A Functional Classification of Text Annotations for Engineering Design

Pedro Company
Universitat Jaume I, Castellón, Spain. pcompany@uji.es

Jorge D. Camba
Purdue University, West Lafayette, IN, USA. jdorribo@purdue.edu

Stanislao Patalano
University of Naples Federico II, Naples, Italy. patalano@unina.it

Ferdinando Vitolo
University of Naples Federico II, Naples, Italy. ferdinando.vitolo@unina.it

Antonio Lanzotti
University of Naples Federico II, Naples, Italy. antlanzo@unina.it

Abstract

Describing and supplementing geometric shapes (parts) and layouts (assemblies) with relevant information is key for successful product design communication. 3D annotation tools are widely available in commercial systems, but they are generally used in the same manner as 2D annotations in traditional engineering drawings. The gap between technology and practices is particularly evident in plain text annotations. In this paper, we introduce a functional classification of text annotations to provide an information framework for shifting traditional annotation practices toward the Model-Based Definition (MBD) paradigm. In our view, the current classification of dimensions, tolerances, symbols, notes, and text does not stress the inherent properties of two broader categories: symbols and text. Symbol-based annotations use a symbolic language (mostly standardized) such as Geometric Dimensioning and Tolerancing (GD&T) to provide precise information about the implications of geometric imperfections in manufacturing, whereas notes and text are based on non-standardized and unstructured plain text, and can be used to convey design information. We advocate that text annotations can be characterized in four different functional types (objectives, requirements, rationale, and intent), which should be classified as such when annotations are added to a model. The identification and definition of a formalized structure and syntax can enable the management of the annotations as separate entities, thus leveraging their individual features, or as a group to gain a global and collective view of the design problem. The proposed classification was tested with a group of users in a redesign task that involved a series of geometric changes to an annotated assembly model.

Keywords: annotations, Model-Based Definition, text annotations

1. Introduction

Annotations are used extensively in engineering and manufacturing to document the design of industrial products. They are usually represented as blocks of text anchored to specific aspects of a CAD model, assembly, or drawing and used as pointers to draw the attention to a particular area of the document [DBP09]. Annotations enrich CAD models and assemblies with information that paves the way towards the Model-Based Definition (MBD) paradigm which, within the context of a digital product definition data set, contributes to convey a product's definition that enables a Model-Based Enterprise (MBE).

From a software development standpoint, most CAD vendors have developed specialized tools (usually via "Product and Manufacturing Information" (PMI) or "Model-Based Definition" (MBD) modules) to manage CAD annotations and support model-based annotation practices. These efforts have focused mainly on the manipulation of annotations and providing location criteria and visualization filters to mitigate visual cluttering on the screen [CCJ14a]. CAD systems allow users to create annotations, link them to the corresponding feature in the model/assembly/drawing, and organize annotations in groups, facilitating interaction and visualization. However, most grouping criteria are view-centric, as they rely on the specific views of the models/drawings where the notes are intended to be displayed. We advocate for a model-centric approach, in which annotations are arranged in a functional manner that enhances the information associated with the model.

It is well known that internal communications, particularly in Small and Medium Enterprises (SMEs), relies heavily on informal and personal interactions, which often complement the formal information conveyed by engineering drawings and other design specifications. The goal of our proposed classification is to pave the way for the development of new tools that can explicitly capture this information and embed it within a CAD model that is easy to interrogate, thus eliminating alternative channels of communication to the main flow of CAD data. Additionally, these tools should enable the exchange of annotations (importing and exporting) with the model to ensure reliable data interoperability between the native CAD model and its derivatives used in downstream processes.

This paper discusses a framework to effectively convey design information through structured textual annotations linked to CAD models and assemblies by proving a functional classification that can facilitate the shift from current criteria that groups textual annotations based on the spatial orientation (i.e. "annotation views") to an approach where information is structured based on function.

The paper is structured as follows. First, the concept of Model-Based Enterprise and its implications for 3D model annotation is reviewed, and the currently available annotation tools is briefly discussed. The review also considers the different levels of adoption of model-based practices in industry and how companies are transitioning from drawing-based environments to model-based environments, and the various scenarios where drawings and models coexist. Next, a standardized set of concepts, practices, and criteria for organizing the information conveyed by text annotations in CAD models is introduced. In addition, assuming entities are grouped into basic categories, a classification (i.e. a formal definition of types, properties, and relationships between entities) is proposed built on the idea that text annotations belong to different functional types, which should be classified and managed separately to leverage their particular characteristics. A simple structure of text annotations is also suggested to eliminate misinterpretations and enable automatic parsing procedures that connect and cluster individual annotations. Finally, the results of a pilot study with a group of participants are discussed to validate the proposed classification and the paradigm shift that the classification contributes to.

2. Background

In the MBE paradigm, product models are the primary source of information that support the design, analysis, and manufacturing of products [Fre11]. 3D CAD models enriched with annotations serve as the central element from which all engineering activities, processes, and outputs flow throughout the entire product lifecycle [Whi12]. The paradigm has evolved mainly in relationship to products with a strong mechanical basis, for which mechanical CAD applications (MCAD), usually parametric and history-based, are used.

The parametric history-based paradigm, however, is only implemented in the form of native file formats, which are usually proprietary and application specific. The lack of standard representations that can fully support the exchange of parametric and history-based information represents a challenge for CAD interoperability. In the words of Schätzle [Sch16], “the structures for the representation of construction history, parameters, constraints and features are not implemented into today's CAD environments. Likewise, it could not be determined if and when CAD vendors plan to include these structures.” This statement implies that the capability to work with annotations is also limited.

It has been stated elsewhere that the implementation and deployment of the 242-protocol described in ISO 10303 [ISOb] is a necessary but not sufficient condition for successful interoperability [CCP23], as the MBE paradigm is supported by the notion that CAD models must be enhanced with design information that is linked to the model. In the following subsections, we review the relevant literature on important topics that relate to our work.

2.1 Structured messages

Our proposed approach advocates for structuring annotations. In this regard, natural language provides a familiar and flexible mechanism to construct messages and favors the communication of rich and complex information. Freeform messages in the form of annotations are useful for communicating information that is self-contained and understandable (at least within controlled contexts), disconnected from other freeform messages, and typically has a short lifespan. However, annotated freeform messages are sometimes intended to convey interrelated messages with longer lifespans. In these cases, natural language messages may lack the required property of having only one meaning, or monosemy. In addition, they are difficult to classify and thus unsuitable for making connections to other messages and building a global communication network of complex design and manufacturing information.

Using the terms *univocal* and *dialogic,* which are used to distinguish the two main types of educational discourses [OES15], we can state that technical communication should favor univocal discourse, which is characterized by communication in which the listener receives the exact message that the speaker intends. Natural language communication is prone to support dialogic discourse. Proactive actions are thus required to ensure that engineering messages conveyed through natural language become univocal. The problem is challenging for various reasons. For example, there are obvious and well-documented cultural differences that affect the capability to produce univocal communications [Wei98] [DJH02]. The peculiarities of a particular language may also affect the communication. For instance, there is certain disagreement about using passive voice in technical writing in English. In this particular case, Wolfe [Wol09] suggests that we should "analyze the audience and rhetorical purpose and select a voice accordingly."

Natural language processing tools and techniques such as those described in [APM15] could be adapted and applied as a long-term strategy to facilitate communication. We note that a particular branch of Natural Language Processing (NLP) is tailored to Requirements Engineering (NLP4RE). While NLP4RE was originally aimed at solving software requirements engineering problems, it was later adapted to include industrial product requirements [LMM12]. Some surveys and reviews in this space include Ferrari's [Fer18] and Zhao et al.'s [ZAF21].

Although some current NLP approaches may suit our purpose [KGG21], they are usually complex which makes adoption challenging. In addition, we intend to use these approaches for indexing messages as described by Deerwester et al. [DDF90] in order to effectively interconnect all messages. In the short term, we selected a simpler strategy consisting on basic guidelines to structure the messages. Our approach aligns with the structured natural language described by Wilson [Wil99] and Uusitalo et al. [URM11], and the controlled natural language (CNL) approach for the specification of functional system requirements proposed by Holtmann et al., which restricts the expressiveness of natural language to disambiguate it [HMV11].

2.2 Annotation tools

Data for enriching CAD models can be organized in attributes and annotations. Attributes are metadata embedded into the model that describe certain qualities. For example, the mass of a part is an attribute of the part. Annotations are instantiations of attributes which are displayed via notes or symbols. For example, the mass of a part can (optionally) be displayed by a note attached to the representation of the part. Two characteristics condition the enrichment of models through annotations: (1) many annotations are based on standardized symbols (GD&T, welds, inspection, etc.) but not all annotations are normalized, and (2) not all attributes are displayed using annotations. In the long term, model annotations will be able to serve as vehicles for accessing and interacting with all the information related to a design. The goal is to provide an environment where all interrogations and interactions can be performed via the annotated model.

Modern CAD systems provide tools to create and link annotations to geometry. Annotation tools also allow users to control how annotations are displayed via grouping and filtering mechanisms such as "annotation views", which act as "layers" of content to avoid the confusion caused by unnatural orientations of the annotations (see Figure 1) as well as reduce cluttering on the screen when working with highly annotated models. A common strategy when using annotation managers consists on adding annotations to specific annotation views based on the orientation—instead of the nature—of the feature that is being annotated.

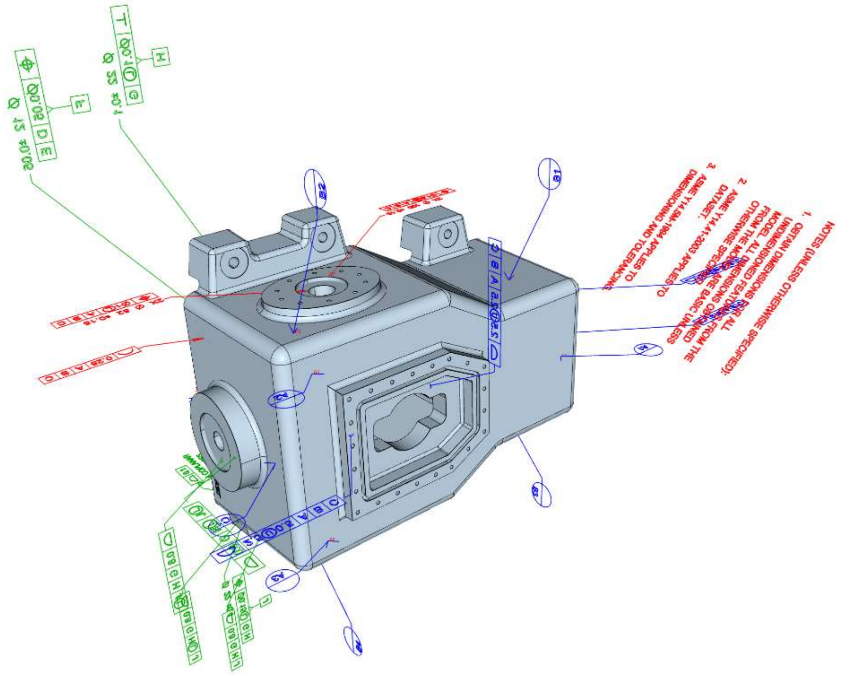


Figure 1. Unnatural orientation of the annotations in an annotated 3D model (model courtesy of NIST: https://pages.nist.gov/CAD-PMI-Testing/graphical-pmi-viewer.html)

In a 3D context, the grouping of annotations by "views" mimics the strategies designed to enrich 2D drawings. Therefore, it is valid but should not inhibit alternative – and semantically richer – grouping strategies and criteria such as grouping by function and/or by related elements in the model. Indeed, functionally related annotations are not always linked to features that share a common orientation. More advanced grouping and organization strategies are needed, some of which can enable data-mining techniques and knowledge management mechanisms. It is important to note that this paper does not claim to provide improved automated methods for knowledge management or knowledge extraction. We speculate that our classification could be indeed beneficial in these tasks. However, the focus of this study is on leveraging text annotations to facilitate the reuse and redesigning of CAD models while maintaining the quality of the original models, which has direct implications in tasks such as knowledge management [CCJ14a] [CCJ14b]. We can use a software engineering analogy to illustrate how annotations can facilitate these roles. In software engineering, source code comments are commonly used to enrich and document source code for maintainability and reusability purposes (main goal). In addition to methodologies and practices to maximize the readability and structure of code, many systems have been proposed to automatically generate design documentation from source code comments, or to study various aspects of the software development process such as productivity and code reusability (additional benefit). The same line of reasoning can be applied to 3D annotations in parametric solid models. The benefits of this vision are particularly relevant in the case of complex models (with perhaps hundreds of features) and assemblies with large numbers of components. In this paper, we are proposing an annotation-based classification that can lay the foundation for the development of mechanisms where models can be commented, explained, described, and ultimately documented to enable adaptability and reusability.

Some systems have been proposed to automatically generate metadata from annotations to enable high-level analysis and management of the information [CCJ14b]. These information flows would allow, for example, connecting and verifying design requirements or simulation data directly to dimensional parameters, constraints, and other elements in the 3D model. Likewise, design intent information related to modeling could be made available in the PLM system, which could be useful for traceability in engineering change scenarios that involve geometry. The embedded information can also facilitate reporting and the automatic generation of design documentation.

In our view, queries are dynamic requests for information associated with certain parts of the model. Annotated models should allow for different types of queries, including displaying model attributes, identifying features and geometric elements in the model, and highlighting datums and supplemental geometry associated with annotations. In general, CAD systems support the interrogation of annotated models, particularly if the annotations are symbolic, as they are more standardized. However, to the best of our knowledge, a standardized protocol to link text annotations to geometry in a manner that facilitates querying a model on the contents of the annotations and vice versa is still missing. This feature is key, since a set of annotations can quickly become impractical as the amount of annotations increases, unless efficient grouping strategies and filters are available to ensure that only contextually relevant information is displayed.

2.3 Annotated models and drawings

Drawing annotation has traditionally been the dominant strategy for including design and manufacturing information in engineering project documentation. Today, annotated drawings remain as the classic tool for documenting projects, but new alternatives are available. Annotated models, as described by the ASME Y14.41 and ISO 16792 standards (ASME Y14.41-2019 [ASM], ISO 16792-2021 [ISOa], support the adoption of paperless strategies to design, analysis, and manufacturing based entirely on electronic documents. These electronic documents can be 3D CAD models, which also eliminates the need for 2D CAD drawings.

Annotated models are also relevant in the context of Model-Based Systems Engineering (MBSE), particularly in the management of requirements. Like annotations, MBSE environments provide mechanisms for recording design information and even connect it to a CAD model. However, these connections are not direct. To the best of our knowledge, no MBSE mechanisms can connect design information to specific parametric 3D geometry, at least not at a level where information in a SysML diagram can drive a dimension in the model and change the geometry, for example, or where the parametric model can be automatically checked against a set of requirements specified in the SysML diagram. In this regard, 3D annotations serve as a bridge by making design information explicit in the model and accessible to other systems.

2.4 Types of annotations

Model annotation standards such ASME Y14.41 and ISO 16792 distinguish among five types of annotations: dimensions, tolerances, symbols, notes, and text. In our view, annotations can be classified in two broader categories: symbols and plain text. Symbolic annotations are highly standardized and available in most commercial CAD systems. For example, Geometrical Product Specifications (GPS) is a standard symbolic language that defines the representations and methods for describing geometric characteristics that are critical for manufacturing products. GPS focuses on determining the critical geometric conditions of a part and specifying the permissible levels of error.

To specify the allowable level of error, GPS provides a clear and concise symbolic language that enriches CAD models and engineering drawings, reduces the assumptions and controversies generated by other forms of requirement specifications, and ensures consistency for mathematical and computer processing. However, the symbolic language in GPS is incomplete. Although three types of GPS symbols coexist: (1) manufacturing processes and surface quality, (2) tolerances and dimensional fits, and (3) geometrical tolerances (GD&T), other symbols such as welding manufacturing procedures have been left out.

Alternatively, text annotations, which are suitable for conveying design intent information, have received significantly less attention than their symbolic counterparts. Our goal in this paper is to define a new framework and establish new formal methods for enhancing CAD models using text annotations.

3. A proposal for a classification of text annotations

We propose a new framework to enhance CAD models using text annotations that acts at two complementary levels. First, the messages must be *structured* to reduce the chances of misinterpretation, and, most importantly, to enable automatic parsing capabilities for message clustering. Second, the messages should be *classified* according to the functionality they describe. We note that the different strategies to store the annotations (internally, within the CAD model; externally, in a separate repository; or hybrid) are not relevant to the proposed classification [CCJ14b]. In other words, the characteristics of the channel used to convey the annotations are not considered. We focus on the functional aspect of the annotations.

The two levels of the framework are discussed next.

3.1 Structured messages

Our view is that a small set of patterns can cover most simple sentences, thus facilitating manual analysis and classification. Additionally, structured sentences with standardized vocabularies of “technical language” increase the quality of the annotations (as they favor the consistency, conciseness, and clarity of the messages). See, for example, the “Commonly Used Words and Phrases” in section 4.27.2 of ASME Y14.100:2017 [ASM17]. The recommended message structure is limiting the messages to basic sentences with single clauses: *subject* + *action* + *object*. The basic guidelines we propose to construct a new message can be summarized as follows:

- Use verbs that describe the action (e.g., *seals*).
- Use nouns that describe the subject (e.g., *ring* seals).
- Use simple grammar constructions to describe the object (e.g., *ring seals cylinder*).
- Use adjectives and nouns only when the description of the subject needs to be clarified (e.g., *piston* ring seals cylinder).
- Use prepositions to reinforce descriptions (e.g., piston ring seals cylinder *by* compression *to* prevent fluid passage).
- Prioritize the least ambiguous and most specialized vocabulary (e.g., piston ring seals *combustion chamber* by compression to prevent fluid passage).

3.2 Functional classification

Through a characterization process based on common categories used in engineering design, we distinguish four main functions of the messages in text annotations for design documentation:

1. Objectives, which describe what needs to be achieved, are aspirations or long-term goals in the design of new products (i.e., *what*).
2. Requirements, which describe the needs that must be satisfied, are the conditions (oftentimes quantifiable) that the design solution must meet (i.e., *how*).
3. Rationale, which describes the decisions that have been made, are the reasons or purposes behind the decisions that were made when specifying a design (i.e., *why*).
4. Intentions, which describe the expected behavior, are the decisions that are made to build the product and determine how the product will operate and react to redesigns (i.e., *what for*).

The motivation behind the selection of these functionalities is discussed next. We include descriptions of what they represent and references with further background, which allows reviewing their scope and content, to finally define a detailed classification of subtypes.

Objectives are descriptions of what needs to be achieved. There are two types of objectives: (1) general objectives constitute the essence of what is expected of the project, and (2) specific objectives describe the different aspects of the general objective in greater detail. There is usually a single general objective, which is met when all the specific objectives are met.

Objectives can be described by clear statements, which are generic for general objectives and concrete and detailed for specific objectives. General objectives can usually be captured in the names and/or titles of the models and assemblies. Statements for specific objectives can describe the *actions* that need to be taken for creating the product, or the *deliverables*. They are occasionally linked to CAD models and assemblies via text notes.

A relevant and meaningful title or file name in a CAD model or assembly can describe the general objective, such as the *non-return valve* depicted in Figure 2. adapted from the book by Company et al. [CG21]. Although polysemy cannot be fully avoided (e.g. other names can be used for this type of valve such as reflux, retention, one-way, etc.), a simple and effective description of the general objective can be clearly communicated by using a descriptive name.

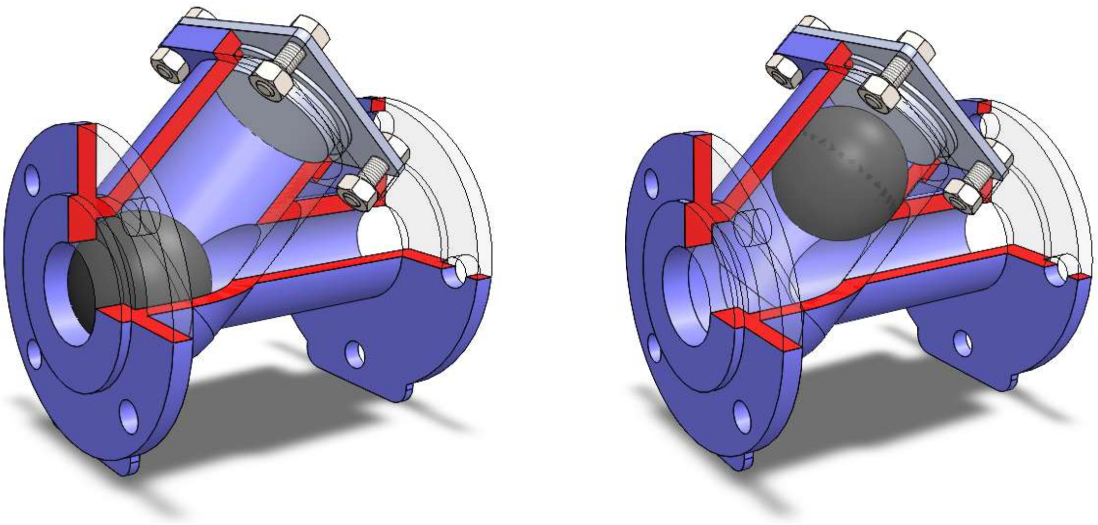


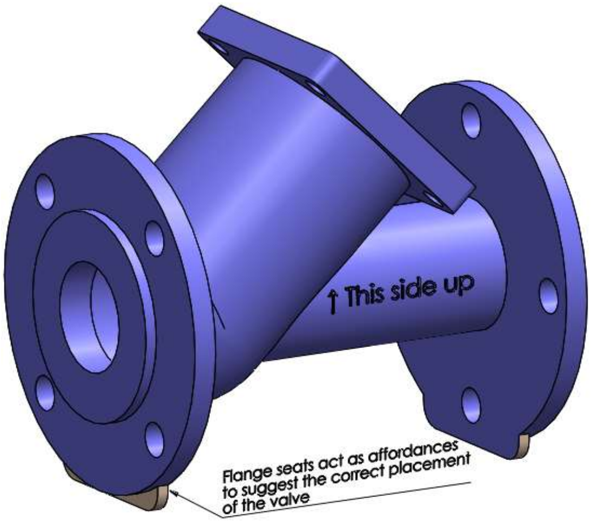
Figure 2. Pictorial view of a *non-return valve in closed (left) and open (right) positions*.

Requirements are the necessary conditions, features, attributes, and capabilities that a product, system, or service must have. Requirements can be further classified as *implicit* (technical limitations or legal regulations), or *explicit* (imposed by the manufacturer or the customers).

Managing requirements is a complex task. Large engineering projects can have hundreds or even thousands of requirements. In fact, PLM system often provide specialized modules dedicated exclusively to requirements management [KG08], [Leu09]. The most effective strategy for incorporating requirements into digital product models is to embed them into the geometric shapes, which does not require further clarification in textual form. For example, a bilateral symmetry in a part can inherently implement a design requirement for reversibility or directionality (e.g. USB C connectors must be reversible, while USB A connectors can only be inserted one way). Some academic approaches such as self-generative models [GKW15] or meta-parametric design [HS17] are built on this concept. Design affordances is another example. Perceived affordances are the characteristics of the product that intuitively facilitate the understanding of how to use it [Nor99]. Assembly affordances are features provided inside parts to make them easier to grasp, move, orient, or insert into an assembly, such as the flange seats shown in Figure 3.

Certain requirements may result in geometric shapes that are too complex for communicating the message explicitly. In such cases, an annotation is an effective mechanism to ensure that the requirement is made available and not ignored in future operations or redesigns. In general, text annotations are necessary but not sufficient to fully represent requirements. Design requirements must be described (linking the requirement with the proposed solution), explained (showing the interaction between the requirement and the solution), and justified (through motivated reasoning). Requirements become *specifications* when they are formulated in an objective and quantifiable manner. Requirements are called *restrictions* when they are formulated by indicating what is not acceptable, and *deliverables* when they must produce something tangible or intangible that is obtained as a result of a project.

For example, in the body of the valve shown in Figure 3, the flange seats are affordances that suggest that the part should be placed in a particular orientation, and clearly expose the technical limitation of the mechanism (i.e. it relies on gravity). The label engraved on the surface fully reinforces the need to position the part in a certain way, avoiding misinterpretations during assembly that could cause malfunction.



Flange seats act as affordances to suggest the correct placement of the valve

Figure 3. Flange seats act as affordances during the assembly of the non-return valve.

Rationale, as defined in ISO 10303-55:2005 [ISOc], captures and represents the decisions made during the design process and the reasons behind them. Shipman and McCall (1996) describe three perspectives on design rationale: argumentation, documentation, and communication [SM96]. In our view, communication is concerned with the capture of rationale, not with the content of the messages. Therefore, the design justification can be *argumentative* or *documentary*. The *argumentation* compiles justifications to serve as knowledge bases and know-how repositories and reason about methodological deficiencies that need to be improved in future designs. It is generally difficult to capture argumentative rationale. In addition, many designers are often reluctant to document their actions in a highly structured manner. Sophisticated procedures and tools such as Issue-Based Information System (IBIS) [KR70] can be used, but most of these methods rely heavily on human intervention, especially for interpreting and entering information into the system, and detach the information from the digital product model.

We note that applications within D-Agree, where AI supports the real-time processing of information, may reduce—or eliminate—these limitations. Although this type of AI-based approaches may lead to interesting hypotheses to inform future developments, the goal of this paper is to lay the foundation for design communication through annotated models by providing a classification that is simple enough to enable the shift from current criteria that group textual annotations based on the spatial orientation (i.e. “annotation views”) to an approach where argumentative and documentary rationale are separated.

Documentation focuses on communicating the justifications to people outside the project. Documentary justifications are less complete than argumentations, because they do not include information that was compiled for internal use of the project team. It is possible to enrich CAD documents with documentary justifications if the information is first captured and structured to the greatest extent possible, and then linked to the model.

Documentary justifications have two additional uses: *supervision* (to allow outsiders to understand, monitor and regulate, if necessary, what the project group does), and *managing intellectual property* (to identify and secure the intellectual property generated in a project). The former is the basis for memorandums; the latter result in images to illustrate patent documents.

The purpose of the justification determines the mode of representation. Historically, a supervisor's justifications were communicated as notes attached to the corresponding design drawings. These notes are now being attached to CAD models and assemblies. Likewise, the justifications for managing intellectual property are documented in specific forms with predefined formats that highlight originality and innovation. In this regard, technical drawings and CAD model images are essential for describing the technical characteristics of a product or invention, particularly for patent purposes. Linking patent claims to patent images through annotations improves the understanding of the concept.

The example shown in Figure 4 illustrates how an annotation linked to one of the holes in the flanges of the valve highlights the decision to shift the position of the holes to prevent incorrect mounting after repairs.

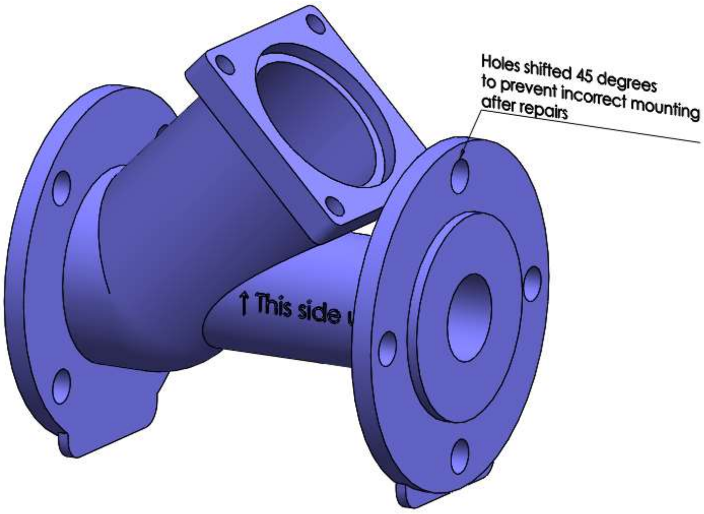


Figure 4. Rationale for the holes in the flanges of the non-return valve.

Intentions, as defined in ISO 10303-108:2005 [ISOd], describe the decisions that were made to build the product and determine how the product operates, and how it will react to changes and alterations. Expected behavior is thus the main type of intent [OCC18]. Nevertheless, we can distinguish between expected behavior during operation, and during redesigns [CNC20]. For example, in Figure 5, a group of annotations is used to describe the operating procedure of the mechanism where a ball moved by the water pressure and gravity controls the flow of liquid inside the valve. The annotations have been numbered as a simple strategy to emphasize that they are mutually related and to better suggest that each annotation conveys part of a global message.

In the context of intentions, some CAD systems provide tools that may be used as a complement—or even as an alternative—to textual annotations. As an example, the need to control the inclination of the pipe to guarantee the optimal action of gravity on the ball can be managed by an embedded alert or notification (also known as "sensor" in many CAD systems) on the corresponding angular dimension in the CAD model. The sensor alerts the designer when the angle of the inclined pipe changes to an unacceptable value. A software-based virtual sensor is a programming abstraction that calculates the value of a variable, instead of measuring it in the physical world. Therefore, virtual sensors in CAD software (also known as design sensors) monitor different attributes of the model or the assembly such as dimensions, mass, volume, etc., and automatically trigger a warning when the value moves out of a predefined range.

CAD sensors are useful for receiving notifications during the modeling process when a specific condition is met (e.g. the mass of the model is greater that a predefined value, or a particular dimension goes over a certain value), or for use in simulation to aid in optimizing designs. Despite their potential for informing design decisions, particular during redesign, CAD sensors are largely under-utilized. The focus of the functional classification proposed in this paper is plain text annotations, thus sensors are not considered.

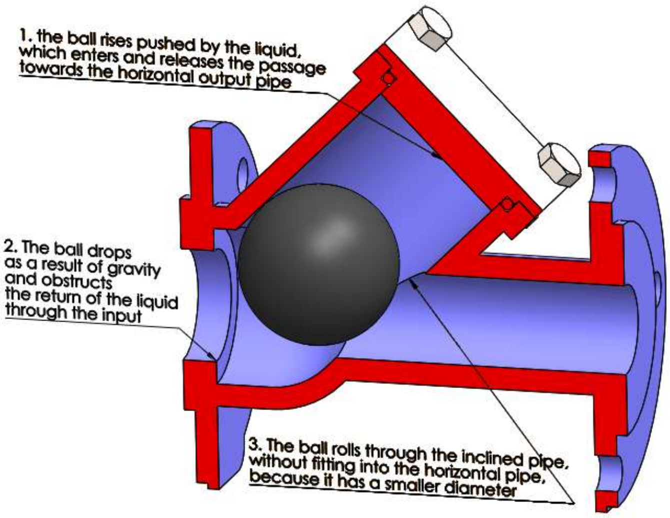


Figure 5. Annotations that describe intention by explaining the behavior of the ball during the operation of the non-return valve.

The proposed functional classification is summarized in Table 1.

Table 1. Functional classification of text annotations in product design.

PURPOSE	SCOPE	CONTENT	SAMPLE DATA

Objectives	General	General goal	Titles, file names, and descriptions
	Specific	Actions	Design document
		Deliverables	Design brief
Requirements	Implicit	Technical limitations and constraints	Standard checklists
		Laws and regulations	Mandatory codes, policies, and legal requirements
	Explicit	Manufacturer specifications	Functional specifications and Technical Requirements Documents
		Stakeholder conditions and requests	Product Requirements Documents
Rationale	Argumentative	Know-how and design decisions	Knowledge bases
	Documentary	Supervision Intellectual property	Revision tables and Memorandums Patents
Intentions	Use/operation	Operating procedures	Instructions and data-sheets
	Redesign	Modification procedures	CAD sensors Engineering Change Orders

It is important to emphasize that our classification is based on the function of the annotations, and not their form. For example, any type of annotation in our classification could include long textual explanations or even tables and diagrams, which would require defining other types of representation and storage mechanisms such as external files. In these cases, a model annotation can be thought of as a link that points to a more complex form of information storage. In any case, we consider this particular type of annotation a method to represent or store the information included in an annotation, but not a functional type of annotation in itself.

4. Experimental study

To validate our proposed functional classification of text annotations, we conducted a pilot experimental study with a group of participants using the assembly model of a wobble plate piston pump shown in Figure 6. The model was created in DS SolidWorks and adapted from an exercise described by Company et al. [CG21]. All models and assemblies ensure CAD quality as described by Company el al. [CCO15] and Otey et al. [OCC19]. In other words, they were refined to remove what Rosso et al. called “smells” [RGB22], which could prevent or difficult the intended redesign. A set of relevant functions was identified, and the corresponding annotations were added to the parts and assemblies. The annotated parts and assemblies (“annotated models”) were then distributed to a group of participants who were asked to complete a redesign task.

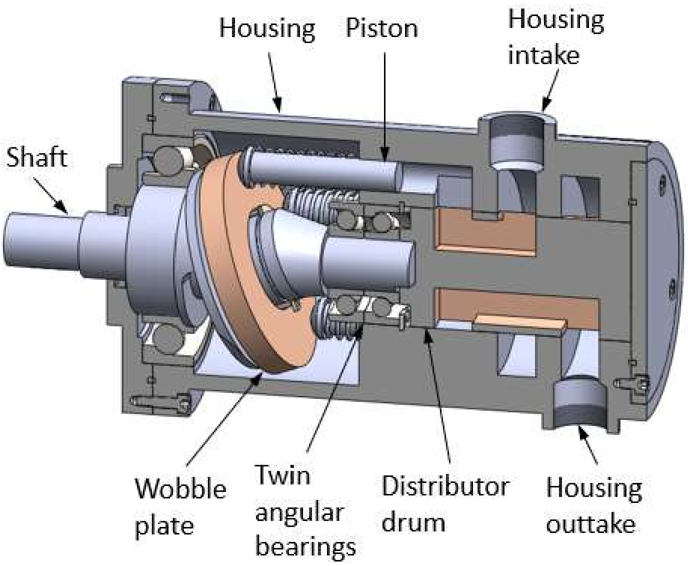


Figure 6. Section view of the wobble plate piston pump with labels identifying main components.

4.1 Structured annotations

First, the general objectives of the product were established and a set of requirements were defined. The design was refined as more specific requirements were established and suitable rationale was developed. Finally, the procedural model was analyzed to determine the relationships between the design intent and the procedure used to build the model, as well as to facilitate changes and future redesigns. Through this analysis, we defined a set of relevant annotations which were classified based on functionality as shown in Table 2.

We note that in some cases, annotations within or across the functional groups may be tightly coupled. An annotation in a particular group may yield other annotations that supplement the original annotation or provide additional details to the model. For example, in the group “laws and regulations” in Table 2, the annotations L1 and L2 (“If pump is anchored to a foundation, a specific mounting bracket is required” and “If pump is anchored to a foundation, top surface of the foundation slab should be held flat…”) produce annotations M3 and M4 in the “manufacturer specifications” group (“Pump unit should be mounted on a fabricated steel or channel base plate, which should be mounted on a concrete foundation” and “Concrete foundation should be 100 to 200 mm longer and wider than the baseplate.”)

Table 2. Annotations added to the CAD models and assembly of the wobble plate piston pump (annotations and their codes have been translated to English from the original table in Spanish used in the experiment).

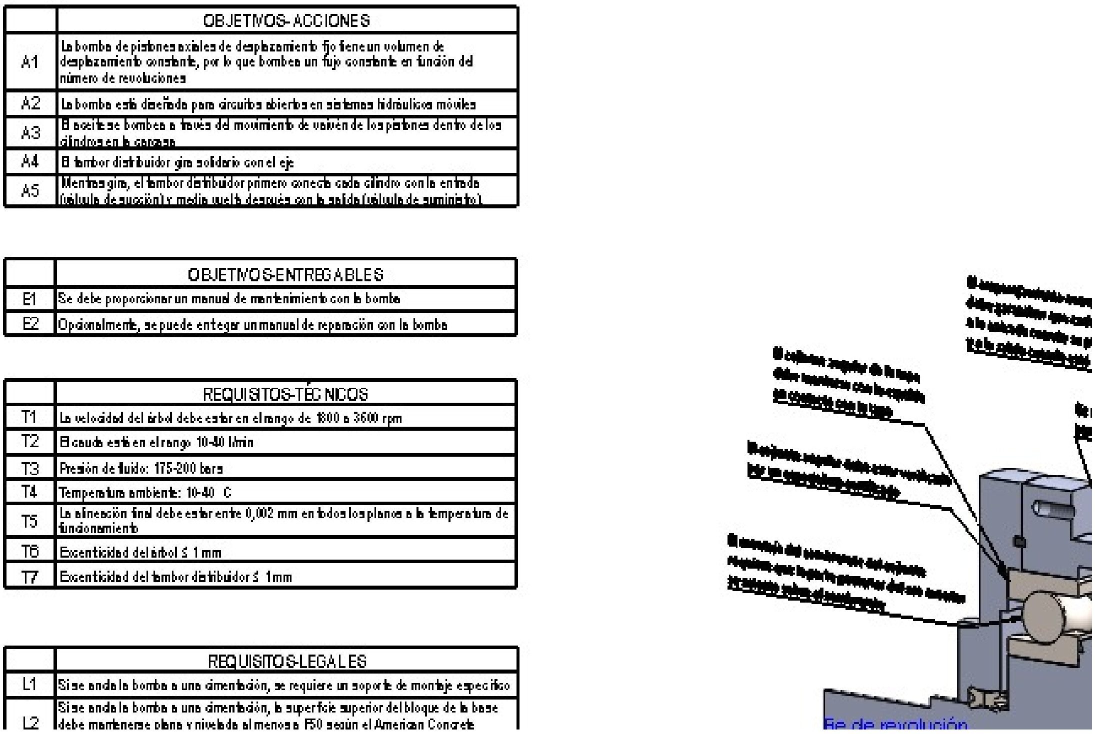
PURPOSE	SCOPE	CONTENT	ANNOTATIONS
Objectives	General	General goal	• (G1) Wobble plate piston pump (title) • (G2) Fixed displacement axial piston pump (subtitle)
	Specific	Actions	• (A1) The fixed displacement axial piston pump has a constant displacement volume, thus pumping a consistent flow as a function of the number of revolutions. • (A2) The pump is designed for open circuits in mobile hydraulic systems.

			• (A3) Oil is pumped through the back and forth motion of the pistons inside the cylinders in the housing. • (A4) The distributor drum rotates jointly with the shaft. • (A5) While rotating, the distributor drum first connects each cylinder with the intake (suction valve), and with the outtake (supply valve) after half a turn.
		Deliverables	• (D1) A maintenance manual must be supplied with the pump. • (D2) Optionally, a self-repair manual can be delivered with the pump.
Requirements	Implicit	Technical limitations and constraints	• (T1) Shaft speed must be in the range 1800 to 3600 rpm. • (T2) Flow rate is in the range 10-40 l/min. • (T3) Fluid pressure: 175-200 bars. • (T4) Ambient temperature: 10 to 40° C. • (T5) Final alignment should be within 0,002 mm in all planes at operating temperature. • (T6) Eccentricity of the shaft ≤ 1 mm. • (T7) Eccentricity of the distribution drum ≤ 1 mm.
		Laws and regulations	• (L1) If pump is anchored to a foundation, a specific mounting bracket is required. • (L2) If pump is anchored to a foundation, top surface of the foundation slab should be held flat and level to at least F50 according to American Concrete Institute (#117) and the Canadian Standards Association (#A23.1). • (L3) Pump must comply with the basic safety requirements set out in directive on machinery: 2006/42/CE. • (L4) Pump must comply with the basic safety requirements set out in directive on low tension: 2006/95/CE. • (L5) Pump must be REACH (Registration, Evaluation, Authorization and Restriction of Chemicals) compliant.
	Explicit	Manufacturer specifications	• (M1) Piping used should be as short and direct as possible. • (M2) Piping line size should be the same diameter as the suction valve. • (M3) Pump unit should be mounted on a fabricated steel or channel base plate, which should be mounted on a concrete foundation. • (M4) Concrete foundation should be 100 to 200 mm longer and wider than the baseplate. • (M5) The weight of the foundation should be 3-5 times the weight of the pump, motor and baseplate. • (M6) The foundation should be 100 to 200 mm longer and wider than the polymer concrete or fabricated steel baseplate. • (M7) Anchor bolt sizes: M24X3. Length: 200 to 250 mm depending on base thickness and overall size.
		Stakeholder conditions and requests	• (S1) Repair time to replace friction disc must be less than two hours. • (S2) Surface finish must be hard chrome coating. • (S3) Use ISO threaded connection for inlet/outlet.
Rationale	Argumentative	Know-how and design decisions	• (K1) Odd number of pistons was selected to reduce the risk of shaft stalling. • (K2) The pairing between the shaft and the distributor drum must ensure that each cylinder connects to the intake when its corresponding piston is in the backing position, and to the outtake when it is in the forward position (inserted into the cylinder). • (K3) The slot that connects the distribution drum to the shaft must be non-rotational symmetric to prevent incorrect assembly.
	Documentary	Supervision Intellectual property	• (P1) Angular bearing checked by certified bearing specialist. • (P2) Manufacturing process for the distributor drum is patent pending. • (P3) Nameplate containing the fundamental specifications must also indicate the pump patent number.
Intentions	Use/operation	Operating procedures	• (U1) Use rubber mallet for the similar fit mounting of the angular bearing with both the housing and the cover. • (U2) Twin angular bearings must be mounted back to back (O position). • (U3) Cover angular bearing must be mounted with its back in contact with the cover.

			• (U4) Never use force to align piping to the pump. • (U5) Ensure conformance to local codes: UL, CSA, TUV, FDA, ISO. • (U6) Do not operate the pump without proper guard. See ISO 14120 and ANSI/ASME B15.1-1996. • (U7) Before start up, distributor drum should be lubricated by filling the pump with water or the pumping fluid through the discharge or suction plug. • (U8) Voltage of the motor supply should not vary by more the 10% from the voltage on the nameplate. • (U9) Use inlet mufflers and isolation mountings to reduce noise levels to the low 30-dBA range (typical range is 40 to 60 dBA). • (U10) Direct drive is recommended for shaft loading. Indirect drive is permitted with special mounting. • (U11) During assembly, all parts must be coated in pump fluid or compatible grease. • (U12) Install plugs in intake and outtake to prevent contamination during packaging and shipping.
	Redesign	Modification procedures	• (R1) The axes of revolution of the main and oblique sections of the shaft must always intersect at a point on the theoretical plane of support for the pistons. • (R2) The positions of the ports of the distributor drum must align with those in the housing. • (R3) The diameter of the distribution drum must match the outer diameter of the twin angle bearings on the shaft. • (R4) The position of the intake (suction valve) is controlled by a reference datum plane. • (R5) The position of the outtake (supply valve) is controlled by a reference datum plane. • (R6) The depth of the cylinders is controlled by a reference datum plane. • (R7) The orientation of the intake (suction valve) is controlled by a datum axis sketched in the datum plane. • (R8) The orientation of the outtake (supply valve) is controlled by a datum axis sketched in the datum plane. • (R9) Reference datum planes of the housing and the distributor drum must be kept paired with each other while redesigning.

A sectional view of the fully annotated model is shown in Figure 7. Only the annotations that are connected to a specific geometric element are shown in the Figure. Annotations related to objectives and requirements were not connected directly to the geometry but provided in tabular form. The tables are visible in the CAD application's viewport next to the model.

The model is deliberately shown at a small scale to illustrate the arrangement and distribution of all the annotation in the model. The content of the annotations, which may not be fully legible in Figure 7, is shown in Table 2. Finally, the annotations in the Figure were translated to Spanish to ensure all the participants in our study could understand the information.



	OBJETIVOS-ACCIONES
A1	La bomba de pistones axiales de desplazamiento fijo tiene un volumen de desplazamiento constante, por lo que bombea un flujo constante en función del número de revoluciones
A2	La bomba está diseñada para circuitos abiertos en sistemas hidráulicos móviles
A3	El aceite se bombea a través del movimiento de vaivén de los pistones dentro de los cilindros en la carcasa
A4	El tambor distribuidor gira solidario con el eje
A5	Mientras gira, el tambor distribuidor primero conecta cada cilindro con la entrada (válvula de succión) y media vuelta después con la salida (válvula de suministro)

	OBJETIVOS-ENTREGABLES
E1	Se debe proporcionar un manual de mantenimiento con la bomba
E2	Opcionalmente, se puede entegar un manual de reparación con la bomba

	REQUISITOS-TÉCNICOS
T1	La velocidad del árbol debe estar en el rango de 1800 a 3600 rpm
T2	El caudal está en el rango 10-40 l/min
T3	Presión de fluido: 175-200 bars
T4	Temperatura ambiente: 10-40 C
T5	La alineación final debe estar entre 0,002 mm en todos los planos a la temperatura de funcionamiento
T6	Excentricidad del árbol ≤ 1 mm
T7	Excentricidad del tambor distribuidor ≤ 1mm

	REQUISITOS-LEGALES
L1	Si se ancla la bomba a una cimentación, se requiere un soporte de montaje específico
L2	Si se ancla la bomba a una cimentación, la superficie superior del bloque de la base debe mantenerse plana y nivelada al menos a F50 según el American Concrete

Figure 7. A general view of the fully annotated model of the valve used in our experiment (note that the annotations in the image are in Spanish, which was the language used in the experiment).

4.2 Annotated models

Participants in the study were divided into experimental and control groups. Two versions of the annotated wobble plate piston pump model were created. Participants in the control group were assigned the annotated models in which the annotations were grouped based on the orientation of the annotated feature (view-centric annotations). This grouping was relatively straightforward to create in the experimental model as the main geometry of the pump can be generated as a revolved solid and the main features can be analyzed through a single sectional view. Alternatively, the annotations provided to participants in the experimental group were arranged according to the proposed functional classification (function-centric annotations).

4.3 Redesign task

All participants were asked to complete a redesign task consisting on relocating the supply valve (outtake) which was originally placed at the bottom of the housing, and placing it next to the suction valve (intake), located at the top of the housing. Participants were also informed that in order to make room for the corresponding hose fittings, the supply valve must not only be rotated 180 degrees but also displaced 15 mm away from the pistons. It is expected that participants find out that the distribution drum must also be redesigned in order to fit the new placement of the outtake ring. The requested change is illustrated in Figure 8.

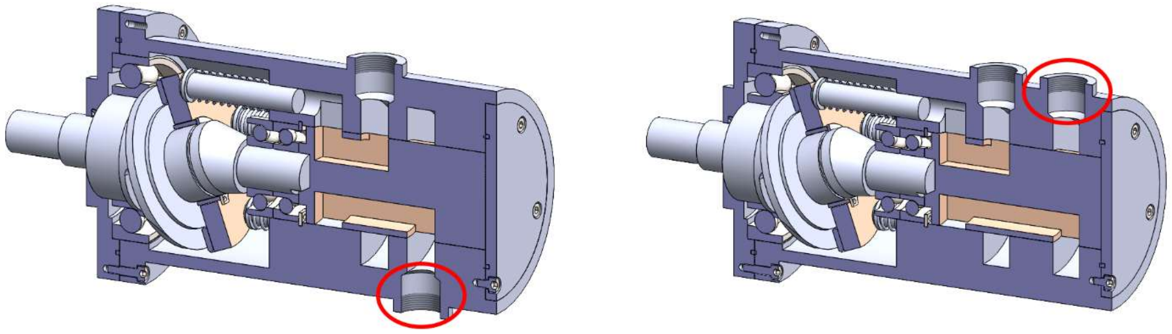


Figure 8. Original arrangement of the valves (left) and redesigned valves (right).

4.4 Analysis

To measure the effectiveness of the proposed classification in the experiment, we assessed the participants’ ability to successfully redesign the wobble pump as well as the time required to complete the redesign task. In addition, qualitative data in the form of interviews were collected from participants after completing the task.

Two CAD experts were initially interviewed, and then two groups of 14 and 19 graduate engineering students, respectively, at a Spanish university. All participants in the study had previous experience with parametric solid modeling tools and techniques as well as exposure to the fundamentals of model annotations, as explained in lessons 4.0, 4.0.1, and 4.4 in the book by Company et al. [CG21].

The first CAD expert was assigned the annotated assembly prepared for the experimental group (i.e., with annotations in functional groups) and the second expert received the assembly of the control group (i.e., no functional grouping for the annotations). The first expert was able to successfully complete the redesign task in 15 minutes, whereas the second expert required approximately 35 minutes to redesign the housing. Furthermore, it was observed that the second expert ignored the recommendations provided by the redesign annotations, as she felt that the task could be solved easily without this information. This strategy proved unsuccessful, as the participant left the distribution drum unchanged.

The average time required for the participants who succeeded was 32 minutes, whereas those who failed required 33 minutes. We suspect that none of the graduate students had reached the level of expertise to be considered experts (as the true experts in our case solved the experimental task in approximately half the time). Regarding the success/failure in redesigning the pump, 18 out of 33 (55%) of the graduate students succeeded in redesigning both the housing (which was the obvious task that was explicitly asked) and the distribution drum (which was an implicit complementary task that participants should discover on their own by reading the corresponding annotations). However, there was no correlation between success and experimental group, as eight of the eighteen students who completed the task were assigned the annotated assembly prepared for the control group. Instead, a correlation was found between the level of expertise of the students and their success in the experiments. Students with higher grades in previous CAD courses solved the redesign, whereas those with lower scores failed to do so. Students who succeeded in redesigning the pump had an average grade of 7.05/10 in their CAD course, whereas those who failed had an average of 5.90/10. We speculate that students are non-expert users that tend not to rely on the arrangement and grouping of annotations (even if they are encouraged to use them), and attempt to go through all the notes (thus taking significantly more time than experts).

After completing the redesign task, all participants were given a questionnaire and asked to provide feedback and reflect on the exercise. The following six questions were included:

1. Indicate the time taken to complete the task. Please describe any significant issues you had during the completion of the exercise, if any.
2. Did you find the annotations that were organized in tabular form easier to check and/or more informative than those connected to the model by leader lines? Explain your answer.
3. When interacting with the assembly model, were any of the annotations particularly helpful to complete the task that were asked to do? Explain your answer.
4. All the annotations in the exercise were linked directly to the assembly, instead of individual parts. Indicate whether you think this approach was effective, or you would have preferred to also have annotations on each part model. Explain your answer.
5. Did you use the annotations folders to control the visibility of the annotations so that only the relevant annotations were visible in the model? Explain your answer.
6. Write any additional comments or observations about the redesign task that you would like to share.

The purpose of the questionnaire was to validate our top-down approach in the use of common categories in engineering design to build our classification, which we then tested to determine whether their categories fitted the opinions of the participants. Since an open questionnaire with plain (unstructured) text was utilized, the participants' responses had to be interpreted by the research team, which is common practice in these types of analyses. Naturally, a certain degree of subjectivity is unavoidable. However, to the best of our knowledge, all the responses correlate highly with the perceived sense of the responses of the subjects. Detailed responses (in Spanish) can be made available in tabulated files upon request.

The following observations and conclusions were drawn based on the participants' responses to six questions:

1. The self-reported "perceived" time to complete the task matched closely the actual time measured by the e-learning platform that was used to distribute the exercise. We conclude that the task was not challenging to the point of being perceived as cumbersome, overwhelming, or too time-consuming. No significant issues were reported.

2.1 In general, participants claimed that annotations with leader lines were easier to use than tabulated ones. This was the case for 10 out of 14 participants in the control group. One participant preferred tabulated annotations and three agreed that annotations were helpful but without clearly distinguishing between the two types. In the experimental group, 12 participants out of 19 preferred annotations with leader lines, and three preferred tables. We speculate that the arrangement of the tabulated annotations in the experimental group may have made a difference.

2.2 Almost all participants agreed that the annotations in tabular form were easy to read, but difficult to relate to particular geometric aspects in the models and/or assemblies. This feedback seems to validate the strategy suggested in Section 3 to limit tabulated annotations to objectives and requirements, which are generally not linked to a particular feature of the model.

2.3. Leader lines provided intuitive cues about the relationship between the annotations and the geometry. The majority of participants resorted to the leader lines attached to the outtake as the main cue to identify the relevant redesign notes. However, linking annotations to geometry is difficult and imprecise, as further explained in Section 4.5.

3 Nearly all participants (30 out of 33) agreed that annotations were helpful to solve the task. Three provided generic reasons, but the other 27 detailed with precision which notes had been the most useful to them. Two of the three participants that claimed not to have used the annotations, but solved the problem by directly parsing the model tree were part of the control group.

4.1 Most participants (28/33) agreed that the assembly annotations were particularly helpful.

4.2 Only 5 out of the 33 participants suggested that linking annotations to individual parts only (not the assembly) would have been more effective. They argued that this strategy would have been helpful to keep the annotations available when opening the parts separately to perform the modifications. These participants seemed to ignore the fact that in order to redesign the assembly, they first needed to know which parts should be modified. Therefore, we speculate that annotating individual parts instead of the main assembly would have been less effective, as participants would have had to seek specific annotations across the assembly tree.

4.3 Eleven subjects suggested that replicating the annotations in the individual parts (in addition to the assembly) could be a bonus, which seems to support our hypothesis on the value of a tool for automatically duplicating relevant notes from the assembly onto the models to make the interaction more efficient.

5.1 Despite being familiar with model annotation concepts, eleven participants admitted having difficulties using practical annotation mechanisms and tools such as the *Annotations folder* in SolidWorks, which seemed to be a determining factor in their decision not to leverage the annotation filtering functionality in the exercise. Since only 5 subjects reported having actively used those tools to facilitate the task, we suspect many participants viewed the *Annotations folder* as a mechanism to control annotations in 2D views, instead of leveraging it as a valid tool for controlling 3D annotations.

5.2 The responses to the questionnaire validated our hypothesis that the experimental and control groups did not behave differently during the redesign task, simply because participants in both groups paid little to no attention to the arrangement of the notes. Most participants admitted that they went through all the notes until they found the information required to complete the redesign task. To further validate this hypothesis, we interviewed the participants a second time and asked: “Discuss the value and usefulness of the annotations for solving the given redesign problem. Explain, in particular, whether you find it useful to have the annotations in groups as well as your opinion on the grouping criteria in the assembly of the pump.” The responses clearly confirmed our hypothesis. We conclude that the time required by experts to solve the task correlates with the arrangement of the annotations, whereas students lack the habit of using the arrangements and groupings of annotations, and should be trained to do so.

6.1 Most participants reported that annotations helped them simplify the process of finding the feature and operations involved in the redesign.

6.2 Most participants stated that the experiment helped them become more aware of the importance of engineering design annotations, how to manage them properly in a model/assembly, as well as the difficulty to manage visual clutter when many annotations are displayed on screen.

6.3 Five participants highlighted the educational value of the experiment, which the research team had already anticipated. A related teaching strategy is proposed and described in Section 4.5.

4.5 Additional remarks

Some participants acknowledged the fact that the annotations were particularly well arranged and did not overlap. This layout along with the fact that all the annotations were located on the same annotation plane (the front plane) made the information easily accessible in the model. For this reason, participants felt it was unnecessary to use tools to interactively manipulate the visualization of the annotations to find the relevant information. However, this perception of the annotated model may be misleading for various reasons. First, arranging text annotations effectively in a 3D model is uncommon. The process is time-consuming and, in many cases, CAD applications automatically rearrange the notes according to their own criteria while the user interacts with the visualization tools. Second, even in models with well-arranged annotations, filtering mechanisms significantly facilitate the perception of the relevant information. As a lesson learned for future experiments, we suggest the use of models and assemblies with yet more annotations and arranged in multiple annotation planes, to compare different functional arrangements that leverage visualization tools.

One participant in the experiment stated that “It would have been helpful to have annotations in the parts that needed to be modified with the names of the features that made up the part to save time during redesign.” The participant was specifically referring to individual parts, but the same could be applied to the assembly. The comment suggests providing labels to visually identify the components (parts in an assembly or features in a model) as shown in Figure 6. Obviously, this information is already available in the corresponding model or assembly trees. Dynamic queries are also available in most CAD applications, where the corresponding component is highlighted when the user selects its name in the tree. Nonetheless, the functionality suggested by the user could be easily implemented in current CAD applications to further facilitate interactions: a set of annotations automatically created by the system to include the names of the components of a CAD model or assembly. Based on the classification shown in Table 2, this type of annotation would be classified in the “general objectives” category.

It is important to mention some of the challenges faced by the research team when preparing and interacting with the annotated model for the experiment (Figure 9) reported in this paper, as the experience can serve as an indicator of the lack of software mechanisms to support annotation technology in current commercial solutions. The lack of effective and intuitive tools or procedures to attach annotations to the desired annotation planes as well as the scope where the annotations need to be applied within the model are particularly relevant.

The CAD application used in this study (SolidWorks) provides an annotation mechanism (the *Annotation folder*) that allows users to create annotation planes and organize the model annotations accordingly. However, the exact location of the annotations cannot be controlled easily by the user. Since the annotation planes which contain model annotations are supposed to be displayed in annotated views, the exact depth of the annotations with respect to the global coordinate system is difficult to control, as it is considered irrelevant. Therefore, the annotation tools in the CAD system enable users to position notes and their leader lines relative to the two-dimensional coordinates system of the annotation plane, but not to control the depth of the actual annotation plane where each annotation is placed. The resulting depth differences in the annotation planes become apparent when the annotated models and assemblies are interacted with and rotated freely without aligning them with pre-defined “3D-views.”

For example, when adding annotations to the model shown in Figure 9, the annotation tool easily snaps to the front face of the geometric shape, placing the point of attachment of the annotation – the tip of the leader line – at any location on that face. As a result, the annotation plane is made coplanar to the front face. However, any faces produced by a section cut (e.g., through the front plane) are not detected. In addition, the tip of the leader line of the annotation can only be attached to the rectangular frame used to visualize the datum plane, and not the plane itself. Resizing the frame does affect leader lines that were placed on the plane previously.

Since the snap functionality is not a reliable mechanism to attach the leader lines of the annotations to the geometry of the model, we opted for adding supplemental geometric elements and use them as handles to link notes to geometry. Leader lines can be easily connected to reference points and frames of reference planes (previously resized to match the frame to the line of interest, such as the midline of the lateral face in the prismatic shape depicted in Figure 11).

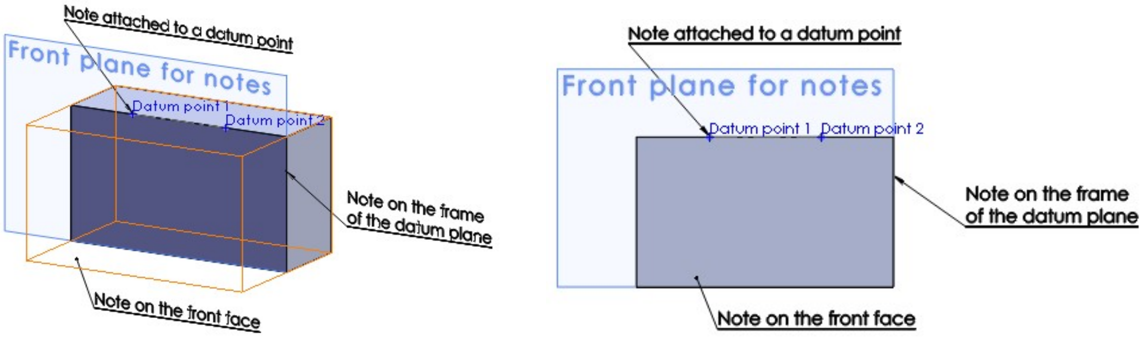


Figure 9. Annotations on planes with the same orientation but at different depths, or linked to different "handles."

The second challenge encountered by the research team when annotating the model of the wobble plate pump has to do with the scope of the annotations. More specifically, the level at which the annotations are created (i.e., part, subassembly, assembly). In the experiment, a decision was made to create all the annotations in the main assembly. From a user's standpoint, managing annotations created in sub-assemblies or parts from the main assembly is more complex as it requires managing the annotation tools (i.e., the *Annotation folders)* of each part/subassembly by navigating to their corresponding trees. Informal tests with expert users revealed that managing design annotations though multiple (and sometimes deeply nested) annotation folders is more difficult and confusing than managing models where all the annotations are centralized at the assembly level. We speculate that although the procedure may be adequate for users interested in annotating 3D models as an intermediate step toward annotating drawings, it is likely impractical to users interested in a true MBD approach, as it discourages the annotation of parts and subassemblies at lower hierarchical levels.

Finally, from an educational standpoint, active learning strategies were incorporated in the experiment to give participants the opportunity to reflect and discuss the use of model annotations in an engineering context. Active learning can be defined as instruction where learners play an active role in the process [SSJ05], [Mic06]. To this end, all participants were debriefed following the experiment. They were told that they were given different annotated assemblies and shown the differences. We then used the annotated assembly assigned to the experimental group (i.e., with functional groups) to demonstrate how annotations can be filtered and to encourage discussion. Although additional studies are needed to draw more general conclusions, comments and feedback from experts and graduate students suggest that this debriefing activity was helpful to emphasize the importance of model annotation practices and the proposed functional classification. We speculate that exposing CAD users to annotated models through activities designed to highlight the interaction with the annotations seems to be a successful strategy to promote MBD in academia.

5. Discussion and Conclusions

Annotations are used extensively in industry to document the design of products. However, there is a gap between technology and practice, as most model-based annotation practices and tools in commercially available systems are essentially 2D mechanisms in annotation planes and used in the same manner as their counterparts in traditional engineering drawings. In addition, despite ongoing standardization efforts and the availability of more advanced model annotation tools, the theoretical frameworks in support of annotations to facilitate the communication and exchange of design information are far from the paradigm promoted by the MBE. The challenges found when annotating and interacting with the assembly model used in our study are a representative example of the gap that exists between theoretical approaches and initiatives, and the capabilities of current annotations tools in commercial CAD systems.

We distinguish between two categories of annotations: symbols and plain text. We argued that symbolic annotations are supported by well stablished standards, such as GD&T, which enable consistent and concise communications. However, the use of symbolic annotations, at a fundamental level, has not changed significantly since the language was originally developed in a drawing context, which raises the question of whether symbolic annotations should be reexamined to enable new types of interactions and automation.

The need for development is more significant in the realm of plain text annotations. Current CAD applications provide text editors to create and format text annotations and link them to the geometry. Visualization managers help users avoid visual cluttering on the screen and some software prototypes have been developed to manage annotations at a higher conceptual level. Yet, a comprehensive conceptual framework is needed to fully leverage the information conveyed by text annotations.

In our view, annotation tools should support interactive interrogation where the user can decide how to classify the notes based on their content. The classification should be simple, so decisions do not disrupt or interfere with the user's creative process. Even a simple classification can subtly predispose the user to put the focus on the functionality of the messages. Our classification is expected to greatly simplify subsequent automatic data mining procedures for annotation classification as well as for extracting knowledge from the annotations.

Complex ontologies are impractical and counterproductive, and designers are often reluctant to use them. In a 3D model annotation context, we advocate for a simple classification that focuses on the content of the messages in which information is structured by function. This approach is novel when compared to the current criteria that groups text annotations based on spatial orientation (i.e. "annotation views"). In our classification, text annotations are categorized in four functional types: objectives, requirements, rationale, and intentions. The four categories are further developed into eleven types of annotations based on their scope.

We are aware that our pilot experiment cannot be used to draw conclusive results regarding the proposed classification. However, the experimental design of the activity itself represents a contribution that provides valuable insights. Our decision to run a pre-experiment with two experts prior to conducting the full experiment with a group of engineering students proved successful, as it enabled us to refine both the classification and the task itself by building on the findings of the previous stage. In addition, our experiment also shed light on the educational component of 3D annotation practices and their value as communication tools in the classroom, which is critical to inform training strategies for the Model-Based Definition (MBD) and the Model-Based Enterprise (MBE). Nevertheless, we acknowledge that a more complete validation with a larger and more heterogeneous group of users is required to reach sufficient statistical power and draw generalizable conclusions. In this regard, we are not claiming a generalizable validation of the proposed classification (which requires higher statistical power and a priori fixing the hypothesis to evaluate). Instead, we have presented a pilot study that enabled us to (1) confirm the experimental design and technique developed, and (2) determine a classification of annotations which has been shown to be sufficiently practical for modeling tasks and simple enough to facilitate the shift from current criteria that groups textual annotations based on the spatial orientation (i.e. "annotation views") to an approach where information is structured based on the function.

Finally, we advocate for developing specialized tools for managing design annotations according to these criteria, which would provide an efficient mechanism for managing the annotations separately, or as a group to gain a global and collaborative view on the problem. Grouping annotations in functional "layers" could be a simple strategy to accomplish this functionality. Although many CAD systems already provide annotation layers to the designer, these mechanisms are designed to favor grouping by views (i.e., annotations can only be visualized and manipulated as part of annotation planes that are predefined in 3D space) instead of grouping by content, which would enable filtering annotations based on the information they convey. The limitation of the grouping criteria in current 3D annotation tools hinder their value as a mechanism to facilitate design decisions.

6. Acknowledgements

The authors would like to thank the anonymous reviewers for their feedback and their valuable input and suggestions for fixing and rearranging the various sections of the paper into a more cohesive and readable structure, which has contributed improve the overall quality and presentation of the material.

7. References

[ASM] American Society of Mechanical Engineers, ASME Y14-41, Digital Product Definition Data Practices. URL: https://www.asme.org/codes-standards/find-codes-standards/y14-41-digital-product-definition-data-practices.

[ASM17] American Society of Mechanical Engineers, ASME Y14-100, Engineering Drawing Practices. URL: https://www.asme.org/codes-standards/find-codes-standards/y14-100-engineering-drawing-practices/2017.

[CCJ14a] Camba, J., Contero, M., & Johnson, M. (2014) Management of Visual Clutter in Annotated 3D CAD Models: A Comparative Study. In: Marcus A. (eds) Design, User Experience, and Usability. User Experience Design for Diverse Interaction Platforms and Environments. DUXU 2014. Lecture Notes in Computer Science, vol 8518. Springer, Cham. https://doi.org/10.1007/978-3-319-07626-3_37.

[CCJ14b] Camba, J., Contero, M., Johnson, M., & Company, P. (2014) Extended 3D annotations as a new mechanism to explicitly communicate geometric design intent and increase CAD model reusability. Computer-Aided Design. 57, pp. 61-73. DOI: 10.1016/j.cad.2014.07.001

[CCO15] Company P., Contero M., Otey J. and Plumed R. (2015) Approach for developing coordinated rubrics to convey quality criteria in MCAD training. Computer-Aided Design (ISSN 0010-4485). Vol. 63, pp. 101–117. DOI: 10.1016/j.cad.2014.10.001.

[CCP23] Company P., Camba J.D., Patalano S., Lanzotti A. (2023). On the Adoption of the STEP Neutral Format for Procedural Mechanical CAD Models. Advances on Mechanics, Design Engineering and Manufacturing IV. JCM 2022. Lecture Notes in Mechanical Engineering. pp. 677–685. DOI: 10.1007/978-3-031-15928-2_59.

[CG21] Company, P., & González-Lluch, C. (2021). CAD 3D con SolidWorks Tomo I: Diseño básico. Publicacions Universitat Jaume I. Ed. 2. http://cad3dconsolidworks.uji.es.

[CNC20] Company, P., Naya, F., Contero, M., & Camba, J.D. (2020) On the role of geometric constraints to support design intent communication and model reusability. Computer-Aided Design and Applications. 17(1), pp. 61-76, DOI:10.14733/cadaps.2020.61-76.

[DBP09] Ding, L., Ball, A., Patel, M., Matthews, J., & Mullineux, G. (2009). Strategies for the Collaborative Use of CAD Product Models. In DS 58-8: Proceedings of ICED 09, the 17th International Conference on Engineering Design, Vol. 8, Design Information and Knowledge, Palo Alto, CA, USA, 24.-27.08. 2009 (pp. 123-134).

[DDF90] Deerwester S., Dumais S.T., Furnas G.W., Landauer T.K., Harshman R. (1990) Journal of the American Society for Information Science, 41(6), pp. 391-407.

[DJH02] Devoss D, Jasken J, Hayden D. (2002) Teaching Intracultural and Intercultural Communication: A Critique and Suggested Method. Journal of Business and Technical Communication, 16(1), pp.69-94. doi:10.1177/105065190201600100З.

[Fer18] Ferrari A. (2018) Natural Language Requirements Processing: from Research to Practice. 2018 ACM/IEEE 40th International Conference on Software Engineering: Companion Proceedings. Doi: 10.1145/3183440.3183467.

[Fre11] Frechette, S.P. (2011, June). Model based enterprise for manufacturing. In Proceedings of the 44th CIRP international conference on manufacturing systems.

[GKW15] Grabowik C., Kalinowski K., Kempa W., & Paprocka I. (2015) A method of computer aided design with self-generative models in NX Siemens environment. Modern Technologies in Industrial Engineering (ModTech2015). IOP Conf. Series: Materials Science and Engineering 95 (2015) 012123 doi:10.1088/1757-899X/95/1/012123.

[HMV11] Holtmann J., Meyer J., Von Detten M. (2011) Automatic Validation and Correction of Formalized, Textual Requirements. Fourth International Conference on Software Testing, Verification and Validation Workshops. DOI 10.1109/ICSTW.2011.17.

[HS17] Harding J.E., & Shepherd P. (2017) Meta-Parametric. Design. Design Studies, 52, pp. 73-95. https://doi.org/10.1016/j.destud.2016.09.005.

[ISOa] International Organization for Standardization. ISO 16792, Technical product documentation—Digital product definition data practices. URL: https://www.iso.org/standard/73871.html.

[ISOb] ISO 10303: 1994, Industrial Automation Systems and Integration—Product Data Representation and Exchange. (https://www.iso.org/search.html?q=10303).

[ISOc] ISO 10303-55: 2005, Industrial automation systems and integration — Product data representation and exchange — Part 55: Integrated generic resource: Procedural and hybrid representation. (https://www.iso.org/standard/38226.html).

[ISOd] ISO 10303-108: 2005, Industrial automation systems and integration — Product data representation and exchange — Part 108: Integrated application resource: Parameterization and constraints for explicit geometric product models (https://www.iso.org/standard/34697.html)

[KG08] Konrad, S., & Gall, M. (2008) Requirements engineering in the development of large-scale systems. In: Proceedings of the 16th International Requirements Engineering Conference, pp 217–222.

[KGG21] Koscinski V., Gambardella C., Gerstner E., Zappavigna M., Cassetti J., Mirakhorli M. (2021) A Natural Language Processing Technique for Formalization of Systems Requirement Specifications. Proceedings of the IEEE International Conference on Requirements Engineering, pp. 350-356.

[KR70] Kunz, W., & Rittel, H. W. (1970). Issues as elements of information systems (Vol. 131, p. 14). Berkeley, California: Institute of Urban and Regional Development, University of California.

[Leu09] Leuser, J. (2009) Challenges for semi-automatic trace recovery in the automotive domain. In: TEFSE '09: Proceedings of the 2009 ICSE Workshop on Traceability in Emerging Forms of Software Engineering, IEEE Computer Society, Washington, DC, USA, pp 31–35, DOI 10.1109/TEFSE.2009.5069580

[LMM12] Lash A., Murray K., Mocko G. (2012) Natural language processing applications in requirements engineering. Proceedings of the ASME Design Engineering Technical Conference 2(PARTS A AND B), pp. 541-549.

[Mic06] Michael J. (2006) Where's the evidence that active learning works? American Journal of Physiology - Advances in Physiology Education, 30(4), pp. 159-167. Doi: 10.1152/advan.00053.2006.

[Nor99] Norman, D. (1999) Affordances, conventions and design. Interactions, 6 (3), pp. 38-43. DOI:10.1145/301153.301168.

[OCC18] Otey J., Company P., Contero M. & Camba J.D. (2018) Revisiting the Design Intent Concept in the Context of Mechanical CAD Education. Computer-Aided Design and Applications. 15(1), pp. 47-60. DOI:10.1080/16864360.2017.1353733.

[OCC19] Otey J., Company P., Contero M. and Camba J.D. (2019) Assessment of Parametric Assembly Models Based on CAD Quality Dimensions. Computer-Aided Design and Applications, 16(4), pp. 628-653. DOI: 10.14733/cadaps.2019.628-653.

[RGB22] Rosso P., Gopsill J., Burgess S.C., Hicks B. (2022) Does CAD Smell Like Code? A Mapping Between Violation of Object Oriented Programming Design Principles and Computer Aided Design Modelling. Proceedings of the Design Society, Volume 2, pp. 1737 - 1746. DOI: 10.1017/pds.2022.176.

[Sch16] Schätzle J. Evaluate How the STEP Standard AP 242 Could Enable Knowledge Transfer between CAD and KBE Environments. Master Thesis, Norwegian University of Science and Technology, Department of Engineering Design and Materials, 2016. URI: http://hdl.handle.net/11250/2395690.

[SSJ05] Smith K.A., Sheppard S.D., Johnson D.W., Johnson R.T. (2005) Pedagogies of engagement: Classroom-based practices. Journal of Engineering Education, 94(1), pp. 87-101. Doi: 10.1002/j.2168-9830.2005.tb00831.x

[URM11] Uusitalo E., Raatikainen M., Männistö T., Tommila T. (2011) Structured natural language requirements in nuclear energy domain: Towards improving regulatory guidelines. 2011 4th International Workshop on Requirements Engineering and Law, RELAW 2011, Proceedings - Held in Conjunction with the 19th International Requirements Engineering Conference 6050274, pp. 67-73

[Wei98] Weiss E.H. (1998) Technical communication across cultures: Five philosophical questions. Journal of Business and Technical Communication, 12(2), pp. 253-269.

[Whi12] Whittenburg, M. (2012). Model-based enterprise: an innovative technology-enabled contract management approach. Journal of Contract Management, 10, 103-112.

[Wil99] Wilson W. (1999) Writing Effective Natural Language Requirements Specifications. CrossTalk. The Journal of Defense Software Engineering, pp 16-16.

[Wol09] Wolfe J. (2009) How Technical Communication Textbooks Fail Engineering Students. Technical Communication Quarterly, 18:4, pp. 351-375, DOI: 10.1080/10572250903149662.

[ZAF21] Zhao L., Alhoshan W., Ferrari A., Letsholo K.J., Ajagbe M.A., Chioasca E.-V., Batista-Navarro R.T (2021) Natural Language Processing for Requirements Engineering. ACM Computing Surveys, 54(3),3444689. Doi: 10.1145/3444689.