A Knowledge-Augmented Socio-Technical Assistance System for Product Engineering

Dominik Mittel[1], Andreas Hubert[2], Uppili Srinivasan[1], Alexander Perzylo[1], Daniel Lemberger[3]

[1]fortiss - Research Institute of the Free State of Bavaria, Munich, Germany,
Email: {mittel, venkatesan, perzylo}@fortiss.org
[2]University of Applied Sciences Aschaffenburg, Aschaffenburg, Germany, Email: andreas.hubert@th-ab.de
[3]IBO Digital GmbH, Kirchheim near Munich, Germany, Email: daniel.lemberger@ibo-tec.de

Abstract—**Manufacturing companies are exposed to increasingly complex products and shorter product engineering cycles. Unstructured data hinders the integration of knowledge over the different product engineering stages and complicates structured product development. However, combining an integrated view on relevant data sources following the Advanced Product Quality Planning (APQP) approach provides guidance for product engineers. In this paper, a semantic Knowledge Base (KB), a Process Execution System (PES), and a Computer Vision System (CVS) are introduced, which, in their interaction, compose a Socio-Technical Assistance System (STAS). We combine semantic models of production knowledge, APQP-guided product development, and ontology-based geometric representations of products and manufacturing resources. The PES coordinates the interaction with the user and other system components. The CVS tracks used tools and parts during the assembly and, therefore, enables traceability features and creates confidence in the quality of the assembly. As a result, the developed STAS prototype offers support from customer inquiry through product design and development to manufacturing and assembly, as well as after-sales support. The assistance system enables handling of complex products efficiently in order to reduce required times and costs.**

Index Terms—**Socio-Technical Systems, Product Development, Ontology, Semantic Technologies**

I. INTRODUCTION

The product engineering process is a crucial step for bringing products to the market. This is especially true for small and medium-sized enterprises (SMEs), which often produce highly specialized and customized products [1]. In these settings, the increased complexity of products, their heterogeneity, and small batch sizes burden the product engineering process. Shorter product life cycles force the companies to design, develop, and produce at higher frequencies, which emphasize the need for digital assistance systems [2], [3].

In this paper, a continuous assistance system is developed and implemented for the entire product engineering process, from product design and development to manufacturing and after-sales support. As an example use case, the product engineering process of an axial bearing is investigated. Currently, the product is designed, manufactured, and assembled mostly due to the knowledge of domain experts. This results in the dependency of companies on individual persons, unclear processes, and a lack of automated documentation [4].

The developed assistance system links various data sources, interacts with the different stakeholders, and combines heterogeneous technologies towards a Socio-Technical Assistance System (STAS). Backbone of the STAS is a semantic Knowledge Base (KB) consisting of information from the various data sources and additional insights, e.g., from domain experts. We combine knowledge of products, processes, and resources (following the PPR paradigm [5]) with knowledge about structured product development to an ontology-based semantic backend. Also, the geometric models, using the ontology-based representation OntoBREP [6], of products and resources are stored in the KB which allows direct interconnections. Structured processes for product development are well known. Here, the Advanced Product Quality Planning (APQP) process is used, which is described in the following sections.

The KB provides an interface to a Process Execution System (PES), which enables dynamic interaction to all various users, i.e., the product engineer, the assembly worker, or the support engineer, and supports them during the product engineering process. Examples include the parameterization of product or processes characteristics and the selection of appropriate manufacturing resources. Lastly, a Computer Vision System (CVS) is added, which is active during the actual assembly. Linking the semantic models with the CVS results in many advantages. On the one hand, the image processing system can be parameterized individually according to requirements. On the other hand, observed assembly steps can be documented and verified. Semantic, anonymized storage facilitates downstream analyses and improves traceability. The CVS is designed to only require a reduced manual training effort. As an interface, the PES connects the heterogeneous technologies in a flexible manner and coordinates communication.

The rest of the paper is structured as follows. Section II states background knowledge and related work. Subsequently, the implementation and the components of the STAS are described. Section IV describes the use case and the results, followed by the conclusion of our work.

II. BACKGROUND & RELATED WORK

A. Advanced Product Quality Planning (APQP)

APQP was developed in the 1980s in the automotive sector. Its goal is to create a quality plan to reduce errors during production, increase product quality, and meet customer requirements. APQP is a methodical approach used in the design and development of products and manufacturing processes that involve the systematic planning, monitoring, and control of product quality throughout the entire product lifecycle [7].

A great number of tools and techniques are used as part of APQP, for example, the Failure Mode and Effect Analysis (FMEA), the Measurement System Analysis (MSA), and the Statistical Process Control (SPC). This involves several APQP documents in standard formats, such as text documents or tabular spreadsheets. These can be further formalized. Formalization with ontologies is a promising approach and have been investigated in the literature, also in our previous work [8].

B. Knowledge Base & Ontologies

Ontologies can be formulated by ontology specification languages, e.g., the Web Ontology Language (OWL), in this case, OWL 2[1]. Ontologies describe explicit knowledge about things, group of things, and relations between them. The knowledge is mostly structured in triples, i.e., in a subject-predicate-object structure, and defines different types of classes and properties, which are typically arranged hierarchically in taxonomies. The explicitly modeled knowledge may be expanded with implicit knowledge, which is automatically derived by reasoners that carry out logical inference based on different rule sets[2]. The ontological representation can be queried and made available with the SPARQL Protocol And RDF Query Language (SPARQL). With SPARQL, facts can be extracted, combined, and reinserted into the KB.

Semantic technologies have been widely investigated in product engineering in recent years. The Foundry Ontology[3], developed by Palantir, may be used to create semantic digital twins of business activities. The ontology connects knowledge representations with the real world to support decision-making processes. Palantir provides several applications, especially for data scientists, e.g., to identify process bottlenecks, but there is currently no application for product engineering. Cao et al. [9] define an ontology to connect design and manufacturing knowledge. They developed an analysis system enabled by semantic reasoning to assess manufacturability of products. At its core, a combination of feature-based modeling, production capabilities, and manufacturing rules are applied. The authors consider size, lead time, resources, and assembly constraints, but do not use geometric representations within their ontology.

Schlegl et al. [10] use an ontology for the use of common language to tackle increasingly complex products, faster development cycles, broad portfolios, and new methods for product and process planning. The authors highlight that according to Albers et al. [11] and their product generation development, there are two hypotheses: (1) every development is based on a reference system; (2) there are three types of variation: principle, attribute, and carryover. We reused this concept in parts of our STAS.

The Industrial Ontologies Foundry (IOF) was formed to create a suite of interoperable ontologies in the manufacturing domain [12]. It uses the Basic Formal Ontology (BFO) as its top-level ontology. The authors suggest a domain-specific mid-level ontology, which does not meet our requirements for a holistic product engineering approach. Schäfer et al. [13] describe a model-based architecture for product, production systems, and their interdependencies. The authors use a domain-specific ontology containing model elements, relationships, and attributes, which targets engineering activities during the initial design phase.

Fenza et al. [14] discuss the integration of Semantic Web technologies into Cyber-Physical Production Systems (CPPS) for enhancing interoperability and knowledge sharing across different systems in a smart manufacturing environment. The work emphasizes the use of the SOSA ontology[4] and C-SPARQL[5] for monitoring and computing key performance indicators, and demonstrates the role of a semantic component in enabling more efficient and informed decision-making processes in the context of Industry 4.0.

C. Socio-Technical Assistance System

In our preliminary work on digital worker assistance, a gesture recognition method was introduced [15] and integrated into an assistance system with self-learning capabilities [16]. Similarly, Besginow et al. [17] applied deep-learning techniques to improve process quality and to detect errors during assembly processes. Their solution classifies the actions of the human worker, instead of using semantic knowledge for parameterization of the vision system. There are also a number of papers investigating the fundamental topic of intelligent assistance systems with a focus on two central issues: Firstly, the support of the activity by a cognitive system providing context-sensitive information [18], [19] and secondly, gamification features to maintain concentration and long-term working power [20], [21]. While traditional assistance systems are usually optimized for individual processes in an elaborate manner, ontology-based methods focus on the automatic generation of dynamic assistive instructions from existing enterprise data, e.g., from the context-level information of users, tasks, environments, and information devices [22], or from the template of a product family [23].

The work presented shows interesting approaches, but do not offer a holistic system. In this work, semantic information models are integrated across products, processes, and resources, like assembly workcells. The intelligent interlinking of the semantic models with the aforementioned ontology-based geometry representation OntoBREP forms the basis for our STAS. Integration with PES and CVS empowers the digital assistance system to provide comprehensive support during the product engineering stages. It verifies customer requirements, generates work instructions, monitors tasks seamlessly, and allows downstream analyses.

---

[1] https://www.w3.org/TR/owl2-primer/

[2] https://www.w3.org/TR/owl2-profiles/

[3] https://www.palantir.com/platforms/foundry/

[4] https://www.w3.org/2015/spatial/wiki/SOSA_Ontology

[5] http://streamreasoning.org/resources/c-sparql

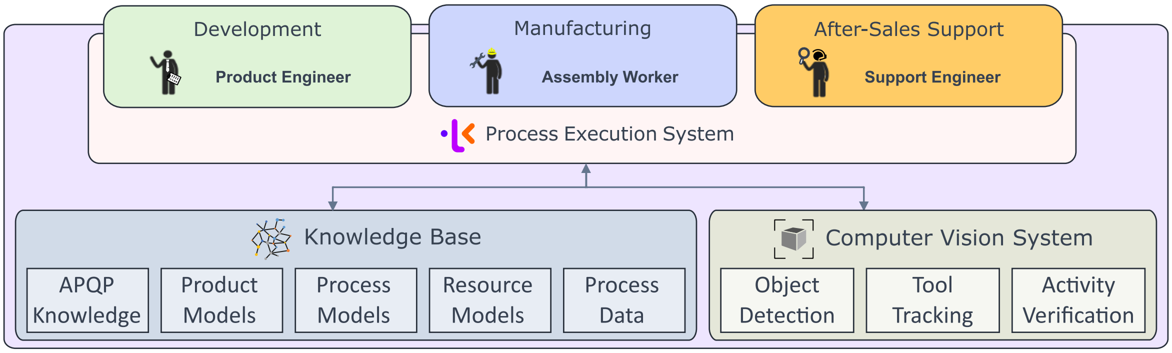


Fig. 1: Concept of a knowledge-augmented Socio-Technical Assistance System for product creation in manufacturing companies.

III. CONCEPT & IMPLEMENTATION

This work focuses on the design and implementation of a knowledge-augmented Socio-Technical Assistance System (STAS). It is an extension to a previously introduced concept [24]. Fig. 1 shows the different components of the STAS and the connections between them, i.e., the Knowledge Base (KB), the Computer Vision System (CVS), and the Process Execution System (PES). The STAS focuses on assisting its users, ranging from product engineers to assembly workers and support engineers, during the three stages of product engineering: product development, manufacturing & assembly, and after-sales support. The semantic KB is used as a persistent repository that stores, provides, and interprets data and knowledge needed or produced by the PES and the CVS. Using OWL2, it stores various types of data, including semantic models about products, processes, and resources, as well as knowledge about APQP to facilitate a structured product engineering process. Process data generated by the CVS during assembly runs is semantically encoded and also stored in the KB. Input into the system is made via the PES through user interfaces and via the CVS through visual detection and tracking algorithms. Outputs are generated and displayed via application-specific user interfaces of the PES. In their combination, the components enable support for the users, as well as tracing of activities and product parts. In the following, the specific components of the STAS are presented.

A. *Knowledge Base*

The core of the assistance system is the semantic knowledge stored in an RDF triplestore, in this case *Ontotext GraphDB*[6]. The knowledge is designed according to the PPR paradigm [5], i.e., the product, process, and resources are described, related to our previous work [25]. The concept is designed to support the product engineering process from the design of the product and its production process to the actual assembly and after-sales support.

1) Product-Process-Resource: The KB stores ontologies to describe the implemented products, the capabilities of resources, like tools and workers, and the processes. The processes combine the requirements of a specific product and the capabilities of the tools to create a feasible process execution. A process consists of individual tasks. In this work, also effects of processes and tasks on the products are described semantically. The products are described by specific characteristics and their geometric representation. OntoBREP [6] is used to represent the exact geometry of products and resources, automatically extracted from the geometry representation given by neutral CAD files like STEP or IGES.

The OntoBREP ontology stores every entity of the geometric model, e.g., every solid, face, edge, and vertex, as its own individual. Consequently, the entities can be directly linked to other knowledge. The geometric entities can be used to parameterize a process. For example, the faces of the bearing can be linked to specific processes like finishing or to control dimensions and their tolerances. Also, the area of a face or the length of an edge could be used for parameterization. Listing 1 shows an excerpt of the ontology file of the geometric representation of the holding ring of the axial bearing in Turtle syntax. The main geometric entity is the *compound*, which contains *solids* and further geometric entities. In this case, the *face* represented by a *cylindrical surface* is shown. The surface is defined by its type, direction, position and radius. These entities can be queried and adapted via SPARQL queries.

The geometric entities can be visualized by the self-developed Angular-based GUI called *OntoBREP Viewer*. It relies only on the semantically represented model and can also retrieve associated data from the KB. Furthermore, the viewer could be integrated into other applications like the PES to emphasize current tasks with appropriate visualization and information. Parts of the bearing visualized in the OntoBREP Viewer can be seen in Fig. 4.

2) Manufacturing & Assembly: A crucial part of manufacturing and assembly is the design of the processes and tasks. The individual tasks are described with different properties, like a *tool* or have some behavior properties, e.g., the *Mapping-Update*, which defines how the abstract descriptions are mapped to the specific description of the current workcell. We adapt the mapping behavior from [25], i.e., the processes and tasks are described in an *abstract* way and mapped to a *specific* workcell to get a *specific* process plan via SPARQL queries. In this case, an important property is the description

---

[6] https://www.ontotext.com/graphdb/

```
@prefix : <http://www.fortiss.org/ont/ontobrep/holdingring.ttl#> .
@prefix cad: <http://www.fortiss.org/ont/ontobrep#> .
:Compound1
    cad:contains :Solid1 ; a cad:Compound, owl:NamedIndividual .
:Solid
    cad:boundedBy :Shell1 ; cad:hasBoundingBox :BoundingBox1 ;
    cad:volume 8931.434878 ; a cad:Solid, owl:NamedIndividual .
:Shell1
    cad:contains :Face1, :Face10, :Face100, ...
    a cad:Shell, owl:NamedIndividual .
:Face3
    cad:area 29.311 ; cad:boundedBy :Wire3 ; cad:color :ColorRGB2 ;
    cad:position :Position14 ; cad:representedBy :CylindricalSurface1 ;
    cad:triangulatedBy :Triangulation3 ; a cad:Face, owl:NamedIndividual
      .
:CylindricalSurface1
    cad:direction :Vector2260 ; cad:position :Position15 ;
    cad:radius 1.5 ; a cad:CylindricalSurface, owl:NamedIndividual .
```

Listing 1: Excerpt of an OntoBREP model in Turtle syntax showing the topological structure and relations between its entities, i.e., from compound to solid, from solid to shell, from shell to face, and from face to surface (in this case a *CylindricalSurface* with its position, orientation, and radius).

of the effect. Therefore, the *Effect-Update* property defines the impact on the semantic models after the execution of the defined process or task. For the current use case, a new screw tightening task must be implemented and parameterized. The explanation of how the geometric information parameterizes the task is given in Section IV.

3) APQP Ontology: The KB aims to formalize APQP knowledge in a semantic framework, providing better insights beyond the mere reconstruction of documents. Utilizing the PPR paradigm, APQP knowledge from diverse formats and various documents, spreadsheets, or tables can be formalized. Important documents for APQP are, e.g., the customer compliance document or compliance matrix and the control plan. In this work, we focus on the control plan. The ontology design for the APQP ontology also focusing on different documents and APQP data can be found in [8].

The control plan is a key APQP document to develop and produce high-quality products that meet customer requirements. Important information are added to the PPR models. Product related information, such as the component material, critical geometric dimensions, or tolerances, are added to the product model. The added information to the resource model, are the used machinery and tools or the required human resources for production. Finally, the process model gets enhanced with knowledge about the process sequence, control dimensions achieved at each process, tolerances, measurement methodology, or frequency of inspection, which are vital to APQP. Furthermore, the process model links products and resources, i.e., which product is processed and what resources are needed for the process. The semantic approach enables structured knowledge modeling and linking accross various phases of product engineering.

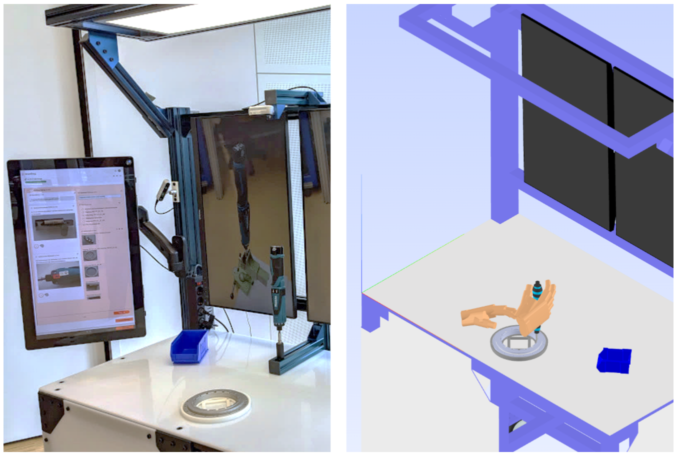


Fig. 2: Manual assembly workcell equipped with screens to show instructions and associated context information, as well as two RGB-D sensors to track hands, tools, and objects (left). Visualization of tracking information semantically labeled and stored in the Knowledge Base (right). Adapted from [24].

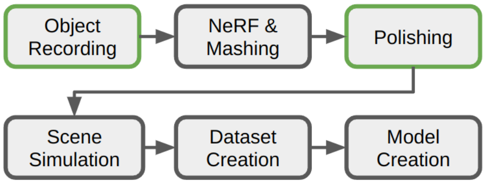


Fig. 3: Pipeline for creation of tracking models. In green are the manual steps and in gray are the automated steps.

B. Computer Vision System

The CVS is part of an assembly workcell, assisting during the assembly steps of a product. The real assembly workcell is depicted in Fig. 2 (left). The goal of the CVS is to gather and document as much information as possible about the assembly process. This information can be used to verify the process in order to increase the process reliability and therefore the confidence of the assembly worker. Central to the STAS is the use of only cameras together with detection methods that leverage both 2D box detection and 6D object tracking. This enables the accurate localization of most objects, including tools and worker's hands, within the workspace.

A crucial aspect of this assistance system is the establishment of a pipeline for the development and application of models capable of 2D detection and 6D pose estimation. This pipeline, see Fig. 3, is characterized by its unique ability to train models using only simulated data, eliminating the need for physical prototypes or extensive manual data collection. The required manual steps are to provide a 3D model of the actual object needing to be tracked.

As 3D modeling itself can be quite time- and resource-consuming, Neural Radiance Fields (NeRF) [26] are utilized for model generation. Although the model quality based on NeRF methods does not come close to a photorealistic representation, it still offers good tracking accuracy. The advantage of NeRF is the simplicity of generating digital models and the fact that it does not require expert knowledge in 3D modeling.

C. Process Execution System

In product engineering, companies are faced with complex challenges that require innovative solutions for effective management. The PES is used in several domains to design, adapt, manage, automatize, optimize, and analyze processes. Crucial to the effectiveness of PESs is the ability to adapt to new processes and requirements so that process experiences can be seamlessly incorporated into workflows [27]. These require a structured user interface. Furthermore, the PES must communicate with different components of the system.

In this work, *trustkey*[7] is used as the PES. The setup of a cloud platform allows parallel collaboration. trustkey orchestrates the organization of both data and processes into action packs, resources, and knowledge repositories. This enables and facilitates dynamic connections with external systems, such as the previously introduced integration with the semantic KB. This means that semantic models can be accessed and flexibly integrated in various phases of product engineering. Communication with the KB is done via dynamically adapted SPARQL queries. The PES can therefore receive and adapt the stored knowledge. In addition, the states of the systems, the KB, the PES, and the CVS are also synchronized via events triggered by trustkey.

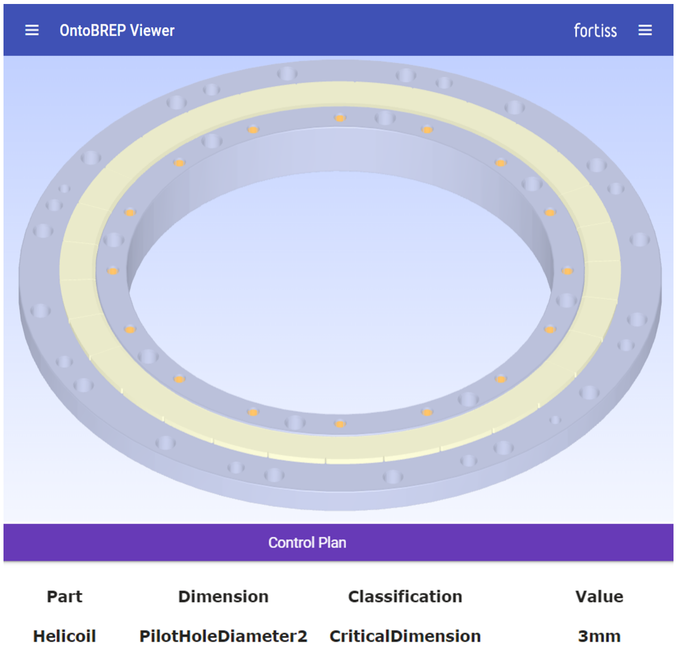
trustkey is an interaction tool for all stakeholders, as described in Fig. 1. For the different stages, the users modify and interact with the same data. The creation of a process starts with the creation of an action pack. Afterwards, the process can be designed, adapted, and launched. This allows structured processes to be defined and flexibly parameterized.

IV. USE CASE & RESULTS

This section describes the use case in detail and shows the implemented solution for the design, manufacturing, and after-sales phases of the product.

A. Use Case

This work investigates the product engineering process of an axial bearing. In particular, we focus on the design, manufacturing, and assembly phases of the bearing and take after-sales support into account. The manufacturing and assembly phases are investigated together and work in a similar way. The axial bearing consists of an aluminum alloy and has to reach high-quality standards to ensure safe operation. Furthermore, the bearing consists of a holding ring, a carrier ring, plastic sliding elements, and an outer ring. These different subparts are finally placed and screwed together crosswise to prevent tension. An electric drill is used for this. The following considerations are mostly based on this crosswise screw tightening task. The bearing can be used in different domains but is mostly designed for the automotive or aviation industry. The product is, therefore, intensively designed and tested to fulfill certain standards and certificates and the resulting requirements.



Part	Dimension	Classification	Value
Helicoil	PilotHoleDiameter2	CriticalDimension	3mm

Fig. 4: Visualization of the highlighted screw thread inserts (in orange) due to interlinking of the product's OntoBREP model and the APQP control plan with SPARQL queries.

B. Product Development with APQP

The product development phase starts with the customer inquiry and is an iterative approach to define the product characteristics. Critical part is to define the needed requirements and the interdependencies of involved processes, components, materials, etc. The design and development phase also has an impact on the other phases. Whenever a company receives a customer inquiry regarding a product, the product engineer has to go through a diverse set of documents to obtain generic information regarding the product. This includes information like individual components of the product, the material used for each component, material composition and standards, sequential processes that each part goes through, important geometric dimensions, or functional values. One of the goals of a semantic backend for APQP knowledge is to address this issue to shorten the product development time. In addition, the centralized KB allows easy knowledge access and reuse.

As mentioned before, the control plan has a central role in ensuring product quality throughout the production lifecycle. Traditionally, it is confined to text documents, tabular spreadsheets, or database formats within a quality management system. It includes crucial information about part geometry, dimensions, tolerances, process data, or measurement methods and frequencies. However, there is a need for enhanced visualization, better understanding, and information extraction. One effective approach is to integrate the control plan data with the geometric representation. To enhance visualization, the

---

[7] https://www.trustkey.eu/

```
PREFIX cad: <http://www.fortiss.org/ont/ontobrep#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?surface ?radius WHERE {
    VALUES (?object ?radiusControl)
      {(<http://www.fortiss.org/ont/perception-1#Bearing-1> 1.5)}
    ?object   cad:hasShape ?compound .
    ?compound rdf:type cad:Compound ; cad:contains* ?solid .
    ?solid    rdf:type cad:Solid ; cad:boundedBy ?shell .
    ?shell    cad:contains ?face .
    ?face     cad:representedBy ?surface .
    ?surface  rdf:type cad:CylindricalSurface ; cad:radius ?radius .
    FILTER (?radius == ?radiusControl ) . }
```

Listing 2: Excerpt of a SPARQL SELECT query to retrieve cylindrical surfaces and their radii that match the dimensions of the screw thread inserts of a specified object.

semantic APQP knowledge is interconnected with OntoBREP via SPARQL queries. The Angular-based OntoBREP Viewer frontend allows to select specific control dimension and shows appropriate context knowledge. The selection parameterizes the predefined SPARQL query, which scans the geometric models and returns the corresponding face. The OntoBREP Viewer highlights the resulted face. This allows the user to visualize specific dimensions from the control plan, which reduces the dependency on domain experts.

The critical dimension of the pilot holes for the screw tightening task is considered as an example. Fig. 4 shows the bearing with the highlighted pilot holes. The automatic parameterized SPARQL query, which asks for the surfaces due to the control dimension, the given bearing, and the surface type, can be seen in Listing 2. The visualization enables the product engineer to verify the critical dimension and to control the position of the highlighted face.

C. Manufacturing & Assembly

During manufacturing and assembly, machines and workers should be parameterized or guided, respectively. In both cases, the product engineer needs to define which process steps are necessary, needs to parameterize them, and has to prepare the data properly. Therefore, the parameterization and preparation are just as varied as the product itself. In this use case, we focus on the assembly of the bearing and the needed instructions. A crucial step is the screw tightening task. The worker needs to fulfill both a defined order for applying the screws and ensure they are tightened with a certain torque.

The PES trustkey gives guidance for the different tasks that need to be done. Example tasks one and three, i.e., the verification task and the crosswise screw tightening task, are depicted in Fig. 6. The PES can be automatically created by the given semantic process descriptions. The visualization is done by automatic screenshots from the OntoBREP Viewer for the different parts. Fig. 6a shows all needed parts and tools queried from the semantic task models. Also, the final assembled bearing is given to instruct the worker.

The crosswise screw tightening task is shown in Fig. 6b. Furthermore, an instruction text is given, and the bearing is visualized in the OntoBREP Viewer. The worker is asked to tighten 16 screws crosswise to combine the carrier ring and the holding ring. Therefore, the holding ring is marked via SPARQL queries to emphasize the screw holes. Each of the screws should then be checked again with a torque meter. The assistance system also supports the worker with live process information, e.g., screw positions, as well as further documents and analytical information.

```
@prefix task: <http://www.fortiss.org/ont/task#> .
@prefix process-bearing: <http://www.fortiss.org/ont/process-bearing#> .
process-bearing:AssemblyTask-3
      rdf:type owl:NamedIndividual , task:CrosswiseTighteningTask ;
      core:hasActor process-bearing:Actor-1 ;
      core:hasNext process-bearing:AssemblyTask-4 ;
      core:hasPickObject process-bearing:Screw-1 ;
      core:hasPlaceObject process-bearing:HoldingRing-1 ;
      core:hasTool process-bearing:Screwdriver-1 .
task:CrosswiseTighteningTask
      rdf:type owl:Class ;
      rdfs:subClassOf core:PickAndPlaceTask ,
      [ rdf:type owl:Restriction ;
        owl:onProperty core:hasEffectUpdate ;
        owl:hasValue task:CrosswiseTightening-EffectUpdate ] ,
      [ rdf:type owl:Restriction ;
        owl:onProperty core:hasMappingUpdate ;
        owl:hasValue task:CrosswiseTightening-MappingUpdate ] .
```

Listing 3: Excerpt of an abstract task model in Turtle syntax showing *AssemblyTask-3* of type *CrosswiseTighteningTask* and the corresponding class definition.

The CVS uses deep learning models to detect and track 6D object poses from multicamera RGB-D images. With the 6D pose estimation based on the mentioned 3D NeRF-generated models, our system could achieve an approximate accuracy of 1 cm in locating the screwdriver tip at a distance of about 1 meter from the camera. This is accurate enough to detect the target positions and verify them. Furthermore, the pose of the electric screwdriver could be tracked over time to verify the correct order when attaching the screws.

Fig. 5 shows the result of the tracked and matched screw positions. The duration in the vicinity of the desired positions can be used to successfully determine which screw has been tightened and thus assess the screw sequence. Furthermore, the CVS, parameterized by the KB, is able to reason about every screw being checked with a torque meter again by the worker. With this capability, the assistance system can also be used in onboarding and production scenarios.

Listing 3 depicts the semantic description of the screw tightening task in Turtle syntax. The screw pattern and their target positions are queried from the geometric representation via SPARQL. The SPARQL query for this task parameterization is also stored in the KB and given by the *CrosswiseTightening-MappingUpdate*. Other properties are the type of screws, which tool to use, and also the effect after the task is completed. Further properties of the specific parts like the screws, e.g. length or thread, or the tool are not shown, but are nevertheless important.

In addition to the screw tightening task, all other assembly steps must also be tracked by the CVS. The KB takes over the parameterization of the perception system so that only

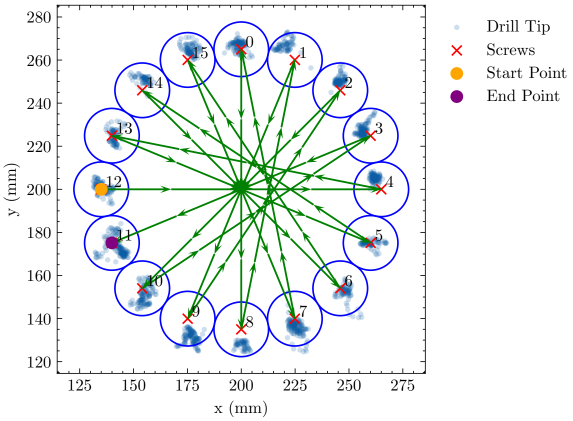


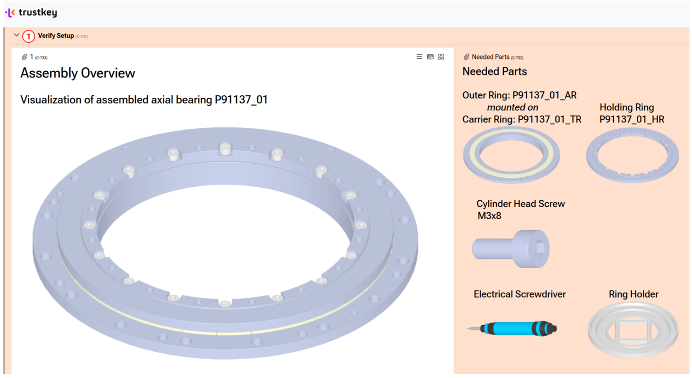
Fig. 5: Recognition of tool activity via the position-based matching of screwdriver tip and screw holes. The detection is carried out in 3D, but projected to a 2D plane for visualization purposes. Blue circles around the screw centers mark the area, where a tip position counts as a screw-tightening position.

the necessary detection models are used for each step. This makes it possible to use more complex models with the same computing power, as not every object or resource has to be tracked at all times. Parameterization also benefits from semantic knowledge about the process. For example, if parts are combined in one step, such as here the holding ring with the carrier ring, the semantic models are updated due to the modeled effects. The semantic information allows to only track one part, in this case the holding ring.

D. After-Sales Support

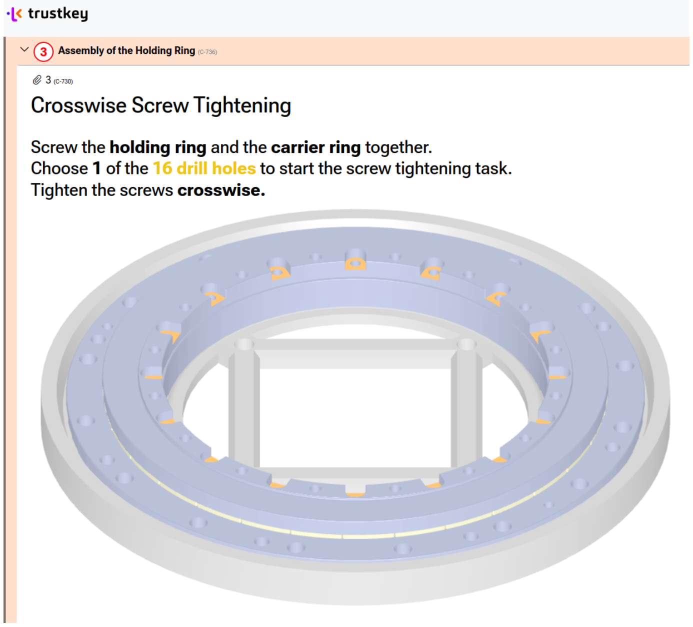
After delivering a product, the company needs to guarantee after-sales support. If a problem occurs, the company has to assess, if there were irregularities during the manufacturing of a product. The KB stores all received data from the perception system. On the one hand, raw data such as detected poses is stored in the KB. On the other hand, generated events are transmitted. For every perception data point, an individual is created that stores the used sensor and algorithm, and the detected objects and their poses. The properties are aligned to the Semantic Sensor Network Ontology[8].

Persistent semantic logging enables multiple advantages. Firstly, the data can be flexibly analyzed using SPARQL queries. For example, if a reclamation comes in, the support engineer may check the specific manufacturing run of the affected product. In particular, the poses of the worker's hands or the appropriate use of specific tools. With SPARQL queries, the poses can be queried very precisely. In Listing 4, an example query for the screw tightening task is given. The query searches for the positions of the tooltip of the electric screwdriver. Furthermore, the target positions that were used in the assembly can be retrieved. With this information, the distances between them can be easily filtered so that not every data point needs to be investigated manually. Secondly, every recorded data point can be visualized by the OntoBREP Viewer due to the availability of geometric representations. Not only the raw data points can be examined, but all points of interest that are retrieved by the predefined SPARQL query can be visualized and even animated (see Fig. 2 on the right). Thirdly, the stored data does not include critical data that could potentially violate worker privacy. Only the raw data points and abstract derived annotations are stored. Furthermore, the geometric representation to visualize human actions, e.g., the hands of the human worker guiding a tool in the workcell, is designed in a generic fashion.



Visualization of assembled axial bearing P91137_01

(a) Verification of the workcell setup with all needed parts of the assembly, the tools, and other equipment.



Crosswise Screw Tightening

Screw the **holding ring** and the **carrier ring** together.
Choose **1** of the 16 drill holes to start the screw tightening task.
Tighten the screws **crosswise.**

(b) Description of the crosswise screw tightening task with automatically highlighted drill holes (in orange) detected by evaluating SPARQL queries on the semantic geometry model.

Fig. 6: Configuration of PES component *trustkey* for presenting task descriptions to assembly workers based on generated images of products and tools from the OntoBREP Viewer.

---

[8] https://www.w3.org/TR/vocab-ssn/

```
PREFIX cad: <http://www.fortiss.org/ont/ontobrep#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX task: <http://www.fortiss.org/ont/task#>
SELECT ?object ?task ?targetTransform ?toolTransform ?tooltip WHERE {
  VALUES (?object ?taskType)
  {( <http://www.fortiss.org/ont/perception-1#Bearing-1>
     <http://www.fortiss.org/ont/task#CrosswiseTighteningTask> )}
  ?task rdf:type ?taskType ; core:object ?object ;
           core:hasTool ?tool ; core:targetTransform ?targetTransform .
  ?tool    core:hasTooltipTransform ?tooltip ;
           sosa:isFeatureOfInterestOf ?observation .
  ?observation core:hasTransform ?toolTransform . }
```

Listing 4: Simplified SPARQL SELECT query to extract required information for comparing a detected position of the screwdriver tool (given as a tool transformation in an observation) and the target position specified in a task.

V. CONCLUSION

The current trajectory in manufacturing, characterized by increased product complexity, variant diversity, and smaller batch sizes, necessitates continual refinement of product engineering processes. In this work, the technical concept of a knowledge-augmented Socio-Technical Assistance System is introduced. We showcase our solution, integrating a semantic Knowledge Base, a Process Execution System, and a Computer Vision System at different stages of product engineering. Different user roles interact with the Process Execution System to receive assistance during product development, manufacturing, and after-sales customer requests. Despite having very different requirements and intentions, a common and shared representation of data and knowledge is used, avoiding manual data transfers or the need for data conversions. With the system, product development cycles can be shortened and process reliablity during the different stages may be improved. The use of semantic technologies has to be seen as a trade-off between extensive initial modeling efforts and large potential efficiency and productivity gains through a higher level of automation and autonomy in the interpretation of data.

ACKNOWLEDGMENT

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany in the project DiProLeA (grant number 02J19B122).

REFERENCES

[1] S. Mittal, M. A. Khan, D. Romero, and T. Wuest, “A critical review of smart manufacturing & industry 4.0 maturity models: Implications for small and medium-sized enterprises (smes),” *Journal of Manufacturing Systems*, vol. 49, pp. 194–214, 2018.

[2] T. Keller, C. Bayer, P. Bausch, and J. Metternich, “Benefit evaluation of digital assistance systems for assembly workstations,” *Procedia CIRP*, vol. 81, pp. 441–446, 2019, 52nd CIRP Conference on Manufacturing Systems (CMS), Ljubljana, Slovenia, June 12-14, 2019.

[3] B. G. Mark, E. Rauch, and D. T. Matt, “Worker assistance systems in manufacturing: A review of the state of the art and future directions,” *Journal of Manufacturing Systems*, vol. 59, pp. 228–250, 2021.

[4] V. Carneiro, A. Barata da Rocha, B. Rangel, and J. L. Alves, “Design management and the sme product development process: A bibliometric analysis and review,” *She Ji: The Journal of Design, Economics, and Innovation*, vol. 7, no. 2, pp. 197–222, 2021.

[5] A. Björkelund, H. Bruyninckx, J. Malec, K. Nilsson, and P. Nugues, “Knowledge for intelligent industrial robots,” in *2012 AAAI Spring Symposium Series*, 2012.

[6] A. Perzylo, N. Somani, M. Rickert, and A. Knoll, “An ontology for CAD data and geometric constraints as a link between product models and semantic robot task descriptions,” in *Proc IEEE Int. Conf. Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015, pp. 4197–4203.

[7] Automotive Industry Action Group (AIAG), *Advanced Product Quality Planning and Control Plan (APQP)*. Southfield: Automotive Industry Action Group, 2008.

[8] U. Srinivasan, D. Mittel, D. Lemberger, and A. Perzylo, “Structured product engineering: Ontology-based advanced product quality planning,” 2024, 18th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 10-12 July 2024.

[9] J. Cao, E. Vakaj, R. K. Soman, and D. M. Hall, “Ontology-based manufacturability analysis automation for industrialized construction,” *Automation in Construction*, vol. 139, p. 104277, 2022.

[10] M. Schlegel, I. Wiederkehr, S. Rapp, C. Koldewey, A. Albers, and R. Dumitrescu, “Ontology for future-robust product portfolio evolution: A basis for the development of models and methods,” *Procedia CIRP*, vol. 119, pp. 764–769, 2023.

[11] A. Albers and S. Rapp, *Model of SGE: System Generation Engineering as Basis for Structured Planning and Management of Development*. Cham: Springer International Publishing, 2022, pp. 27–46.

[12] M. Drobnjakovic, F. Ameri, C. Will, B. Smith, and A. Jones, “The Industrial Ontologies Foundry (IOF) Core Ontology.”

[13] L. Schäfer, M. Günther, A. Martin, M. Lüpfert, C. Mandel, S. Rapp, G. Lanza, H. Anacker, A. Albers, and D. Köchling, “Systematics for an Integrative Modelling of Product and Production System,” *Procedia CIRP*, vol. 118, pp. 104–109, 2023.

[14] G. Fenza, M. Gallo, V. Loia, D. Marino, F. Orciuoli, and A. Volpe, *Semantic CPPS in Industry 4.0*. Springer Intern. Publishing, 2020.

[15] S. Köhler, K. Doll, S. Kebinger, D. Schmitt, M. Kröhn, M. Fried, and R. Börsig, “Gestenerkennung in einem hochautomatisiert lernenden Assistenzsystem für manuelle Montageprozesse,” in *AUTOMATION 2018, VDI-Berichte 2330*, 2018, pp. 145–156.

[16] J. Jung, A. Hubert, K. Doll, M. Kröhn, and J. Stadler, “Prozessinnovation: Selbstlernendes Assistenzsystem für die manuelle Montage,” 2022.

[17] A. Besginow, S. Büttner, N. Ukita, and C. Röcker, *Deep Learning-Based Action Detection for Continuous Quality Control in Interactive Assistance Systems*. Cham: Springer International Publishing, 2023.

[18] M. Buss and M. Beetz, “Cotesys—cognition for technical systems,” *KI-Künstliche Intelligenz*, vol. 24, no. 4, pp. 323–327, 2010.

[19] A. K. Noor, “Potential of cognitive computing and cognitive systems,” *Open Engineering*, vol. 5, no. 1, 2014.

[20] O. Korn, “Industrial playgrounds: how gamification helps to enrich work for elderly or impaired persons in production,” in *Proc. of the 4th ACM SIGCHI Symp. on Engineering Interactive Computing Systems*, 2012.

[21] O. Korn, A. Schmidt, and T. Hörz, “Assistive systems in production environments: exploring motion recognition and gamification,” in *Proceedings of the 5th international conference on pervasive technologies related to assistive environments*, 2012, pp. 1–5.

[22] A. Claeys, S. Hoedt, H. Van Landeghem, and J. Cottyn, “Generic model for managing context-aware assembly instructions,” *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1181–1186, 2016.

[23] T. Rusch, J. Steuer, M. König, R. Sochor, K. Fink, B. Stelzle, M. Romanelli, and F. Kerber, “Tool-based automatic generation of digital assembly instructions,” *Procedia CIRP*, vol. 99, pp. 454–459, 2021.

[24] D. Mittel, A. Hubert, J. Ding, and A. Perzylo, “Towards a knowledge-augmented socio-technical assistance system for product engineering,” in *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2023, pp. 1–4.

[25] A. Perzylo, I. Kessler, S. Profanter, and M. Rickert, “Toward a knowledge-based data backbone for seamless digital engineering in smart factories,” in *Proc. IEEE Int. Conf. Emerging Technologies And Factory Automation (ETFA)*, Vienna, Austria, 2020, pp. 164–171.

[26] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “Nerf: Representing scenes as neural radiance fields for view synthesis,” in *ECCV*, 2020.

[27] E. Deelman, D. Gannon, M. Shields, and I. Taylor, “Workflows and e-science: An overview of workflow system features and capabilities,” *Future Generation Computer Systems*, vol. 25, no. 5, pp. 528–540, 2009.