams as used in [3], [5] or [42] show little value to either creating or implementing a model.

Information like use case or topology diagrams is beneficial as long as it provides context to the descriptions and gives a second clue to the contents to the reader.

Sequence diagrams, flowcharts and activity diagrams on the other hand usually describe the data exchange directly. As such, they are considered helpful when implementing specifications, as they illustrate the usage scenario.

If extensive descriptions of use cases are available, they have potential to enhance understandability of the specification. Keeping the use case section compact and moving the detail to an annex may be appropriate.

Links to the specification make orientation in the document easier for a reader. Knowing what part of a model to consider for a specific use case is a direct hint to usage and implementation of the model and directly provides added value to a reader of the specification. As links to the specification can be found in 20 of the documents reviewed, this concept is common in specifications as of now.

If a use case can be linked to contents of the model, it may even be considered a candidate for an OPC UA Profile. While CSs may compile OPC UA Profiles in multiple ways, and not only based on the model contents, taking use cases as a basis is a structured way to communicate the relation between model and use case. As OPC UA Profiles are tested and can be certified, such a connection clearly states what needs to be implemented to fulfil the use case. In that context, especially the aspects to provide requirements for the use case, the OPC UA mapping, possible variants, the data in question, preconditions, postconditions, the actors, triggers and sequence and the communication partner may all help in refining the use cases context as well as designing test cases for the specification that help to gauge if a use case can be realized.

An overview of the aspects with the relation to the viewpoints whether they have a context to the model and can thus be used for profiles and test cases is provided in Table 4.

Table 4. Use case aspects

Aspect	**Choose Specification**	**Implement Specification**	**Context to Model**
Background	X	X	
Scope	X	X	
Requirements	X	X	X
OPC UA Mapping	X	X	X
Variants	X	X	X
Data	X	X	X
Pre-/Postconditions		X	X
Actor		X	X
Trigger		X	X
Sequence		X	X
Contribution to Spec Intention		X	
Comm. Partner	X		X
Spec Intention	X		
Goal of Use Case	X		

In the categorization, the prevalence of mere data transfer underlines the importance OPC UA specifications take in many domains: providing coordinated data in a defined way. OPC UA offers further ways of interaction between systems, like file transfer or method calls. These are mentioned less frequently in the use cases. However there are 80 use cases in the categories “Operation”, Engineering” “Write values” and “Service” that concern topics using these concepts. So the relevance of interface functions other than reading access is present in CS in more than a quarter of use cases.

When considered from the viewpoints stated, creating a model and implementing a model, some categories seem of greater interest than others. When creating a model, readers want to know if the concepts in the specification fit their own needs directly or if the concepts in the specification can be adapted to their needs. All categories may include such concepts. However, for the categories “communication with external system”, “adhere to 3rd party protocol” and “standardized interface generic” it seems less likely they include concepts that are applicable in a different domain. For the category “identification” in theory a generic solution for a digital nameplate that all CSs can reuse would be ideal. In that case the use case description would become a notice that the generic concept is used with the possibility to state special applications (e.g., additional values for the domain).

In general, more generic approaches are more likely to be useful from the viewpoint of creating a model. Generic approaches have a greater potential to fit to a variety of domains and needs. Rather specific use cases are interesting if the domains or concepts are closely related, they can also be handy in providing examples for a model design.

When taking the viewpoint to implement a specification, the context of the use cases becomes of interest, e.g., in the category “analyze/use data to a defined goal”. The analysis method and the goal for analysis can contribute to both the understanding of the data itself and its use. In addition, the use cases that use a communication method different from mere consumption are of special interest. With their descriptions, readers interested in implementing the specification are directly pointed to considering the correct means of communication. Especially if the system they are implementing is expected to behave a certain way, the use case description can aid understanding in conjunction with the specification text.

6. Conclusion

The use case sections of 51 CSs are analyzed regarding their level of detail, and 289 individual use cases have been identified and categorized. For the use case descriptions, aspects and good practices have been elicited and stated.

A main finding is that the CSs defined so far often focus on the existence of a standardized interface for the domain. However, in about a third of the use cases, a special behavior is expected by both communication partners in a data exchange or the use of OPC UA methods.

At the same time, the use case descriptions are found to be rather compact, by providing at most three different aspects for 80% of the specifications. By discussing the benefits of the use case description for two groups of readers – one with the aim to design an OPC UA model, the other one with the aim to implement a CS – the various aspects in descriptions found in the specifications all prove their merits.

So for editors of specifications, Table 4 provides an overview of aspects to consider. Not every aspect will be applicable to every specification and use case, but they can serve as a starting point to provide readers with a more nuanced context to use cases. It is also advisable to describe the same aspects for every use case in a structured way and provide figures and links to the specification if applicable. If the use cases are directly related to the model, defining OPC UA Profiles for the use cases may be considered.

The findings of this work will be provided to the OPC Foundation as a suggestion for the CS Template.

Acknowledgement

The Project "Clean OPC UA Information Modeling (CLOU)" (23084 BG) is supported within the programme for promoting the Industrial Collective Research (IGF) of the German Federal Ministry of Economic Affairs and Climate Action (BMWK), based on a decision by the German Bundestag.

References

[1] International Electrotechnical Commission. IEC TR 62541-1:2020 OPC unified architecture – Part 1: Overview and Concepts. ISBN 978-2-8322-9076-7. 2020.
[2] OPC Foundation. OPC 20020: Companion Specification Template. V 1.01.19, 2025.
[3] OPC Foundation. OPC 30000: PLC Model based on IEC 61131-3. V 1.02, 2020.
[4] OPC Foundation. OPC 30010: AutoID Devices. V 1.01.1, 2021.
[5] OPC Foundation. OPC 30030: BACnet. V 2.00.1, 2023.
[6] OPC Foundation. OPC 30040: AutomationML. V 1.00, 2016.
[7] OPC Foundation. OPC 30060: Tobacco Machinery. V 2.00.1, 2022.
[8] OPC Foundation. OPC 30081: Process Automation Devices – PADIM. V 1.01, 2024.
[9] OPC Foundation. OPC 30090: Fiel Device Tool. V 1.01, 2021.
[10] OPC Foundation. OPC 30110: Powerlink. V 1.00, 2017.
[11] OPC Foundation. OPC 30120: IO-Link Devices and IO-Link Masters. V 1.00, 2018.
[12] OPC Foundation. OPC 30130: CSP+ForMachine . CCLink. V 1.00, 2017.
[13] OPC Foundation. OPC 30140: Profinet. V 1.00.1, 2021.
[14] OPC Foundation. OPC 30141: PROFIenergy. V 1.00, 2021.
[15] OPC Foundation. OPC 30142: PROFI-RemoteIO. V 1.00.1, 2022.
[16] OPC Foundation. OPC 30143: PROFI-Encoder. V 1.00, 2023.
[17] OPC Foundation. OPC 30144-1: PROFINET GSD. V 1.00, 2023.
[18] OPC Foundation. OPC 30145: PROFINET Drives. V 1.00, 2024.
[19] OPC Foundation. OPC 30250: DEXPI P&ID. V 1.00, 2021.
[20] OPC Foundation. OPC 30500-1: Laboratory and Analytical Devices. V 1.0.0, 2023.
[21] OPC Foundation. OPC 34100: Energy Consumption Management. V 1.0.0, 2024.
[22] OPC Foundation. OPC 40001-1: Machinery Basic Building Blocks. V 1.03.0, 2023.
[23] OPC Foundation. OPC 40001-2: Machinery Process Values. V 1.00, 2023.
[24] OPC Foundation. OPC 40001-3: Machinery Job Management. V 1.0.1, 2024.
[25] OPC Foundation. OPC 40001-101: Machienry Result Transfer. V 1.00, 2022.
[26] OPC Foundation. OPC 40010-1: Robotics – Vertical Integration. V 1.00,2019.
[27] OPC Foundation. OPC 40020-1: Cranes&Hoists – MotionDevicesSystemBase. V 1.00, 2023.
[28] OPC Foundation. OPC 40077: PlsticsRubber – Injection Moulding Machines to MES. V 1.01, 2020.
[29] OPC Foundation. OPC 40082-1: PlasticsRubber – Peripheral devices – Temperature control devices. V 1.01, 2020.
[30] OPC Foundation. OPC 40083: PlasticsRubber – General Types. V 1.03, 2021.
[31] OPC Foundation. OPC 40084-1: PlasticsRubber – Extrusion – General Type Definitions. V 2.00, 2022.
[32] OPC Foundation. OPC 40100-1: Machine Vision – Control, Configruation management, recipe managemen, result management. V 1.00, 2019.
[33] OPC Foundation. OPC 40100-2: Machine Vision – Asset Mgmt and Condition Monitoring. V1.00, 2024.
[34] OPC Foundation. OPC 40200: Weighing Technlogy. V 1.01.0, 2023.
[35] OPC Foundation. OPC 40210: Geometric Measuring Systems.V 1.00, 2023.
[36] OPC Foundation. OPC 40223: Pumps and Vacuumpumps. V 1.00, 2021.
[37] OPC Foundation. OPC 40250-1: Compressed Air Systems – Main Control System. V 1.00.1, 2021.
[38] OPC Foundation. OPC 40301: Flat Glass Processing. V 2.00, 2024.
[39] OPC Foundation. OPC 40400-1: UA for Powertrain. V 1.00, 2024.
[40] OPC Foundation. OPC 40450-1: Joining Systems Base. V 1.00, 2024.
[41] OPC Foundation. OPC 40501-1: Machine Tools – Monitoring and Job Management. V 1.02.0, 2024.
[42] OPC Foundation. OPC 40502-1 CNC Systems. V 1.00, 2017.
[43] OPC Foundation. OPC 40503-1: Metal Forming. V 1.00, 2023.
[44] OPC Foundation. OPC 40504-1: Cutting Tools – Manufacturing. V 1.0.0, 2024.
[45] OPC Foundation. OPC 40530: UA for Laser Systems. V 1.0.0, 2023.
[46] OPC Foundation. OPC 40550-1: Woodworking Machinery – Vertical Interface. V 1.01, 2023.
[47] OPC Foundation. OPC 40560: Mining – General. V 1.00, 2022.
[48] OPC Foundation. OPC 40569-1: Mining – ACandUC – LongWallOperation. V 1.00, 2022.
[49] OPC Foundation. OPC 40569-1: Mining – ACandUC – DiscontinuousLoading. V 1.00, 2022.
[50] OPC Foundation. OPC 40569-3: Mining – ACandUC – DiscontinuousUnloading. V 1.00, 2022.
[51] OPC Foundation. OPC 40569-4: Mining ACandUC – Dozing. V 1.00, 2022.
[52] OPC Foundation. OPC 40600: Weihenstephan Standards. V 1.00, 2021.
[53] OPC Foundation. OPC 40740: Process Air Extraction and Filtration Systems, V 1.0.0, 2023.